



Version 29.0: Winter '14

Visualforce 開発者ガイド

14



最終更新日: 2013/10/16

© Copyright 2000–2013 salesforce.com, inc. All rights reserved. Salesforce.com およびその他の名称や商標は、salesforce.com, inc. の登録商標です。本ドキュメントに記載されたその他の商標は、各社に所有権があります。

目次

第 1 章: Visualforce の導入.....	1
Visualforce とは.....	2
Visualforce ページとは?.....	2
Visualforce マークアップ.....	3
Visualforce コントローラ.....	3
Visualforce ページを使用できる場所.....	4
Visualforce をサポートするエディション.....	4
Visualforce の開発に必要な権限.....	4
Visualforce のアーキテクチャ設計.....	5
Visualforce の利点.....	6
Visualforce を使用する必要がある状況は?.....	7
Visualforce ページと Sコントロールの比較.....	8
Visualforce のバージョン設定方法.....	9
バージョン 29.0 の Visualforce の新機能.....	10
ドキュメント表記規則.....	10
第 2 章: Visualforce 開発用ツール.....	12
開発モードフッターの使用.....	13
Visualforce エディタについて.....	15
第 3 章: Visualforce のクイックスタート.....	18
Visualforce の正常なコンパイル.....	18
最初のページの作成.....	19
Visualforce による項目値の表示.....	20
Visualforce コンポーネントライブラリの使用.....	22
Visualforce ページによる既存のページの上書き.....	24
標準オブジェクトリストページへのリダイレクト.....	27
ページでの入力コンポーネントの使用.....	27
入力項目の表示ラベルの追加とカスタマイズ.....	29
フォームの項目のタブ順序の設定.....	30
ページへの連動項目の追加.....	32
Visualforce ダッシュボードコンポーネントの作成.....	36
カスタムオブジェクトの関連リストの表示.....	36
インライン編集の有効化.....	36
PDF としてのページの表示.....	39
ページでのデータのテーブルの作成.....	41
ページでのデータのテーブルの編集.....	42
ページでのクエリ文字列パラメータの使用.....	43
クエリ文字列パラメータの取得.....	44

リンクでのクエリ文字列パラメータの設定.....	45
単一ページでのクエリ文字列パラメータの取得と設定.....	46
ページでの Ajax の使用.....	47
コマンドリンクとボタンによる部分ページ更新の実装.....	47
非同期操作のための状況の提供.....	48
任意のコンポーネントでのイベントへの Ajax 動作の適用.....	49
第 4 章: Visualforce ページの外観と HTML 出力のカスタマイズ.....	51
Visualforce ページのスタイル設定.....	51
Salesforce スタイルの使用.....	51
カスタムスタイルの使用.....	54
コンポーネントの DOM ID を使用したスタイルの定義.....	55
HTML コメントと IE 条件付きコメント.....	56
Visualforce で追加または変更される HTML タグ.....	57
HTML5 文書型の整理の緩和.....	57
<html> および <body> タグの自動生成の手動による無効化.....	58
カスタム doctype の使用.....	59
カスタム ContentType の使用.....	60
Visualforce コンポーネントのカスタム HTML 属性の設定.....	61
HTML5 manifest 属性を使用したオフラインキャッシング.....	63
第 5 章: 標準コントローラ.....	64
Visualforce ページへの標準コントローラの関連付け	64
標準コントローラによるデータへのアクセス	65
標準コントローラアクションの使用.....	65
入力規則と標準コントローラ.....	67
標準コントローラを使用するページのスタイル設定.....	67
オブジェクトのアクセシビリティの確認.....	67
第 6 章: 標準リストコントローラ.....	69
Visualforce ページへの標準リストコントローラの関連付け	70
リストコントローラによるデータへのアクセス	70
標準リストコントローラアクションの使用.....	71
リストコントローラによるページネーション.....	72
標準リストコントローラによるリストビューの使用.....	72
リストコントローラによるレコードの編集.....	74
第 7 章: カスタムコントローラおよびコントローラ拡張.....	76
カスタムコントローラおよびコントローラ拡張とは?.....	76
カスタムコントローラの作成.....	77
コントローラ拡張の作成.....	79
カスタムリストコントローラの作成.....	80

コントローラメソッド.....	82
コントローラクラスのセキュリティ.....	85
大量のデータセットを使用した作業.....	85
ページ全体での参照のみモードの設定.....	86
コントローラメソッドでの参照のみモードの設定.....	87
カスタムコントローラおよびコントローラ拡張の作成に関する考慮事項.....	87
Visualforce ページ内の実行順序.....	88
Visualforce ページの get 要求の実行順序.....	89
Visualforce ページの postback 要求の実行順序.....	91
Visualforce ページの実行順序の例.....	93
カスタムコントローラおよびコントローラ拡張のテスト.....	100
入力規則とカスタムコントローラ.....	103
transient キーワードの使用.....	104
第 8 章: 高度な例.....	106
初めてのカスタムコントローラの作成.....	106
カスタムコントローラクラスの作成.....	106
getter メソッドの定義.....	107
action メソッドの定義.....	109
navigation メソッドの定義.....	111
ウィザードの作成.....	113
商談ウィザードコントローラ.....	114
商談ウィザードのステップ 1.....	116
商談ウィザードのステップ 2.....	118
商談ウィザードのステップ 3.....	119
高度な Visualforce ダッシュボードコンポーネント.....	120
Visualforce と Google Chart の統合.....	121
カスタムリストコントローラによるレコードの一括更新.....	125
第 9 章: Visualforce によるボタン、リンク、およびタブの上書き.....	128
標準リストコントローラを使用したタブの上書き.....	129
Visualforce のカスタムボタンおよびリンクの定義.....	130
標準リストコントローラを使用したカスタムリストボタンの追加.....	132
レコードタイプの表示.....	133
第 10 章: 静的リソースの使用.....	135
静的リソースの作成.....	135
Visualforce マークアップでの静的リソースの参照.....	136
第 11 章: カスタムコンポーネントの作成と使用.....	138
カスタムコンポーネントとは?.....	138
カスタムコンポーネントの定義.....	139

カスタムコンポーネントのマークアップ.....	140
Visualforce ページでのカスタムコンポーネントの使用.....	141
カスタムコンポーネントのバージョン設定の管理.....	141
カスタムコンポーネントの属性.....	142
カスタムコンポーネントコントローラ.....	143
第 12 章: 動的 Visualforce バインド.....	145
標準オブジェクトでの動的参照の使用.....	146
カスタムオブジェクトおよびパッケージでの動的参照の使用.....	154
Apex 対応付けとリストの参照.....	162
項目セットの使用.....	159
グローバル変数への動的参照.....	162
\$Resource を使用した静的リソースへの動的参照.....	162
\$action を使用した action メソッドへの動的参照.....	164
\$ObjectType を使用したスキーマ詳細への動的参照.....	166
第 13 章: 動的 Visualforce コンポーネント.....	170
動的コンポーネントの制限.....	171
動的コンポーネントの作成と表示.....	171
関連リストの使用例.....	174
第 14 章: Visualforce とメールの統合.....	180
Visualforce を使用したメールの送信.....	180
メッセージクラスを使用したカスタムコントローラの作成.....	180
メール添付ファイルの作成.....	182
Visualforce メールテンプレート.....	186
Visualforce メールテンプレートの作成.....	186
Visualforce メールテンプレートでのカスタムスタイルシートの使用.....	189
ファイルの添付.....	191
Visualforce メールテンプレート内のカスタムコントローラの使用.....	194
第 15 章: Visualforce Charting.....	196
Visualforce Charting の制限および考慮事項.....	196
Visualforce Charting のしくみ.....	197
単純なグラフ作成の例.....	197
グラフデータの提供.....	198
コントローラメソッドを使用したグラフデータの提供.....	199
JavaScript 関数を使用したグラフデータの提供.....	200
JavaScript 配列を使用したグラフデータの提供.....	201
グラフデータの形式.....	201
Visualforce Charting を使用した複雑なグラフの作成.....	202
更新されたデータによるグラフの更新.....	206

<apex:actionSupport> を使用したグラフデータの更新.....	206
JavaScript Remoting を使用したグラフデータの更新.....	208
グラフの外観の制御.....	212
グラフの色.....	213
グラフのレイアウトとアノテーション.....	214
棒グラフ.....	215
その他の線形系列グラフ.....	216
円グラフ.....	219
ゲージグラフ.....	220
レーダーグラフ.....	220
第 16 章: Visualforce でのフローの表示.....	222
Visualforce ページにフローを埋め込む.....	222
<flow:interview> を使用した高度な例.....	224
フローの finishLocation 属性の設定.....	226
フローのユーザインターフェースのカスタマイズ.....	228
第 17 章: Visualforce でのテンプレートの使用.....	230
<apex:composition> を使用したテンプレートの定義.....	230
<apex:include> を使用する既存ページの参照.....	233
第 18 章: モバイルデバイスの開発.....	236
Salesforce Classic とは?.....	236
Salesforce Classic および Visualforce Mobile を実行できるデバイス.....	236
モバイルアプリケーションの機能と制限事項.....	237
Visualforce Mobile を使用する必要がある状況.....	238
iPhone および BlackBerry 用のページの開発.....	239
iPhone の考慮事項.....	241
BlackBerry の考慮事項.....	243
クロスプラットフォームの互換性のあるページの開発.....	244
JavaScript ライブラリの使用.....	247
Visualforce ページのモバイル化.....	249
モバイル準備完了の Visualforce タブの作成.....	249
モバイル設定への Visualforce タブの追加.....	250
モバイル設定を作成する.....	250
データセットを定義する.....	251
モバイルオブジェクトプロパティを編集する.....	252
モバイルタブをカスタマイズする.....	252
Visualforce Mobile ページのテスト.....	253
例: iPhone 用の対応付けアプリケーションの作成.....	254
カスタムコントローラの作成.....	255
対応付けおよびリストビューの作成.....	256

詳細ページの作成.....	259
第 19 章: Force.com AppExchange アプリケーションへの Visualforce の追加.....	261
Visualforce ページとコンポーネントのパッケージバージョン設定の管理.....	262
第 20 章: Visualforce ページでの JavaScript の使用.....	264
JavaScript を使用したコンポーネントの参照.....	264
Visualforce での JavaScript ライブラリの使用.....	265
Apex コントローラの JavaScript Remoting.....	267
JavaScript Remoting の例.....	274
第 21 章: ベストプラクティス.....	276
Visualforce のパフォーマンス向上のためのベストプラクティス.....	276
コンポーネント ID へのアクセスのベストプラクティス.....	278
静的リソースのベストプラクティス.....	281
コントローラおよびコントローラ拡張のためのベストプラクティス.....	281
コンポーネント facet の使用のためのベストプラクティス.....	282
ページロックコンポーネントのベストプラクティス.....	284
PDF を表示するためのベストプラクティス.....	285
<apex:panelbar> のベストプラクティス.....	287
第 22 章: 標準のコンポーネントの参照.....	288
analytics:reportChart	288
apex:actionFunction	290
apex:actionPoller	292
apex:actionRegion	294
apex:actionStatus	295
apex:actionSupport	298
apex:areaSeries	300
apex:attribute	302
apex:axis	304
apex:barSeries.....	306
apex:canvasApp	309
apex:chart.....	312
apex:chartLabel.....	315
apex:chartTips	316
apex:column	317
apex:commandButton.....	323
apex:commandLink.....	326
apex:component.....	330
apex:componentBody	333
apex:composition	335
apex:dataList.....	336
apex:dataTable	338

apex:define	345
apex:detail	346
apex:dynamicComponent	348
apex:emailPublisher	349
apex:enhancedList	351
apex:facet.....	353
apex:flash	354
apex:form	355
apex:gaugeSeries	359
apex:iframe	361
apex:image	362
apex:include	364
apex:includeScript	365
apex:inlineEditSupport	366
apex:input	368
apex:inputCheckbox	371
apex:inputField.....	375
apex:inputFile	379
apex:inputHidden	382
apex:inputSecret	383
apex:inputText	386
apex:inputTextarea	389
apex:insert	393
apex:legend	394
apex:lineSeries	395
apex:listViews	397
apex:logCallPublisher	398
apex:message	400
apex:messages	401
apex:milestoneTracker	403
apex:outputField	404
apex:outputLabel	405
apex:outputLink	408
apex:outputPanel	411
apex:outputText	413
apex:page.....	416
apex:pageBlock	422
apex:pageBlockButtons	425
apex:pageBlockSection	428
apex:pageBlockSectionItem	431
apex:pageBlockTable	434
apex:pageMessage	440
apex:pageMessages	441
apex:panelBar	442

apex:panelBarItem	444
apex:panelGrid	447
apex:panelGroup	451
apex:param	452
apex:pieSeries	453
apex:radarSeries	455
apex:relatedList	457
apex:repeat	459
apex:scatterSeries	461
apex:scontrol	463
apex:sectionHeader	464
apex:selectCheckboxes	465
apex:selectList	470
apex:selectOption	474
apex:selectOptions	477
apex:selectRadio	478
apex:stylesheet	482
apex:tab	483
apex:tabPanel	486
apex:toolbar	490
apex:toolbarGroup	494
apex:variable	496
apex:vote	497
chatter:feed	497
chatter:feedWithFollowers	498
chatter:follow	499
chatter:follower	500
chatter:newsfeed	500
chatter:userPhotoUpload	501
chatteranswers:aboutme	501
chatteranswers:allfeeds	502
chatteranswers:changepassword	503
chatteranswers:categoryfilter	503
chatteranswers:feedfilter	504
chatteranswers:feeds	505
chatteranswers:forgotpassword	505
chatteranswers:forgotpasswordconfirm	506
chatteranswers:guestsignin	506
chatteranswers:help	507
chatteranswers:login	507
chatteranswers:registration	508
chatteranswers:searchask	509
chatteranswers:singleitemfeed	509
flow:interview	510

ideas:detailOutputLink	511
ideas:listOutputLink	512
ideas:profileListOutputLink	514
knowledge:articleCaseToolbar	516
knowledge:articleList	516
knowledge:articleRendererToolbar	518
knowledge:articleTypeList	518
knowledge:categoryList	519
liveAgent:clientChat.....	520
liveAgent:clientChatAlertMessage	520
liveAgent:clientChatEndButton.....	521
liveAgent:clientChatInput.....	522
liveAgent:clientChatLog.....	522
liveAgent:clientChatMessages.....	523
liveAgent:clientChatQueuePosition.....	524
liveAgent:clientChatSaveButton.....	524
liveAgent:clientChatSendButton.....	525
liveAgent:clientChatStatusMessage	525
messaging:attachment	526
messaging:emailHeader	527
messaging:emailTemplate	528
messaging:htmlEmailBody	530
messaging:plainTextEmailBody	531
site:googleAnalyticsTracking	532
site:previewAsAdmin	533
social:profileViewer.....	534
support:caseArticles	535
support:caseFeed	537
support:clickToDial	538
support:portalPublisher	539
topics:widget	540
 付録.....	541
付録 A: グローバル変数、関数、および式の演算子.....	541
グローバル変数.....	541
\$Action.....	543
\$Api.....	551
\$Component.....	551
\$ComponentLabel.....	552
\$CurrentPage.....	552
\$Label.....	553
\$Label.Site.....	553
\$ObjectType.....	555
\$Organization.....	561

\$Page.....	561
\$Profile.....	562
\$Resource.....	562
\$SControl.....	563
\$Setup.....	563
\$Site.....	564
\$System.OriginDateTime.....	565
\$User.....	566
\$User.UITheme および \$User.UIThemeDisplayed.....	566
\$UserRole.....	567
関数.....	567
式の演算子.....	579
付録 B: Apex および Visualforce 開発のセキュリティのヒント.....	582
クロスサイトスクリプト (XSS).....	582
Visualforce ページのエスケープされない出力と式.....	584
クロスサイトリクエストフォージェリ (CSRF).....	586
SOQL インジェクション.....	587
データアクセスコントロール.....	588
付録 C: Visualforce コントローラで使用する Apex クラス.....	590
ApexPages クラス.....	591
ApexPages インスタンスマソッド.....	591
Action クラス.....	593
Action インスタンスマソッド.....	594
Cookie クラス.....	595
Cookie インスタンスマソッド.....	597
IdeaStandardController クラス.....	599
IdeaStandardController インスタンスマソッド.....	600
IdeaStandardSetController クラス.....	601
IdeaStandardSetController インスタンスマソッド.....	603
KnowledgeArticleVersionStandardController クラス.....	604
KnowledgeArticleVersionStandardController インスタンスマソッド.....	606
Message クラス.....	607
Message インスタンスマソッド.....	608
PageReference クラス.....	610
PageReference インスタンスマソッド.....	613
SelectOption クラス.....	619
SelectOption インスタンスマソッド.....	620
StandardController クラス.....	623
StandardController インスタンスマソッド.....	624
StandardSetController クラス.....	629

StandardSetController

第1章

Visualforce の導入

過去数年にわたり、salesforce.com はオンデマンドアプリケーションを作成するための包括的なプラットフォームを作成してきました。他の高度なアプリケーション開発プラットフォームと同様に、Force.com プラットフォームでは次の事項を定義するための別個のツールを提供します。

- ・ データの構造 (データモデル)
- ・ データの操作方法の詳細を設定するルール (ビジネスロジック)
- ・ データの表示方法を指定するレイアウト (ユーザインターフェース)



メモ: データモデル、ビジネスロジックまたはユーザインターフェースに影響するかどうかに基づいたアプリケーション開発ツールの分割は、Model-View-Controller (MVC) (Model はデータモデル、View はユーザインターフェース、および Controller はビジネスロジックを表す) アプリケーション開発パターンとしても知られています。

アプリケーション用のデータモデルおよびビジネスロジックを作成するこのツールは、Force.com プラットフォームサーバでネイティブに実行される強力なソリューションです。ユーザインターフェースを定義する既存のツールには次のような特定の制限がありました。

- ・ ページレイアウト。アプリケーション開発者がレコード詳細ページで項目、ボタンおよび関連リストを編成できるポイント & クリックツールです。情報セットの表示方法については柔軟に対応できない場合があります。項目は必ず関連リストの上に表示する必要があります。ボタンは必ず項目の上に表示する必要があります。Sコントロールとカスタムリンクは特定の領域内にのみ配置できます。
- ・ Sコントロール。アプリケーション開発者が詳細ページまたはカスタムタブにカスタム HTML を表示できるツール。次の点を除いて、ページレイアウト以上の柔軟性を提供します。
 - ◊ ブラウザ内から実行する。そのため、一度に多数のレコードの値を表示または更新する場合はパフォーマンスが低下します。
 - ◊ カスタムユーザインターフェース要素に標準の Salesforce ページと同じデザインを提供する簡単な方法が備わっていない。
 - ◊ 開発者は項目の一意性およびメタデータのその他の連動関係を自分で適用する必要がある。



重要: Sコントロールは、Visualforce ページに置き換えられました。組織で以前に Sコントロールを使用していない場合、Sコントロールは作成できません。既存のSコントロールに影響はありません。今後も編集できます。

これらの理由から、salesforce.com は、Force.com プラットフォームに高度なカスタムユーザインターフェースを作成するための次世代のソリューションである Visualforce を導入しました。

関連リンク

[Visualforce のアーキテクチャ設計](#)

[Visualforce の利点](#)

[Visualforce をサポートするエディション](#)

[Visualforce ページとSコントロールの比較](#)

[Visualforce とは?](#)

[バージョン29.0 のVisualforce の新機能](#)

Visualforce とは?

Visualforce は、Force.com プラットフォームでネイティブにホストできる高度なカスタムユーザインターフェースを、開発者が作成できるようにするフレームワークです。Visualforce フレームワークには、HTML に似たタグベースのマークアップ言語が含まれています。

Visualforce マークアップ言語では、各 Visualforce タグが、ページのセクション、関連リスト、または項目など、大まかなユーザインターフェースコンポーネントまたはきめの細かいユーザインターフェースコンポーネントに対応しています。Visualforce コンポーネントの動作は、標準の Salesforce ページと同じロジックを使用して制御することも、開発者が独自のロジックを Apex で記述されたコントローラクラスと関連付けることもできます。

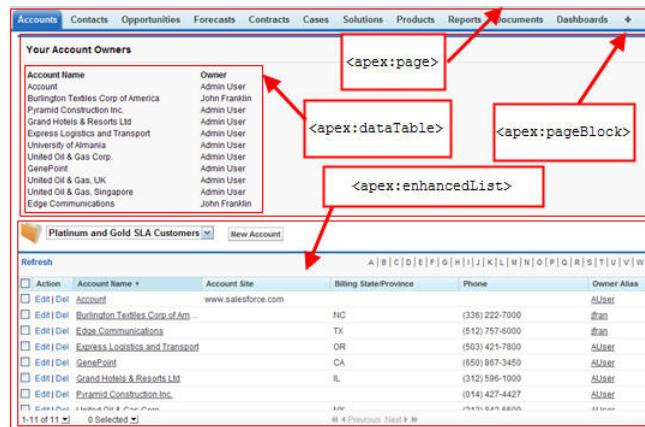


図 1 : Visualforce コンポーネントと、それに対応するタグのサンプル

関連リンク

[カスタムコントローラの作成](#)

[コントローラ拡張の作成](#)

Visualforce ページとは?

開発者は Visualforce を使用して Visualforce ページ定義を作成できます。ページ定義は次の 2 つの主要な要素で構成されます。

- Visualforce マークアップ
- Visualforce コントローラ

Visualforce マークアップ

Visualforce マークアップは、Visualforce タグ、HTML、JavaScript、または 1 つの `g p p` タグ内に埋め込まれている他の Web 対応コードで構成されています。マークアップでは、ページに含める必要のあるユーザインターフェースコンポーネントとその表示方法を定義します。

Visualforce コントローラ

Visualforce コントローラは、関連付けられた Visualforce マークアップで指定されたコンポーネントをユーザが操作(ボタンやリンクのクリックなど)したときの動作を指定する命令のセットです。コントローラを使用すると、ページに表示されるデータにアクセスでき、また、コンポーネントの動作を変更できます。

開発者は Force.com プラットフォームが提供する標準コントローラを使用するか、または Apex で記述されたクラスを含むカスタムコントローラロジックを追加できます。

- **標準コントローラ**は、標準の Salesforce ページで使用されているものと同じ機能およびロジックで構成されます。たとえば、標準取引先コントローラを使用する場合、Visualforce ページで [保存] ボタンをクリックした場合、標準の取引先編集ページで [保存] をクリックした場合と同じ動作が行われます。

ページで標準コントローラを使用した場合に、ユーザにそのオブジェクトへのアクセス権がないと、ページには権限がないというエラーメッセージが表示されます。オブジェクトへのユーザのアクセシビリティを確認し、コンポーネントを適切に表示することでこれを回避できます。

- **標準リストコントローラ**では、一連のレコードを表示または操作できる Visualforce ページを作成できます。レコードセットを使用する既存の Salesforce ページの例として、リストページ、関連リスト、一括アクションページなどがあります。
- カスタムコントローラは Apex で記述されるクラスで、標準コントローラを使用せずにすべてのページのロジックを実装します。カスタムコントローラを使用する場合、新しいナビゲーション要素または動作を定義できますが、標準コントローラにすでに定義された機能も再実装する必要があります。

他の Apex クラスと同様に、カスタムコントローラ全体はシステムモードで実行されます。このモードでは現在のユーザのオブジェクトと項目レベルの権限は無視されます。カスタムコントローラ内で、ユーザプロファイルを用いてアクセスするか否かを独自に決定することができます。

- コントローラ拡張は、Apex で記述されるクラスで、標準コントローラまたはカスタムコントローラの動作を追加するか、動作を上書きします。拡張を使用すれば、独自のカスタムロジックを追加する一方で、別のコントローラの機能も使用できます。

標準コントローラはユーザモードで実行し、現在のユーザの権限、項目レベルのセキュリティ、共有ルールが強制されるため、標準コントローラを拡張すると、ユーザ権限を重視する Visualforce ページを構築できます。拡張クラスはシステムモードで実行しますが、標準コントローラはユーザモードで実行します。カスタムコントローラと同様、ユーザプロファイルを参照してプログラムでアクセスさせるか否かを指定できます。



メモ: カスタムコントローラとコントローラ拡張クラスはシステムモードで実行されるため、ユーザ権限や項目レベルのセキュリティを無視しますが、クラス定義に `h g` キーワードを使用することによって、ユーザの組織の共有設定、ロール階層、および共有ルールを使用するかどうかを選択できます

す。詳細は、『[Force.com Apex コード開発者ガイド](#)』の「[ワードの使用](#)」を参照してください。

または [r ih sphi a キー i s](#)

Visualforce ページを使用できる場所

開発者は Visualforce ページを使用して次のことを実行できます。

- 取引先の [新規] ボタンまたは取引先責任者の [保存] ボタンなどの、標準ボタンを上書きする
- [取引先] タブホームページなどのタブ概要ページを上書きする
- カスタムタブを定義する
- 詳細ページレイアウトにコンポーネントを埋め込む
- ダッシュボードコンポーネントまたはカスタムヘルプページを作成する
- Salesforce コンソールのサイドバー（カスタムコンソールコンポーネント）をカスタマイズ、拡張、統合する

Visualforce をサポートするエディション

Visualforce は、Contact Manager Edition、Group Edition、Professional Edition、Enterprise Edition、Unlimited Edition、Performance Edition、および Developer Edition で使用できます。

Visualforce の開発に必要な権限

Visualforce の開発には、特定の活動に応じてさまざまな権限が必要です。

必要なユーザ権限	
Visualforce 開発モードを有効化する	「アプリケーションのカスタマイズ」
Visualforce ページを作成、編集、削除する	「アプリケーションのカスタマイズ」
カスタム Visualforce コンポーネントを作成、編集する	「アプリケーションのカスタマイズ」
カスタム Visualforce コントローラまたは Apex を編集する	「Apex 開発」
Visualforce ページセキュリティを設定する	「ユーザの管理」 および 「アプリケーションのカスタマイズ」
Visualforce ページのバージョン設定を行う	「アプリケーションのカスタマイズ」
静的リソースを作成、編集、削除する	「アプリケーションのカスタマイズ」
Visualforce タブを作成する	「アプリケーションのカスタマイズ」

Visualforce のアーキテクチャ設計

次のアーキテクチャの図に示すように、すべての Visualforce ページ全体は、開発者がページを作成する場合もエンドユーザがページを要求している場合も、両方とも Force.com プラットフォーム上で実行されます。

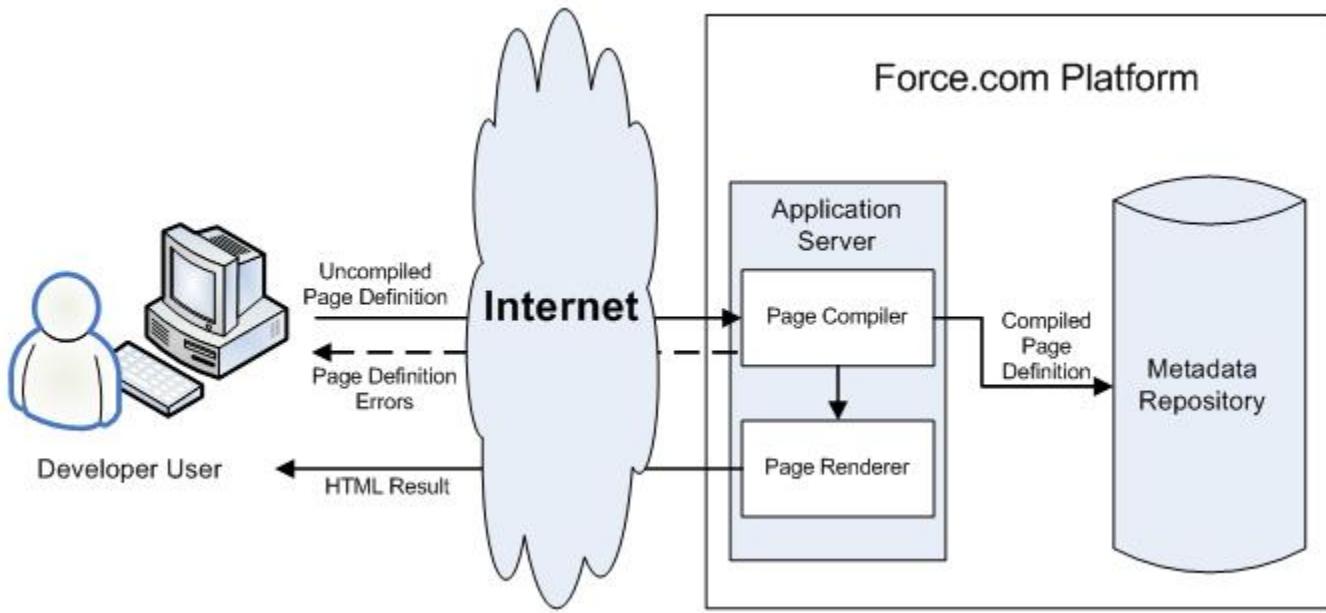


図 2: Visualforce システムアーキテクチャ - 開発モード

開発者が Visualforce ページの記述を完了して、プラットフォームに保存すると、プラットフォームアプリケーションサーバは、そのマークアップを、Visualforce レンダラによって解釈可能な抽象的な命令セットにコンパイルすることを試みます。コンパイルでエラーが生成されると、保存が中止され、開発者にエラーが返されます。それ以外の場合は、命令がメタデータリポジトリに保存され、Visualforce レンダラに送信されます。レンダラはその命令を HTML に変換し、開発者のビューを更新します。そのため、マークアップに変更が行われるたびに開発者に即座にフィードバックが提供されます。

以下のアーキテクチャの図では、開発者ではないユーザが Visualforce ページを要求するときのプロセスフローを示します。このページはすでに命令にコンパイル済みであるため、アプリケーションサーバはメタデータリポジトリからページを取得して、HTML に変換するためにそれを Visualforce レンダラに送信します。

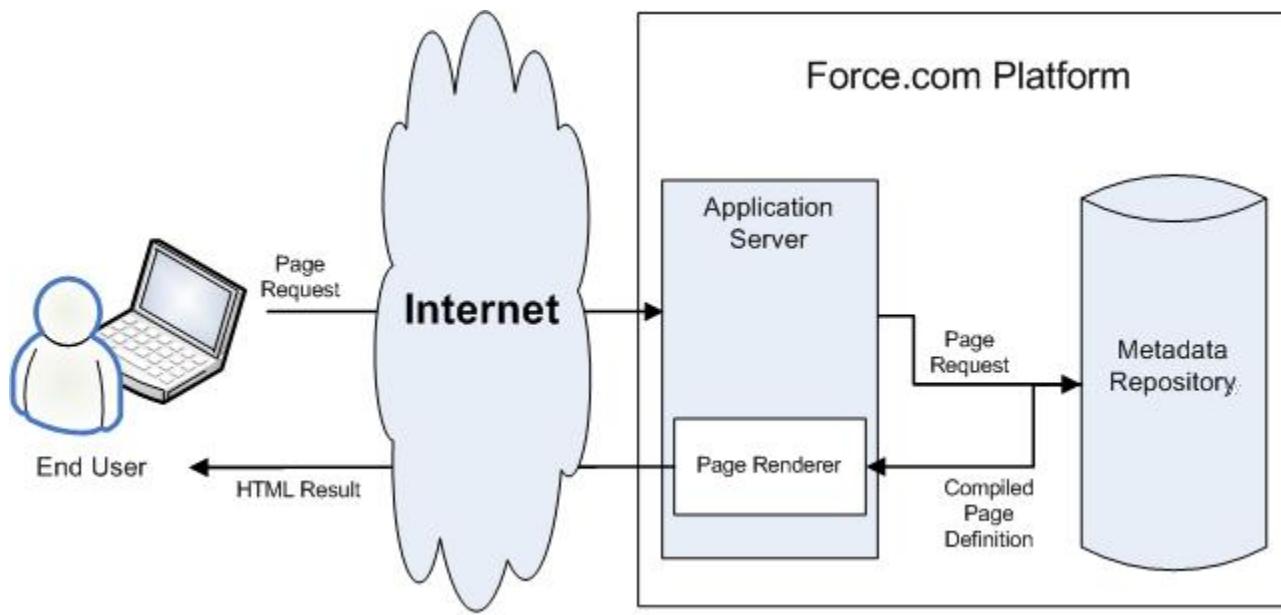


図 3 : Visualforce システムアーキテクチャ - 標準ユーザモード

 メモ: Visualforce ページは salesforce.com サーバではなく force.com サーバの 1 つで実行できます。

関連リンク

[Visualforce とは](#)

[Visualforce の利点](#)

[Visualforce ページと s コントロールの比較](#)

Visualforce の利点

Visualforce には、マークアップ言語として次のような利点があります。

ユーザフレンドリーな開発

開発者は結果ページを表示するウィンドウと同じウィンドウで Visualforce のマークアップを編集できます。そのため、開発者はコードを保存するだけで編集結果を直ちに確認できます。Visualforce エディタペインには、オートコンプリートと構文の強調表示機能もあります。

また、Visualforce では、開発者が即座にサポートコンポーネントを作成できる「クイック修正」もサポートしています。たとえば、開発者は Salesforce にログインして、URL に新しいページの名前を入力するだけで、新しい Visualforce ページを定義できます。wiki とほぼ同様に、ページがまだ存在していない場合は、プラットフォームによって自動的に作成されます。

他の Web ベースのユーザインターフェース技術との統合

Visualforce マークアップは最終的に HTML で表示されるため、設計者は Visualforce タグと、標準 HTML、JavaScript、Flash、またはプラットフォーム上の HTML ページ内で実行できるその他の任意のコード (Force.com プラットフォームの差し込み項目および式など) を併用できます。

Model-View-Controller (MVC) スタイルの開発

Visualforce は、アプリケーションのビュー (Visualforce マークアップで定義されるユーザインターフェース) と、アプリケーションの仕組みを特定するコントローラ (Apex で記述された Visualforce コントローラで定義されるビジネスロジック) の間を明確に分けることで、Model-View-Controller (MVC) 開発パターンに準拠しています。このアーキテクチャでは、設計者および開発者は、新規アプリケーションの作成で行う作業を簡単に分割できます。設計者はユーザインターフェースのデザインに集中し、開発者はアプリケーションを機能させるビジネスロジックに取り組むことができます。

簡潔な構文

Visualforce ページは、約 90% 少ない行数のコードで、S コントロールと同じ機能を実装できます。

データ主導のデフォルト

Visualforce コンポーネントは、プラットフォームによってインテリジェントに表示されます。たとえば、ページ設計者は、さまざまな種別の編集可能な項目 (メールアドレス、カレンダーの日付など) に対して異なるコンポーネントタグを使用する必要はありません。すべての項目に汎用の `u l d p p` タグを使用できます。Visualforce レンダラは各項目の適切な編集インターフェースを表示します。

ホストされているプラットフォーム

Visualforce ページは、Force.com プラットフォームによってその全体がコンパイルおよび表示されます。これらのページは非常に緊密に統合されているため、表示または編集されるデータ量に関係なく、標準の Salesforce ページと同じパフォーマンスを示します。

自動アップグレード可能

Visualforce ページは、Force.com プラットフォームの他の部分がアップグレードされた場合でも書き換える必要はありません。ページはメタデータとして保存されているため、システムの残りの部分と共に自動的にアップグレードされます。

Visualforce を使用する必要がある状況は?

Salesforce では、強力な CRM 機能を提供するアプリケーションが組み込まれています。また、Salesforce では組織に応じて組み込みアプリケーションをカスタマイズする機能も用意されています。ただし、組織には、既存の機能ではサポートされていない複雑なビジネスプロセスがあります。Force.com プラットフォームには、このような場合に高度な管理者や開発者がカスタム機能を実装できるさまざまな方法が搭載されています。そうした方法の中には、Visualforce、Apex、および SOAP API などがあります。

Visualforce

Visualforce では、タグベースのマークアップ言語を使用して、開発者はより効果的にアプリケーションを開発したり、Salesforce のユーザインターフェースをカスタマイズしたりできます。Visualforce を使用して、次のことができます。

- ・ ウィザードやその他のマルチステッププロセスの構築
- ・ アプリケーションを介した独自のカスタムフローコントロールの作成
- ・ 最適かつ効果的なアプリケーションの相互作用を目的とした、ナビゲーションパターンやデータ固有ルールの定義

Apex

次のような場合に Apex を使用します。

- ・ Web サービスを作成する
- ・ メールサービスを作成する
- ・ 複数のオブジェクトに複雑な検証を実行する
- ・ ワークフローでサポートされていない複雑なビジネスプロセスを作成する
- ・ カスタムトランザクションロジック (1 つのレコードやオブジェクトだけでなく、トランザクション全体で発生するロジック) を作成する
- ・ レコードの保存などの別の操作にカスタムロジックを追加し、ユーザインターフェース、Visualforce ページ、SOAP API のいずれでも、その操作が実行されるようにする

詳細は、『[Force.com Apex コード開発者ガイド](#)』を参照してください。

SOAP API

一度に 1 つのレコードタイプのみを処理し、トランザクション制御 (Savepoint の設定や変更のロールバックなど) を必要としない複合アプリケーションに機能を追加する場合、標準の SOAP API コールを使用します。

詳細は、『[SOAP API 開発者ガイド](#)』を参照してください。

Visualforce ページと S コントロールの比較



重要: S コントロールは、Visualforce ページに置き換えられました。組織で以前に S コントロールを使用していない場合は、作成できません。既存の S コントロールに影響はありません。今後も編集できます。

Visualforce ページは、次世代の S コントロールとされ、パフォーマンスの向上および作成の簡略化のために、可能な限り S コントロールの代わりに使用する必要があります。次の表は、Visualforce ページと S コントロールの相違点を説明したものです。

	Visualforce ページ	S コントロール
必要な技術的スキル	HTML、XML	HTML、JavaScript、AJAX ツールキット
言語スタイル	タグマークアップ	手続き型コード
ページ上書きモデル	タグを使用して標準コンポーネント ページ全体の HTML およびおよびカスタムコンポーネントを作成 JavaScript の作成	
標準 Salesforce コンポーネントライ ブラリ	はい	いいえ
組み込みプラットフォームへのアクセス	はい。標準コントローラを使用	いいえ
データ分割	はい	いいえ
	開発者は、テキストボックスなどの入力コンポーネントを特定の項目(取引先名など)で分割することができます	開発者は入力コンポーネントを特定の項目に分割することはできません。代わりに、API を使用する

Visualforce ページ	Sコントロール	
ます。ユーザがその入力コンポーネントに値を保存すると、値はデータベースにも保存されます。	JavaScript コードを記述して、ユーザがデータベースを更新する必要があります。	
スタイルシートの継承	いいえ。Salesforce スタイルシートを手動で指定する必要があります。	
一意性など、項目メタデータの設定	はい(デフォルト) ユーザが一意性または必須性の項目属性に反しているレコードを保存しようとすると、エラーメッセージが自動的に表示され、ユーザは再度試行することができます。	はい。 API コールを使用して JavaScript でコード化する場合。 ユーザが一意性または必須性の項目属性に反しているレコードを保存しようとすると、エラーメッセージは Sコントローラ開発者がこれらの属性をチェックしたコードを記述した場合にのみ表示されます。
Apex との統合	直接。カスタムコントローラにバンド。e ew e b r 間接。API を経由して Apex メソッドを使用。v is	
パフォーマンス	マークアップが Force.com プラットフォーム上で実行されるため、すぐに対応します。	API へのすべてのコールではサーバとの往復が必要になるため、応答速度が低い。付加はパフォーマンスを調整する開発者によって異なる。
ページコンテナ	ネイティブ	iFrame 内

関連リンク

[Visualforce とは?](#)

[Visualforce の利点](#)

[Visualforce のアーキテクチャ設計](#)

Visualforce のバージョン設定方法

Summer '09 リリースから、Visualforce ページおよびコンポーネントはバージョン設定されています。ページまたはコンポーネントにバージョン番号がある場合、古い Visualforce 要素の機能は新しい実装が導入されても変わりません。Visualforce バージョンは 15.0 から開始します。Visualforce ページのバージョンを 15.0 より前のバージョンに設定しようとすると、自動的に 15.0 に変更されます。

下位互換性を持たせるため、各 Visualforce ページおよびカスタムコンポーネントは、特定のバージョンの Visualforce のほか、指定されたバージョンの API のバージョン設定で保存されます。Visualforce ページまたはコンポーネントがインストール済み管理パッケージを参照する場合、このページまたはコンポーネントが参照する各管理パッケージのバージョン設定も同時に保存されます。Visualforce、API、および管理パッケージのコンポーネントが次のバージョンにアップグレードした場合、Visualforce ページおよびコンポーネントは特定の、既知の動作のバージョンにバインドされます。

Visualforce ページで参照されているカスタムコンポーネントは必ずそのバージョン番号で実行されます。つまり、カスタムコンポーネントがバージョン 15.0 で設定されている場合、バージョン 15.0 または 16.0 ページのどちらで実行されているかを問わず、必ず Visualforce バージョン 15.0 の動作をします。

リリースノートには、Visualforce バージョン間での変更を記載しています。また、コンポーネントの参照には、どのVisualforce のバージョンで標準コンポーネントが導入されているのか、また、あるバージョンでコンポーネントまたは属性が廃止になっているかどうかが記載されています。

Visualforce ページまたはカスタムコンポーネントに Salesforce API および Visualforce のバージョンを設定する手順は、次のとおりです。

1. Visualforce ページまたはコンポーネントを編集して、[バージョン設定] をクリックします。



メモ: [設定] の [開発] からページまたはカスタムコンポーネントを編集する場合にのみ、ページまたはカスタムコンポーネントのバージョン設定にアクセスできます。[開発モード] で編集する場合は、バージョン設定にアクセスできません。

2. Salesforce API の [バージョン] を選択します。このバージョンは、ページまたはコンポーネントで使用する Visualforce のバージョンも示します。
3. [保存] をクリックします。

関連リンク

[カスタムコンポーネントのバージョン設定の管理](#)

規則	説明
斜体	<p>構文の記述では、斜体は変数を示します。実際の値を入力してください。次の例では、3つの値を入力する必要があります。<code>datatype variable_name [= value];</code></p> <p>構文で太字かつ斜体のテキストは、クラス名や変数の値など、ユーザが指定する必要があるコード要素を表します。</p> <pre>t t blu plc { c c YourClassHere a ss a ...</pre>
Bold Courier font	<p>コードサンプルと構文の記述では、太字の Courier フォントはコードまたは構文の部分を強調します。</p> <p>< ></p> <p>構文の記述では、不等号 (< >) は表示されたとおりに入力します。</p> <pre>e ex e B e l go pbl p lt u nt thko ua joact tnGra c or! ! ca c e x e f lo u t hnu o & c amc v a! ! e x e f lo u t hnu o tr d ac gw a! ! e x e f lo u t hnu on < } co a c hv a! ! e ex e B l go pbl p c < T k/a a a></pre>
{ }	<p>構文の説明では、中かっこ ({ }) は表示されたとおりに入力します。</p> <pre>e ex e { e ll de g p p < a > a! e ex g p p < mH \$ / a > ! s E s N a!</pre>
[]	<p>構文の記述では、角かっこで囲まれるものはすべて省略可能です。次の例では、<code>value</code> の指定は省略可能です。</p> <pre>datatype variable_name [value] = [;</pre>
	<p>構文の記述では、パイプ記号は「または」を意味します。次のいずれか(すべてではない)を実行できます。次の例では、2つの方法のいずれかを使用して未入力のセットを作成するか、次のようにセットを入力することができます。</p> <pre>e t data_type set_name S > e e wnt data_type = < S > ;() e e wnt { data_type value < value2 >] , [. . . ;</pre>

第2章

Visualforce 開発用ツール

Visualforce ページおよびコンポーネントの開発を始める前に、さまざまな作成環境についてよく理解しておいてください。

- Visualforce を作成する最適な方法は、Visualforce 開発モードを有効にすることです。Visualforce 開発モードは、「アプリケーションのカスタマイズ」権限を持つユーザのみが使用できます。開発モードには、次の機能があります。
 - ◊ 各 Visualforce ページの特殊な開発フッター。このフッターには、ページのビューステート、関連コレクション、コンポーネントの参照ドキュメントへのリンク、およびコンポーネントタグと属性名の強調表示、検索置換機能、自動候補を提供するページマークアップエディタが含まれます。
 - ◊ 一意の URL を入力するだけで新規 Visualforce ページを定義できる機能。
 - ◊ 標準ユーザが受け取るスタッカ追跡よりも詳細な情報が含まれたエラーメッセージ。
- Visualforce 開発モードを有効化する手順は、次のとおりです。
 1. Salesforce ページ上部で、名前の横にある下向き矢印をクリックします。名前の下にあるメニューで、[設定] または [私の設定] のどちらか表示されるほうを選択します。
 2. 左ペインで、次のいずれかを選択します。
 - ◊ [設定] をクリックした場合は、[私の個人情報] > [個人情報] を選択します。
 - ◊ [私の設定] をクリックした場合は、[個人用] > [高度なユーザの詳細] を選択します。
 3. [編集] をクリックします。
 4. [開発モード] チェックボックスをオンにします。
 5. 必要に応じて、[開発モードでビューステートを表示] チェックボックスをオンにして開発フッターの[ビューステート] タブを有効にします。このタブは、Visualforce ページのパフォーマンス監視に役立ちます。
 6. [保存] をクリックします。
- [設定] で [開発] > [ページ] をクリックすると、Salesforce ユーザインターフェースで Visualforce ページを開発することもできます。Visualforce コンポーネントの場合は、[設定] で [開発] > [コンポーネント] をクリックします。
- Force.com IDE は、他にはない機能を備えた Eclipse IDE のプラグインです。Force.com IDE には、Force.com アプリケーションを構築およびリリースするための統合インターフェースがあり、ソースコードエディタ、プロジェクトウィザード、統合ヘルプなどのツールが用意されています。この IDE は、上級開発者および開発チーム向けに設計されています。

開発モードフッターの使用

開発モードを有効にすると、そのページの URL に移動してページコンテンツの表示と編集ができます。たとえば、ページの名前が `http://salesforce.com/instances/roofrcc` である場合、ブラウザのアドレスバーに `http://salesforce.com/instances/roofrcc/ord&psa/sa` と入力します。開発モードでは、特殊な開発フッターを使用して、Visualforce ページとカスタムコントローラの編集や、Visualforce のパフォーマンス監視もできます。

開発モードを有効にすると、すべての Visualforce ページで、ブラウザの下部に開発モードフッターが表示されます。

- ・ ページの名前のタブをクリックしてページエディタを開くと、[設定] 領域に戻らずに、関連付けられた Visualforce マークアップの表示と編集が行えます。変更は、ページを保存するとすぐに表示されます。
- ・ カスタムコントローラがページに使用されている場合、コントローラクラスの名前はタブとして提供されます。そのタブをクリックすると、関連付けられた Apex クラスを編集できます。
- ・ ページでいずれかのコントローラ拡張を使用する場合、拡張それぞれの名前がタブとして表示されます。そのタブをクリックすると、関連する Apex クラスを編集できます。
- ・ [設定] で有効にすると、[View State (ビューステート)] タブには、Visualforce ページのビューステートに影響する項目に関する情報が表示されます。
- ・ [Save(保存)] (編集ペインのすぐ上) をクリックすると、変更を保存し、ページのコンテンツを更新できます。
- ・ [Component Reference (コンポーネントの参照)] をクリックすると、サポートされているすべての Visualforce コンポーネントのドキュメントを表示できます。
- ・ [使用場所] をクリックすると、カスタムタブ、コントローラ、またはその他のページなど、そのページを参照する Salesforce の全項目の一覧が表示されます。
- ・ 開発モードフッターパネルを折りたたむには、縮小ボタン (□) をクリックします。展開するには、展開ボタン (■) をクリックします。
- ・ 開発モードを完全にオフにするには、[Disable Development Mode (開発モードを無効化)] ボタン (✖) をクリックします。開発モードは、個人設定の個人情報ページで再度有効化されるまで、オフのままになります。

[View State (ビューステート)] タブについて

Web ページのビューステートは、サーバ要求時 (データの送受信など) にコントローラの状態を維持するために必要なすべてのデータで構成されます。ビューステートは、ページの合計サイズに含まれるため、ページのパフォーマンスは、ビューステートを効率よく管理するかどうかによって変わることがあります。開発モードフッターの [View State (ビューステート)] タブでは、Salesforce とやり取りするときの Visualforce ページの状態に関する情報を提供します。



メモ: [View State (ビューステート)] タブは、ページ要求プロセスを理解している開発者が使用する必要があります。このタブを使用する前に、Visualforce ページでの実行の順序についてよく理解してください。

[View State (ビューステート)] タブを有効にする手順は、次のとおりです。

1. Salesforce ページ上部で、名前の横にある下向き矢印をクリックします。名前の下にあるメニューで、[設定] または [私の設定] のどちらか表示されるほうを選択します。
2. 左ペインで、次のいずれかを選択します。
 - ・ [設定] をクリックした場合は、[私の個人情報] > [個人情報] を選択します。

- [私の設定] をクリックした場合は、[個人用] > [高度なユーザの詳細] を選択します。
- [編集] をクリックします。
- [開発モード] チェックボックスがオフの場合は、オンにします。
- [開発モードでビューステートを表示] チェックボックスをオンにします。
- [Save (保存)] をクリックします。



メモ: ビューステートはフォームデータにリンクしているため、[View State (ビューステート)] タブは、ページに `m p f` タグが含まれる場合にのみ表示されます。また、[View State (ビューステート)] タブは、カスタムコントローラまたはコントローラ拡張を使用するページにのみ表示されます。

[View State (ビューステート)] タブは、フォルダノードで構成されます。いずれかのフォルダをクリックすると、[コンテンツ] タブのある円グラフが表示されます。このグラフには、フォルダの子 Visualforce カスタムコントローラ、Apex オブジェクト、または項目が表示されます。グラフの各片の上にマウスポインタを重ねると、親の合計サイズを占める各要素を表示できます。これは、個々のテキストノードと同じ情報です。グラフを表示するには、ブラウザで Flash バージョン 6 以降が有効である必要があります。

Salesforce で許容される Visualforce ページの最大ビューステートサイズは 135KB です。[View State (ビューステート)] タブには、ページのどの要素がその領域を占めているかが表示されます。一般に、ビューステートサイズが小さいほど読み込み時間が短くなります。ページのビューステートを最小限にするには、Apex コントローラコードを最適化し、使用されている不要な Visualforce コンポーネントを削除します。次に例を示します。

- ビューステートの大部分をコントローラまたはコントローラ拡張で使用されているオブジェクトから取得していることが分かった場合は、Visualforce ページに関連するデータのみを戻すように SOQL コールの絞り込みを検討する。
- ビューステートが大規模なコンポーネントツリーの影響を受けている場合は、ページが依存しているコンポーネント数の削減を試みる。

[View State (ビューステート)] タブを使用した Visualforce の改善方法の詳細は、「[Visualforce のパフォーマンス向上のためのベストプラクティス](#)」(ページ 276)を参照してください。

[View State (ビューステート)] タブには、次の列が含まれます(アルファベット順)。

列	説明
<code>f</code>	親の割合 % Pa () カスタムコントローラ、Apex オブジェクト、または項目が親の合計サイズに占める割合。
<code>N</code>	名前 Na () カスタムコントローラ、Apex オブジェクト、または項目の名前。
<code>i</code>	サイズ Si S () カスタムコントローラ、Apex オブジェクト、または項目のビューステートサイズ。
<code>y</code>	種別 Yi () カスタムコントローラ、Apex オブジェクト、または項目の種別。
<code>V</code>	値 Va () 項目の値。

[名前] 列には、Visualforce ページのさまざまな部分を定義するノードが含まれます。次のようなノードがあります(アルファベット順)。

ノード	説明
コンポーネントツリー 内部 数式 状態 ピューステート	<p>ページの全体構造を表します。サイズは、ページに配置するコンポーネントの数によって影響されます。一般に、コンポーネントの数が少ないほど、コンポーネントツリーが小さくなり、読み込み時間が短くなります。ピューステートサイズのうち、コンポーネントツリーがどの程度を占めているかを表示するには、ピューステート] フォルダをクリックします。</p> <p>Visualforce ページによって使用される内部 Salesforce データを表します。開発者はこのノードを制御できません。ピューステートサイズのうち、内部要素がどの程度を占めているかを表示するには、状態] フォルダをクリックします。</p> <p>Visualforce ページで定義される数式によって使用されるデータを表します。</p> <p>このフォルダには、Visualforce カスタムコントローラ、Apex オブジェクト、または項目がすべて含まれます。子の Controller フォルダおよび Controller Extension フォルダを展開すると、ページ上の各オブジェクト、その項目、項目の値を表示できます。一般に、これらは Apex コントローラロジックによって異なります。</p> <p>このフォルダにはすべてのノードが含まれます。クリックすると、Visualforce ページのピューステートに関する全体的な情報が表示されます。[Capacity(容量)] タブには、割り当てられたピューステートサイズのうち、使用中のサイズが表示されます。割り当て量を超えると、グラフにも超過したサイズ(キロバイト単位)が表示されます。</p>

Visualforce エディタについて

開発モードフッターまたは[設定]から Visualforce ページを編集する場合、次の機能を持つエディタを使用できます。

構文の強調表示

エディタは、キーワードとすべての関数および演算子について、自動的に構文を強調表示します。

検索

検索により、現在のページ、クラス、またはトリガの中のテキストを検索できます。検索を使用するには、**検索**] テキストボックスに文字列を入力し、[Find Next(次を検索)] をクリックします。

- 検出した検索文字列を他の文字列で置き換えるには、**置換**] テキストボックスに新しい文字列を入力し、そのインスタンスだけを置き換える場合は [replace] をクリックし、そのインスタンスと、それ以外にそのページ、クラス、またはトリガに出現する検索文字列のすべてのインスタンスを置き換える場合は、[Replace All(すべて置換)] をクリックします。

- 検索操作で大文字と小文字を区別するには、[Match Case (大文字と小文字を区別する)] オプションをオンにします。

- 検索文字列として正規表現を使用するには、[Regular Expressions (正規表現)] オプションをオンにします。正規表現は、JavaScript の正規表現規則に従います。正規表現を使った検索では、折り返されて複数行になる文字列も検索できます。

正規表現で検出した文字列を置換操作で使用する場合、検出した検索文字列から得られる正規表現のグループ変数（`\1`、`\2`など）をバインドすることもできます。たとえば、`< h 1 タグを h 2 タグで置き換 < />` を検索し、それを `h 1 > 2 で置 $ $ < 1` と置き換えます。

指定行に移動 (→)

このボタンにより、指定した行番号を強調表示できます。その行が現在表示されていない場合は、エディタがその行までスクロールします。

元に戻す (⬅) およびやり直し (➡)

[Undo (元に戻す)] を使用すると編集動作を取り消します。[Redo (やり直し)] を使用すると元に戻した編集動作をやり直します。

フォントサイズ

ドロップダウンリストからフォントサイズを選択し、エディタに表示される文字のサイズを制御します。

行と列の位置

カーソルの行と列の位置は、エディタ下部のステータスバーに表示されます。これは、[Go To Line (指定行に移動)] (→) と共に使用し、エディタ内をすばやく移動できます。

行と文字の計数

行と文字の合計数は、エディタ下部のステータスバーに表示されます。

エディタは、次のキーボードショートカットをサポートします。

Tab

カーソル位置にタブを追加する

SHIFT+Tab

タブを削除する

CTRL+f

[検索] ダイアログを開く、または、現在の検索の次の検索値を検索する

CTRL+r

[検索] ダイアログを開く、または、現在の検索の次の検索値を指定した置換文字列に置き換える

CTRL+g

[Go To Line (指定行に移動)] ダイアログを開く

CTRL+s

適用の実行

CTRL+Z

最後の編集操作を取り消します。

CTRL+Y

元に戻した最後の編集操作をやり直します。

第3章

Visualforce のクイックスタート

Visualforce の重要な要素を紹介するために、この章には言語の機能を説明する例のセットが含まれています。これらの例は、すべてのタグやコントローラのあらゆる詳細、ルール、または例外を網羅するものではありませんが、Visualforce の新しい開発者は、このガイドの残りの部分に記載される詳細な説明に進む前に、このチュートリアルで、Visualforce がどのように機能するかを理解することができます。

これらの例は、初級と上級のセクションに分けられています。初級の例では、主に Visualforce マークアップを使用します。上級の例では、Visualforce マークアップのほかに、Force.com Apex コードを使用します。

Apex を必要とする [高度な例](#) は、独立した章で説明されています。

Visualforce の正常なコンパイル

Visualforce ページとコンポーネントは、正しくコンパイルされない限り保存できません。次は、Visualforce ページを作成するときの注意事項のリストです。

- e • コンポーネントのタグが `<apex:component>`(の後にコロン) のような正しい名前空間の識別子で始まっていることを確認します。
- すべての開始引用符と開始括弧に対する終了引用符と終了括弧があることを確認します。
- コントローラまたはコントローラ拡張の名前が正しく付けられていることを確認します。
- Salesforce API バージョン 19.0 以降を使用して作成された Visualforce ページとコンポーネントは、適切なフォームの XML で記述されている必要があります。一般的に、要素が正しくネストされている必要があること、空でない要素に終了タグが必要であること、空の要素は閉じスラッシュ()で終了している必要があること、などを意味します。World Wide Web コンソーシアム (W3C) では、[適切なフォームの XML の仕様に関する記事](#)を提供しています。

次の例外が許可されます。

- ◊ 適切なフォームの XML に違反するコードは JavaScript 内では許可されます。たとえば、Visualforce で `<apex:actionFunction name="myAction" />` タグを使用する必要はありません。
 - ◊ 適切なフォームの XML に違反するコードは式内では許可されます。たとえば、数式では引用符をエスケープする必要はありません。
- 例 `len($on) > do < gc v? $1 = "T" など、ページの開始に通常必要とされる XML ディレクティブは、e.g. <apex:page> や <apex:form> などの最上位のコンテナタグ内で使用できます。`

最初のページの作成

開発モードが有効な場合、ブラウザのアドレスバーに次のようなページの URL を入力することによって、最初の Visualforce ページを作成できます。

```
http://mySalesforceInstance.salesforce.com/a/
```

たとえば、「HelloWorld」というページを作成するときに、組織が使用している場合、「HelloWorld」と入力します。

ページはまだ存在しないため、中間ページが表示され、そこから新規ページを作成できるようになっています。[Create Page <myNewPageName> (ページ *nameOfNewPage* を作成)] をクリックして、自動的に作成します。



メモ: Visualforce 開発モードが有効でない場合は、[設定] で [開発] > [ページ] をクリックしてから [新規] をクリックし、新しいページを作成できます。

Visualforce ページは、セットアップのこの部分以降で常に編集できるようになりますが、編集の結果を確認するには、ページの URL に移動する必要があります。そのため、ほとんどの開発者は、開発モードを有効にして作業することによって単一のウィンドウでページの参照と編集を行います。

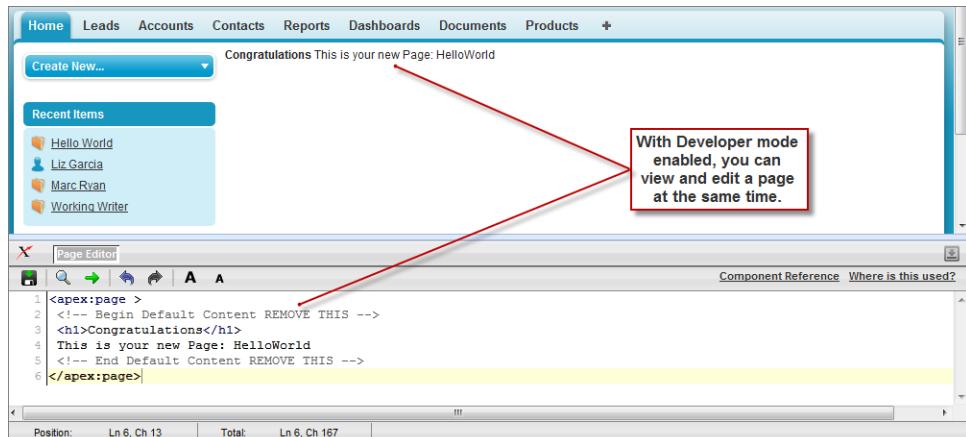


図4: 新しいVisualforce ページ

デフォルトのテキストを含む Visualforce ページが表示されました。新しいページを編集するには、ブラウザの下に表示されている[ページエディタ]のバーをクリックします。ページエディタが展開されて、次の Visualforce マークアップが表示されます。

```
<apex:page>
<!-- Begin Default Content REMOVE THIS -->
<h1>Congratulations</h1>
This is your new Page: HelloWorld
<!-- End Default Content REMOVE THIS -->
</apex:page>
```

このデフォルトのマークアップには、すべてのページに必要な `<apex:page>` タグ (ページのマークアップを開始/終了するタグ)のみが含まれます。`<apex:page>` の開始タグと終了タグに組み込まれているのはブレーンテキストで、一部は `<h1> Congratulations </h1>` 標準 HTML タグでフォーマットされています。

必須の `g p p` タグを維持するかぎり、プレーンテキストや有効な HTML をこのページに必要なだけ追加できます。たとえば、ページエディタに次のコードを入力して [Save(保存)] をクリックすると、ページに「Hello World!」という太字のテキストが表示されます。

```
11 < H W > < ! >/
```



ヒント: 警告には注意してください。開始タグに一致する終了タグがない HTML が含まれているページを保存すると、Visualforce エディタに警告が表示されます。ページは保存されますが、不正な形式の HTML は、表示されるページに問題を引き起こす原因となります。

Visualforce による項目値の表示

Visualforce ページでは、数式と同じ式の言語を使用します。つまり、`m$` 内のすべてが、現在コンテキストにあるレコードから得られる値にアクセスできる式として評価されます。たとえば、`m$` 式をページに追加すると、現在のユーザの「名」を表示できます。

```
11 < m$ U ! s E s N a! >
```

は、現在のユーザレコードを常に表すグローバル変数です。すべてのグローバル変数は、`$` 記号で参照されます。Visualforce で使用できるグローバル変数のリストについては、「[グローバル変数](#)」(ページ 541)を参照してください。

特定の取引先、取引先責任者、またはカスタムオブジェクトレコードなど、グローバルに使用できるようになつていないレコードの項目にアクセスするには、ページをコントローラに関連付ける必要があります。コントローラを使用すると、特定のオブジェクトのレコードにアクセスする方法を指定するロジックなど、アプリケーションを実行するためのデータやビジネスロジックをページで使用できます。ページのカスタムコントローラは Apex を使用して定義できますが、Salesforce には、すべての標準およびカスタムオブジェクトに使用できる標準コントローラが含まれています。

たとえば、取引先に対して標準コントローラを使用するには、`g p e pd` タグに `or l d r aC` 属性を追加し、取引先オブジェクトの名前を割り当てます。

```
11 < m$ U ! s E s N a! >
```

ページを保存したら、ページの [取引先] タブが強調表示され、ページのコンポーネントのデザインが [取引先] タブに適用されます。さらに、`c<fieldName>` 式の構文を使用して、現在コンテキストにある取引先レコードの項目にアクセスできるようになります。

たとえば、取引先の名前をページに表示するには、ページのマウスアップで `cc m` を使用します。

```
11 < m$ U ! s E s N a! >
```

式は、標準取引先コントローラの `getRecord()` メソッドにコールし、現在コンテキストにある取引先のレコード ID を返します。その後、ドット表記を使用してそのレコードの `項目名` 項目にアクセスします。



メモ: この式の言語を使用して親オブジェクトにアクセスすることはできません。言い換えると、`!isError()` はエラーを返します。



メモ: ページを保存すると、`inputText` など、すべての入力コンポーネントの `label` 属性について、文字テキストや空白を含まない单一式であり、単一のコントローラメソッドまたはオブジェクトプロパティへの有効な参照であるかどうかが検証されます。エラーが発生するとページを保存できません。

取引先レコードを現在のコンテキストに取り込むには、レコードの ID を指定するページ URL にクエリパラメータを追加する必要があります。手順は、次のとおりです。

- 任意の方法で、取引先の ID を検索します。そのための簡単な方法として、取引先レコードの詳細ページを表示し、URL の最後にある文字コードをコピーするやり方があります。たとえば、次の URL で取引先詳細ページに移動するとします。

この場合、`id` が取引先の ID になります。⁵

- ページに戻り、ブラウザのアドレスバーの URL にクエリ文字列パラメータとして取引先 ID を追加します。たとえば、ページが次の場所にあったとします。

URL の最後に `id` を追加します。^{000B003 5}



メモ: URL に `instance` パラメータを使用する場合、そのパラメータは、標準コントローラで参照されるエンティティと同じエンティティを参照する必要があります。

取引先 ID を URL に指定したら、次の図のようにページに適切な取引先名が表示されます。

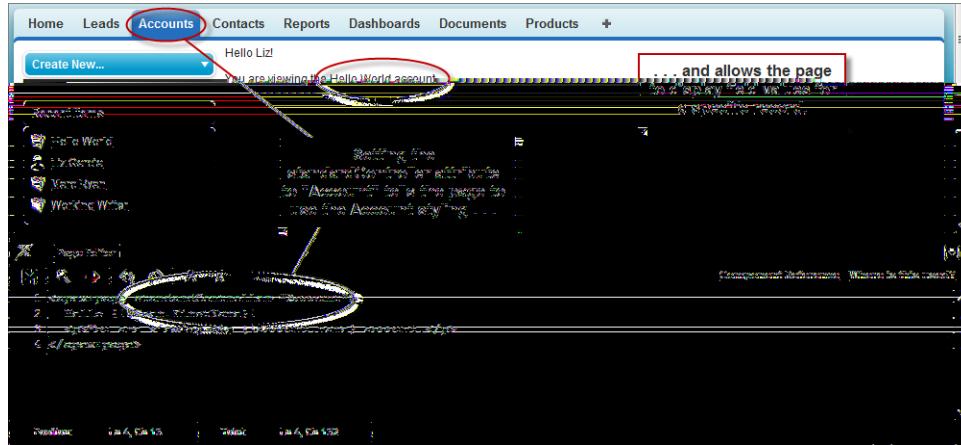


図 5: Visualforce ページの取引先データの表示

Visualforce コンポーネントライブラリの使用

ここまでで、例に使用された唯一の Visualforce タグは、すべての Visualforce マークアップの先頭と末尾に配置する必要がある必須の `g p p` タグです。ただし、`ol` または `<` タグを使用して HTML ドキュメントに画像やテーブルを挿入できるのと同様に、Visualforce コンポーネントライブラリに定義されたタグを使用して、Visualforce ページにユーザインターフェースコンポーネントを追加できます。

たとえば、詳細ページでセクションのように見えるコンポーネントを追加するには、`l go p p c` コンポーネントタグを使用します。

```
t n tegndprd por hñ C au acc a a " " "
x teBe l go p d p cll <de kra r}a i mH $!" U ! s E s N a!! "
ex e return t egn n dnj tY nc m a oin cc ! a . a a .
B l go p p c < k/a a >
g p p < /a a >
```

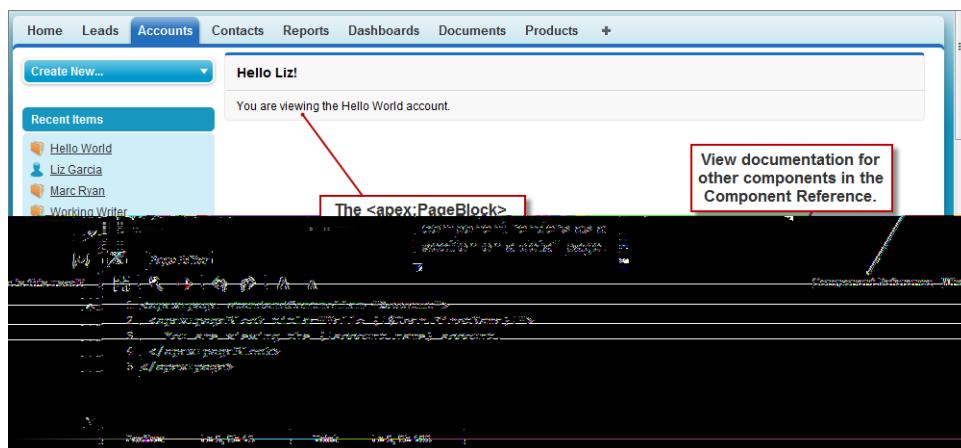


図 6:<apex:pageBlock> コンポーネント

タグは、関連リスト、詳細ページ、および入力項目などの、一般的な Salesforce インターフェースコンポーネント用にも存在します。たとえば、詳細ページのコンテンツを追加するには、`<apex:component>` コンポーネントタグを使用します。

```
t n tegndprd por id AnC au acc a a = " " >
x teBelgo p d p cll <de kra r}a i mH $! U ! s s N a! >
e e w@ttn tegrn n du tY ac maoi cc ! a . a a .
ex B l go p p c < k/a a >
x d l p < a i a > /
ext g p p < /a a >
```

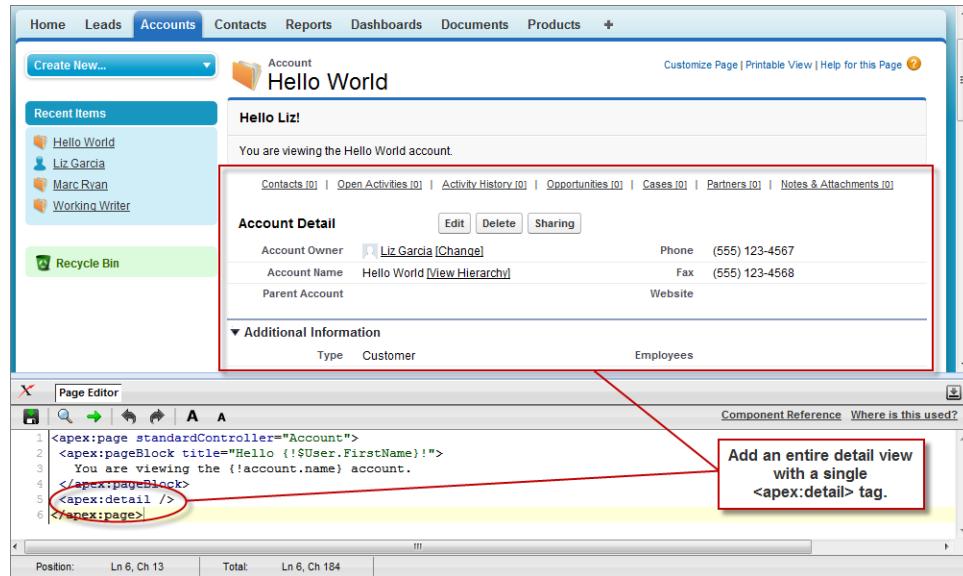


図 7: 属性がない `<apex:detail>` コンポーネント

タグに特定の属性が何も指定されていなくても、`<apex:component>` は、コンテキストレコードの完全な詳細ビューを表示します。どのレコードの詳細を表示するか、関連リストやタイトルを表示するかなど、プロパティを変更するにはタグで属性を使用できます。たとえば、次のマークアップは、コンテキストの取引先所有者の詳細を表示しますが、関連リストや色付きのタイトルバーは表示しません。

```
t n tegndprd por id AnC au acc a a = " " >
x teBelgo p d p cll <de kra r}a i mH $! U ! s s N a! >
e e w@ttn tegrn n du tY ac maoi cc ! a . a a .
ex B l go p p c < k/a a >
x d l p < a i a > /
ext g p p < /a a >
```

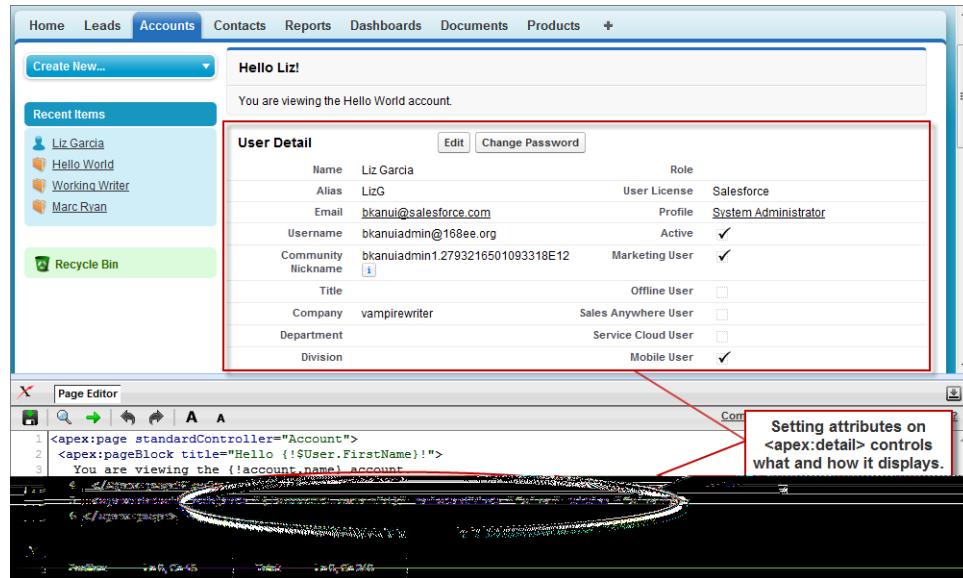


図 8: 関連リストまたはタイトル要素のない<apex:detail> コンポーネント

コンポーネントライブラリを参照するには、ページエディタで [Component Reference (コンポーネントの参照)] をクリックします。このページから、任意のコンポーネントにドリルダウンして、定義した任意のカスタムコンポーネントを含め、各コンポーネントで使用可能な属性を参照できます。

関連リンク

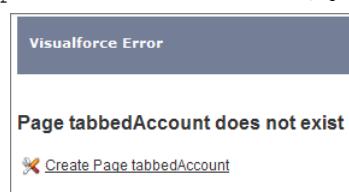
[標準のコンポーネントの参照](#)

Visualforce ページによる既存のページの上書き

標準の取引先詳細ページなど、既存のページ形式を変更するとします。取引先のすべての情報は単一ページに表示されます。多くの情報がある場合は、延々とスクロールする必要があります。Visualforce ページを使用すると、取引先責任者、商談など、取引先の各セクションをタブに表示できます。

最初に、クイック修正を使用して新しい Visualforce ページを作成します。

1. ブラウザで、テキスト「tabbedAccount」を Salesforce インスタンスの URL に追加します。たとえば、Salesforce インスタンスが「https://prod-001.fmc.cs1.salesforce.com」の場合、新しい URL は「https://prod-001.fmc.cs1.salesforce.com/tabbedAccount」になります。次のエラーメッセージが表示されます。



2. 新規ページを作成するには、[Create Page tabbedAccount (ページ tabbedAccount を作成)] をクリックします。
3. ページの左下のページエディタリンクをクリックします。これにより、新しいページのコードが次のように表示されます。

```

g p p < a a
e B t e t g tn lm o f < C ! --> EM E I a T V R G S >-

```

```

tn t or n̄gu C o <1 h > ai <ās >
e ew d̄u T tg h̄ond i s̄is ḡc Pa a
n t edt tn l̄mE o f < ! D̄EM E I a T V R H S>-
g p p < /a > a

```

4. 既存のコードを次のコードで置き換えて、[Save(保存)] をクリックします。

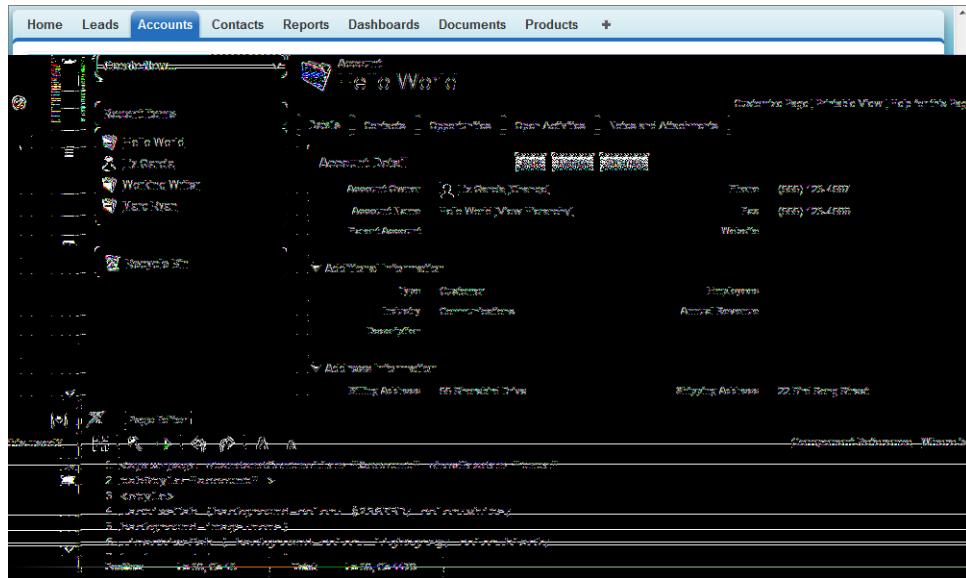
```

t n tegndprd por t̄k KneC ew āu et āco dar ār u ='' " " a'' "
et b t l n ou cc a Sy=" a " >
l < s y>
et { b ba r ōl Tdoro B̄ekat : l̄ato a c :h # B F ; 6 i ;
et n { b dn r ōdT dorā l v̄ckarg :g ahorobā c -: d ;ky ;a
et n { b dn r ōdT dorā l v̄ckarg :g ahorobā c -: d ;ky ;a
l < s / y>
te n w ke lep te ene k̄dh ād̄p tal P̄bsi b̄d y=" i T " s ='' a ai sa
et n b̄u b ccl et b l T i =" C b c a Pa T ass=a v̄'a a '
te e tb l b̄d p n el t < A m āe lta b̄da" b̄d̄ sal =D i s̄ai ='' ai sa >
e ext e e td t r d p d t et a fir au L ās =" s̄ai ='' > /
te b p < /a > a
te b ltb lep a t t̄c cmte ta d̄ha" b ōsa Ḡa c s̄ai ='' a " a >
e ext t t̄el { d p but jn< t̄out h̄cc ás os e' ! c a " i s ='' sa > /
te b p < /a > a
te b t̄k et m ero m < t p p ta nm āo ā O ppi i s' ='' a O i i s'
e ext t t̄el { d p but jn< } oua Lcc ás s ='' ! a "
te b p < /a > a
te b l b̄nle p A e m < p d a et Ma ā O v̄ci i s' ='' a O v̄ i i s'
te d n b A p i =" Oa " >
te t̄el { d p but jn< } oua Lcc ás s ='' ! a "
te b p < /a > a
te e tb l b l pteo Ad n a m dn ā N s a a s'
e ext t t̄ne mAo Ad n t e tm t̄dh b A N s a s' i ='' Na " >
te t̄el { d p but jn< } oua Lcc ás s ='' ! a "
te t e t l ttne Ao Ad n m dn i s ='' N s a s' > /
te b l p < /a > a
te g p p < /a > a Pa>

```

5. [取引先] ページにはデータがありません。前のページと同様に、URL に特定の取引先の ID を指定する必要

ttがあります。たとえば、`https://salesforce_instance_name//page/dtcc /a /a I` の 1 頁 00
のように指定します。取引先 ID を追加した後、ページは次のように表示されます。



このページマークアップについては、次の点に留意してください。

- は、実際には Visualforce マークアップではなく CSS マークアップの一部です。このマークアップでは、activeTab と inactiveTab という 2 種類のタブのスタイルが定義されます。
- が、タブの生成に使用されます。次の属性の使用方法を確認してください。
 - 属性: 有効になっているタブの表示に使用されるスタイルクラスを指定します。
 - 属性: 無効になっているタブの表示に使用されるスタイルクラスを指定します。
- タブパネルの定義内に、それぞれの子のタブコンポーネントの定義があります。最初のタブはタグを使用してページの詳細ビューの該当部分を返します。

```
exte e tb l b d p n el t <A m ae lta dd a" bd sal =@ i s ai =" ai sa >
exte e td t r d p d t e t < /a fir au L a s =" s " a i =" " > /
exte b p < /a >a
```

残りのタブは `<tab>` を使用して取引先ページの異なる部分を指定します。取引先責任者のタブは次のようにになります。タブでは、取引先責任者の既存のリストが使用されます。

```
exte b ltb le p a t tC cmte ta dd a" b o sa @ a c s ai =" a " a > /
exte t rel {d p but j n< t ou t hcc a s os e! c a " i s =" a sa > /
```

これで取引先をタブで表示するページが作成できたため、このページを使用してすべての取引先の詳細ビューを上書きできるようになりました。

- [設定] から、[カスタマイズ] > [取引先] > [ボタン、リンク、およびアクション] をクリックします。
- [ビュー] の横にある [編集] をクリックします。
- 上書き手段 で、[Visualforce ページ] を選択します。
- [Visualforce ページ] ドロップダウンリストで、tabbedAccount を選択します。
- [保存] をクリックします。

[取引先] タブをクリックし、任意の取引先を選択します。取引先の詳細がタブに表示されます。

標準オブジェクトリストページへのリダイレクト

ユーザを標準タブに移動するボタンやリンクの場合、コンテンツをリダイレクトして標準オブジェクトのリストを表示できます。

次のマークアップで Visualforce ページを作成します。

```
t { n g p t p c An< t And a e oau da uCL !BR F t A i ) cau $L p a s , o y . " )> /
```

ユーザには、次のようなページが表示されます。

Action	Account Name	Account Site	Billing State/Province	Phone
Edit Del	University of Almania	AZ	(520) 773-9050	
Edit Del	United Oil & Gas, UK	UK	+44 191 4956203	
Edit Del	United Oil & Gas, Sing...	Singapore	(650) 450-8810	
Edit Del	United Oil & Gas Corp.	NY	(212) 842-5500	

図 9: 取引先詳細ページの上書き

Visualforce ページでは、標準オブジェクトへの参照を変更して、取引先責任者など、他の標準オブジェクトも参照できます。次に例を示します。

```
t { n g p t p c A&lt; tn da e ca da uCL !BR F t ( i )oc . $L p a sc , o y . a" )> /
```

ページでの入力コンポーネントの使用

ここまででは、このクイックスタートチュートリアルの例で、Visualforce ページにデータを表示できる方法を説きました。ユーザの入力を取得するには、1つ以上の入力コンポーネントを含む `<form>` タグと、`<input>` タグまたは `<apex:input>` タグを使用します。

フォームでもっとも使用される入力コンポーネントタグは `<input type="text">` です。このタグは、標準またはカスタムオブジェクト項目の種別に基づいて適切な入力ウィジェットを表示します。たとえば、データ項目を表示するために `<input type="text">` タグを使用すると、カレンダーウィジェットがフォーム上に表示されます。選択リスト項目を表示するために `<input type="select">` タグを使用すると、代わりにドロップダウンリストが表示されます。`<input type="checkbox">` タグは、任意の標準またはカスタムオブジェクト項目のユーザ入力を取得するために使用できます。このタグは、項目が必須または一意であるか、あるいは現在のユーザに表示または編集の権限があるなど、項目の定義に設定されているすべてのメタデータを順守します。

たとえば、次のページでは、ユーザは取引先の名前の編集と保存を行えます。

 メモ: このページに取引先データを表示するには、有効な取引先レコードの ID をページの URL のエンドパラメータとして指定する必要があります。次に例を示します。

```
h e : p Salesforce_instance m xp x bJ / a /y Real= 00 008 s
```

レコードの ID の取得についての詳細は、[Visualforce での項目値の表示](#) (ページ 20) を参照してください。

```

t      n t e gndprd por hñ n C   au   acc   a a      " " >
x      ro m p f   < a >
ex     teB e l g o p d p cll <æ k r a i mH$! U ! s   s N   a! >
e     e   w e t u n t e g n n dñ tY nc m á o i i c c ! a p. a a < . > /
e     e   n t e a n g k u c c m :   ñ a   N < a > / 
e     x   et n e { u l d p l t e p n n< o ñ a c c m v a! ! p a . " a < > / > /
e     x   tt oBndmm w e n o   ñ ñ e a l w u a t e a n i a = b u k c b m a! v s ñ
ex    B l g o p p c < k/a a >
x     ro m p f   < /a >
g p p   < /a >

```

この例では、次の点に留意してください。

- u l d p e p** タグは、タグの **值** 属性を設定することによって、取引先の 項目にバインドされます。式には、ページのほかの場所に項目の値を表示するためによく使用される cc m ドット ! 表記が含まれます。
- oBndmm w e ot** タグには、**c a** 属性があります。この属性の値は、標準取引先コントローラの アクションを呼び出します。このアクションは、標準の取引先編集ページの [保存] ボタンと同じ動作を 実行します。



メモ: ページを保存すると、**u xcept n t p < u pi p** など、すべての入力コンポーネントの **值** 属性について、文字テキストや空白を含まない单一式であり、単一のコントローラメソッドまたはオブジェクトプロパティへの有効な参照であるかどうかが検証されます。エラーが発生すると ページを保存できません。

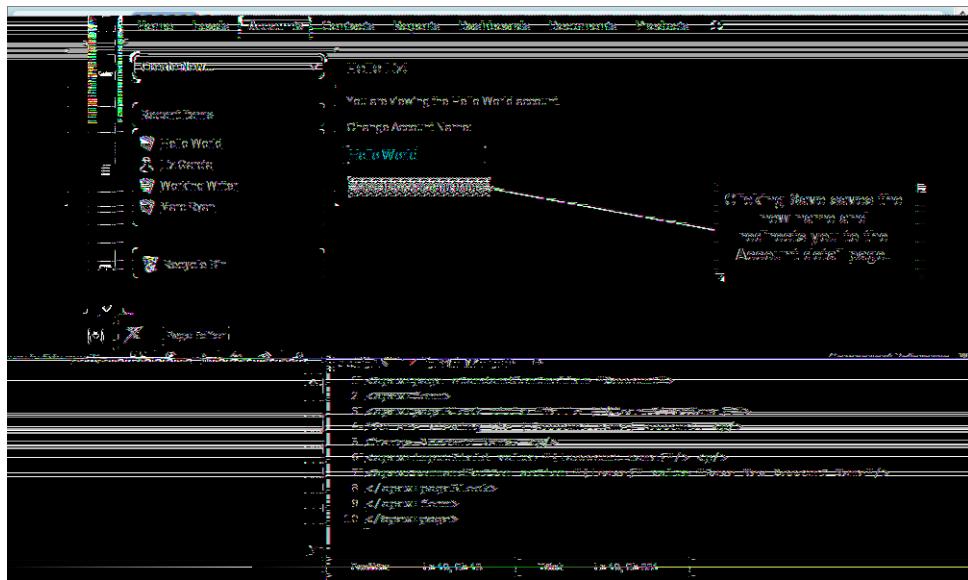


図 10: 単一の入力項目を使用する `<apex:form>` コンポーネント

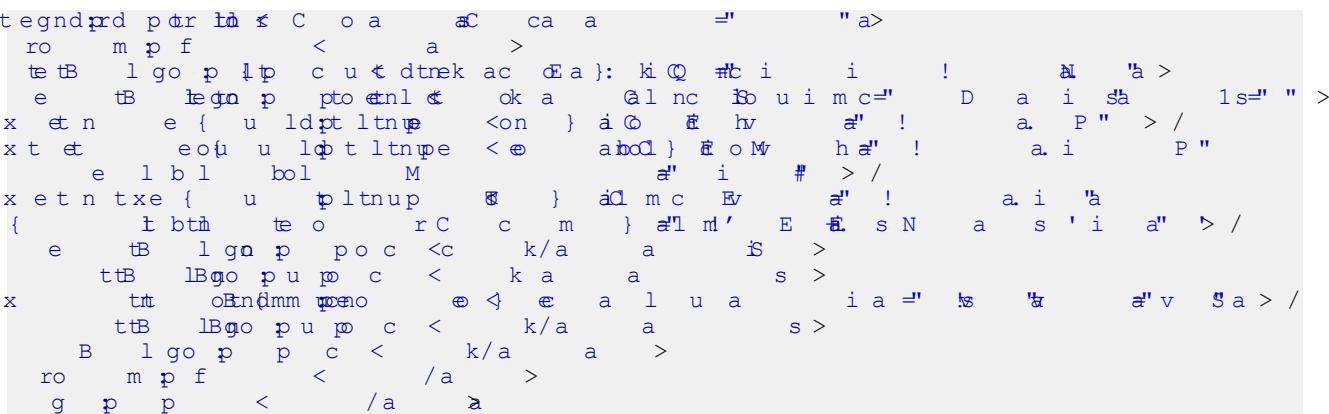
u l d p p タグが表示できない唯一の項目は、Apex で記述されたカスタムコントローラクラスのメンバ変数として定義された項目です。それらの変数で使用するデータを収集するには、代わりに

u p o t e p c <en u l d p t e p < u xcept n t p < u p p s、または **a**

u p p タグを使用します。

入力項目の表示ラベルの追加とカスタマイズ

Visualforce 入力コンポーネントおよびいくつかの出力コンポーネントは、`label` 属性をもつコンポーネント内で使用されている場合、項目のフォーム表示ラベルを自動的に表示します。標準またはカスタムオブジェクト項目に対応付けられているコンポーネントでは、表示されるラベルは、デフォルトでは、オブジェクト項目表示ラベルです。このデフォルト値を上書きする場合、およびオブジェクト項目に直接対応付けされていないコンポーネントの場合は、コンポーネントの `label` 属性を使用して表示ラベルを設定できます。次に例を示します。



Quick Edit: Howard Jones

▼ Contact Details

Phone

Mobile #

Howard's Email



メモ: このページで取引先責任者データを表示する場合、有効な取引先責任者レコードの ID をページの URL のクエリパラメータとして指定する必要があります。次に例を示します。

http://*salesforce_instance*.my.salesforce.com/3000000R5

レコードの ID の取得についての詳細は、[Visualforce での項目値の表示](#)(ページ 20)を参照してください。

属性には、文字列、または評価結果が文字列になる式を指定できます。
と、その項目のフォーム表示ラベルは表示されません。

属性は次の Visualforce コンポーネントで設定できます。

```

et n      • u ldp p < á   直    >
et t      • u r p pc < á       S    >
et n tx   • u p p  á >
et n etx  • u p p  á       a> a
et cu u lph p < a   直    >
et tx cu u p p <T a       >
et e lx   pb oc kc c < ks           s >
et t l    p c < as L  i s >
et l d po c < as     R i a>

```

カスタム表示ラベルとエラーメッセージ

属性が設定されている場合、項目が必須であるときまたは一意になっている必要があるときなどに、この属性はコンポーネントレベルのエラーメッセージで使用されます。カスタム表示ラベルはカスタムエラーメッセージでは使用されません。デフォルトのオブジェクト項目の表示ラベルが代わりに使用されます。属性を空の文字列に設定すると、デフォルトのオブジェクト項目の表示ラベルはすべてのエラーメッセージで使用されます。

フォームの項目のタブ順序の設定

Visualforce フォームには入力項目をタブで移動する場合の「自然な順序」(左から右および上から下)が用意されています。一部のフォームでは、この順序が必ずしも最も効率が良い配置方法、あるいはアクセスしやすい配置方法ではない場合があります。ページの入力コンポーネントおよびその他のコントローラーに `tabIndex` 属性を設定してこのタブ順序を任意の順序に変更できます。次に例を示します。

```

t n t egndprd por id nC   au acc a a   " " >
x ro m p f < a >
x teTB l gat p l pt dAn < dAn cca oj:i ce" mi ! N "a >
ex e TB legn p po et la k oan ad n bo u i m c=" D i sa 1s=" " >
e ex x et n e { u ldp lteAn <t oej a ooc mib v a" I! N "a a =" " 4> /
e ex x et n e { u ldp let an t< ot e)cc nñ v d a" WI . $ " a =" " 3" > /
e ex x et n e { u ldp lt an t& out a)car ik lvb d a" ! I .2 s "y a =" " > /
e ex x et n e { u ldp lt an e an bu antneE vnk ba" d! I. Ra " al =" " > /
ex e TB l gat p po c <c k/a a >
ex ro m p f < /a >
x g p p < /a >

```



メモ: このページに取引先データを表示するには、有効な取引先レコードの ID をページの URL のクエリパラメータとして指定する必要があります。次に例を示します。

```
http://salesforce_instance.com/exp/x bJ /a /y Real=00 008 s
```

レコードの ID の取得についての詳細は、[Visualforce での項目値の表示](#)(ページ 20)を参照してください。

このページを表示して Tab キーを押すと、通常期待する順序とは逆順に有効な項目が変わります。

`tabOrder` 属性は 0 ~ 32767 の整数、または評価結果がこの範囲の整数値になる式である必要があります。タブ順序は、ユーザが Tab キーを押すと選択される最初のコンポーネントであるコンポーネント 0 から始まります。

`tabIndex` 属性は次の Visualforce コンポーネントで設定できます。

```
t •obndmm pc o < a a >
```

	n	no	dmm	p	<	k	L	ia	>
et n	*	u	b	o	b	c	<	ká	
et n	.	u	l	d	p	p	<	á	í
ten	.	u	l	p	p	p	<	á	í
etn t	.	u	r	p	p	c	<	á	s
etn tx	.	u	p	p	p	t	<	á	
etn ex	.	u	p	p	p	t	<	á	a> a
te	c	u	u	b	b	p	<	a	L
t	n	o	u	u	p	p	<	ka	L
et	e	l	x	p	b	o	c	<	ks
t	t	l		p	c	c	<	as	L
t	l	d	p	o	c	c	<	as	R i a>

反復内のコンポーネントに対する tabIndex の設定

de e ble ex t または a p a ap a の内部などで反復コンポーネントによって繰り返し処理されるコン te nk ポーネントに I 属性を設定できます。ただし、この場合、必要な作業が若干増えます。

アクセスされると自動的に増分を行う Apex の getter メソッドを使用して I を設定するというソリューションは、明解に見えますが、機能しません。Visualforce は getter メソッドの結果をキャッシュするため、ページを使用するたびに getter メソッドがコールされるとは限りません。Visualforce の getter メソッドの実装方法についての詳細は、「[Visualforce ページ内の実行順序](#)」(ページ 88)を参照してください。

代わりに、反復処理されるコレクションの各要素でオブジェクト参照または項目参照と共に I 値を指定できます。次に、Visualforce ページ内にこれがどのように表示されるかを示します。

	te n	ro m p f	g o r pl a ep a	< or l d a pp a	= " O s	" >
x te	ed{	bl p l	ed{	bl p l	ut n T t r o	u ak p p a v a e h " I ! o r W id i sipp pp " v W = " a a" >
x	n lo u m p x	n lo u m p x	n lo u m p x	n lo u m p x	< a >	
e ex	te e n e	p m d t en roex u at pp = a h " o i < y / a a >				
e ex	xt et e o u u l p d l p	xt et e o u u l p d l p	xt et e o u u l p d l p	xt et e o u u l p d l p	< o e r n a l o p vppm w p ! a . . " a > /	
e x	n lo u m p x	n lo u m p x	n lo u m p x	n lo u m p x	< /a >	
e ex	te e n e	p m d l ex o t m a a = a p f " a > < /a a >				
e ex	et n e { u l d p d l p	et n e { u l d p d l p	et n e { u l d p d l p	et n e { u l d p d l p	& r t d a no h p p ip a W = ! a . a " > /	
e x	{ nk b d l t e r n d h p p	{ nk b d l t e r n d h p p	{ nk b d l t e r n d h p p	{ nk b d l t e r n d h p p	ip a W = ! a . a " > /	
x te	d bl p	d bl p	d bl p	d bl p	< /a >	
ro g p p	ro g p p	ro g p p	ro g p p	ro g p p	< /a >	

d bl p コンポーネントは商談レコードのリストを反復処理しません。ただし、I d o pp pp W pp ! として参照される商談と、nd b pp pp W t e! として参照されるその I をラップするオブジェクトのリストは反復処理します。次に、このコレクションを提供するコントローラを示します。

```
blu t e n p c c or l d r pp i ss aO s
e t t e t o n ro u f pp /G sa o i i s
e x blA t e n ptc g t el p d e t{or l d m C oBa . S @ a S s
e t n { n o fllic i s( == )
e t e newn ex A t C e n t g t el p d s t e r l d r t C b Bat S g a a s ro D c ( a s q a L y
e t e e net E m e Toum l l p o d [St ct na, rg M a pp s ] E R O O i y;
e t e t e n e o z C g s s s @ a S ( ; 5)
t e tn r rm o C s ;
}
```

```

        s ;
    t bl t ten pto ut et pp g ro i <s pp i y o i i s ( )
    ntr rut t en to e u e tn pp o L g esd o ci y s ) . R s ;( )

t bl e e p r t ret pp tpp t l e t i <pp h o u i pp s = i s . o i i s ;( )
t t n ro u e pp t l e t i <pp h o u i pp s = i s . o i i s ;( )
t e re r e o pp w n p d l <pp o pp w a > r s pp a p l <pp o a > ;( )
t ne x r gl I i l = ;
t to n ro fu o pp o pp o pp i y s ) 
e e r wnd add pp ppr W ix os pp d pp . Wap( o a ( i , ; ) )
x d ++ i ; 

n re ru o r d pp pp W s a;
bl e l p c rc r pp pp p W ss a o a
t bl t n eo pte to pp lg ip p o i y ; s ;
e et n b t e { ex p g b l dt I g i a ; s ;
e bl t p t r n rep t uppn t wip { Onx g p a d o i I y , a )
t t e n k t b d h nk b d I i s . a = a ; 
}
}

```

次の点を確認してください。

- 内部クラス pp p は商談への参照とインデックス番号を組み合わせます。
- リストは、それぞれの OppWrapper の I の位置を計算して OppWrapper のリストを作成します。

ページへの連動項目の追加

連動項目では、Visualforce ページに表示される項目の値を絞り込むことができます。連動項目は、条件を決定する制御項目と、値が条件で絞り込まれた連動項目の 2 つで構成されます。連動項目は、選択リスト、複数選択リスト、ラジオボタン、およびチェックボックスなどの項目の値を動的に絞り込むことができます。連動選択リストは、Salesforce API バージョン 19.0 以降を使用する Visualforce ページでのみ表示できます。詳細は、Salesforce オンラインヘルプの「連動選択リストについて」を参照してください。

この例では、連動選択リスト(サブカテゴリ)を Visualforce ページに追加します。まず、カスタム選択リストを作成します。

- [設定] で [カスタマイズ] > [取引先] > [項目] をクリックします。
- ページの [カスタム項目 & リレーション] セクションにある [新規] をクリックします。
- [選択リスト] を選択し、[次へ] をクリックします。
- [項目の表示ラベル] に「サブカテゴリ」と入力します。
- 値のリストに次の語句を入力します。

- りんご農場
- ケーブル
- とうもろこし畑
- インターネット
- ラジオ

- ・ テレビ
- ・ ワイナリー

6. [次へ] を 2 回クリックし、[保存] をクリックします。

サブカテゴリの項目の連動関係を定義する手順は、次のとおりです。

- [設定] で [カスタマイズ] > [取引先] > [項目] をクリックします。
- [項目の連動関係] をクリックします。
- [新規] をクリックします。
- 制御項目として [業種] を、連動項目として [サブカテゴリ] を選択します。
- [次へ] をクリックします。
- 制御項目の各値 (業種) は、一番上の行にリストされており、連動項目の各値 (サブカテゴリ) はその下の列に表示されます。この画像に一致するように、項目の連動関係を設定します。

Industry:	Agriculture	Communications
Subcategories:	Apple Farms Cable Corn Fields Internet Radio Television Winery	Apple Farms Cable Corn Fields Internet Radio Television Winery

図 11: サブカテゴリの項目の連動関係マトリックス

上記に表示されていない他の [業種] の種別は無視できます。

7. [保存] をクリックします。

次のような Visualforce ページを作成します。

```

t      n   tegndprd por hñ c     au   acc   a a      " " >
x      ro   m p f <           a >
e      ex   e B   l go p do pm dc <           ka   a      " i " >
e      ex   tB   lbgo p u p o c <           ka   a      s >
e      ex   x   tt   obnømm øeno   ø ø ø a l u a   ia =''   ø   "   a' v   sa > /
e      ex   tB   lbgo p u p o c <           k/a   a      s >
e      ex   e   tB   hega epe p o c inø n   kda   a l p ø   dø uD =m ck   ø   ø s s'   s=" " >
e      x   et n   e { u ldp lt øn tñ ou ø car ø v   æ" !   a i.   s   "y > /
e      x   et n   e { u ldp lt øn eou øucc øro} g c   æ" !   ca   s   ai   s   " > /
e      ex   e   tB   l ga p   p o c <c   k/a   a   ø >
x      ex   B   l go p   p   c <           k/a   a >
x      ro   m p f <           /a   a >
x      g   p   p <           /a   a >

```

[業種] 選択リストから [農業] を選択すると、サブカテゴリ選択リストには [りんご農場]、[とうもろこし畑]、および [ワイナリー] が表示されます。[通信] を選択すると、[サブカテゴリ] 選択リストには、上記で定義したすべての通信の種別の項目が表示されます。

連動選択リストの考慮事項

Visualforce ページで連動選択リストを使用するときは、次の点に留意してください。

- 選択リスト、複数選択リスト、ラジオボタン、およびチェックボックスなどのさまざまな項目の種別で制御項目と連動項目を混在させて使用できます。
- 1 つのページに使用できる連動選択リストのペアは最大 10 個です。これはすべてのオブジェクトで合計されます。したがって、Account オブジェクトに 5 つの連動選択リストを使用し、Contact オブジェクトに 5 つの連動選択リストを使用できますが、それ以上使用することはできません。ただし、連動選択リストの同じペ

`<u>`を `<p>` などの反復タグで繰り返し使用することができ、この場合、制限との関係では1回のみカウントされます。

- ・ ページを表示しているユーザに制御項目への参照のみのアクセス権がある場合、連動選択リストが期待どおりに動作しないことがあります。この場合、連動選択リストには、参照のみの値で絞り込まれる代わりに、選択可能なすべての値が表示されてしまいます。これは Visualforce の既知の制限です。
 - ・ 連動選択リストを使用する場合、ページに制御項目を含める必要があります。ページに制御項目が含まれていない場合、ページを表示するときにランタイムエラーが発生します。
 - ・ インライン編集を有効にした項目と同じ連動関係グループの通常の入力項目を混合しないでください。たとえば、制御項目の標準の入力項目とインライン編集を有効にした連動項目は混合しないでください。

t egnd^rd por h^h & n C a u & cc a a = " " >
ro m p f < a >
x t o n m t e n d r d < uD! l d- ip' f s a a ia i . . . >-
e u l d p l w < á ñ y ñ

ラを使用しているか、または StandardSetController クラスにバインドされたページを参照している必要があります。Visualforce ページは、これらの要件を満たさない場合、ダッシュボードコンポーネントの「ページ」ドロップダウンリストにオプションとして表示されません。

b ad h という Visualforce ページを作成します。次のマークアップは、標準リストコントローラを使用する、ダッシュボード内で使用できる Visualforce ページの例を示しています。このページは、組織に関連付けられたケースのリストを表示します。

```
t      n t e g n d p r d p o e l d s e C t a r e r d a c a      = " c s " a v s = " a s s a >
x      B l g o p p c < k a a >
e x      et ro d m p f r o h < a i = " F " >
e ex      n l r d p l o p u m c < a i G s = " " >
e e ex      t e l e p t l c u e g r z d a s f l i s v a i i " s l = " " >
e e ex      t t n e t e d u p r o e n e s t o p n a r g i d a d r S l v = " a = " i s " >
e e ex      t t l e n { p o e t c l u w t p k l n a s j o o O i s p a i i s i i s " >
e ex      n l r d p p < / a a i G >
e ex      e t B l g a p p o c & k a a s >
{ e ex      t t d e { p r e l & c t a L d a s t = " a v a ! s s a " i = " i s " >
e ex      t ) b u j c c ! s >
e ex      t t d p < / a L i a s a >
x      ro m p f < / a a >
B l g o p p c < k a a >
g p p < / a a >
```

この Visualforce ページを使用するダッシュボードを作成する手順は、次のとおりです。

1. ダッシュボードを表示し、[編集] をクリックします。
2. 任意の列の上部にある [コンポーネントの追加] をクリックします。
3. コンポーネントの種類として [Visualforce ページ] を選択します。
4. 必要に応じて、ダッシュボードコンポーネントの上部に表示するヘッダーを入力します。
5. 必要に応じて、ダッシュボードコンポーネントの下部に表示するフッターを入力します。
6. ページ] ドロップダウンリストから、 を選択します。
7. [保存] をクリックします。

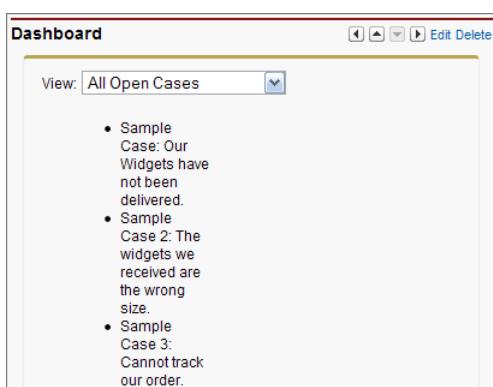


図 12: ダッシュボードで実行する Visualforce ページのサンプル

カスタムリストコントローラを使用する、より複雑な例については、「高度な Visualforce ダッシュボードコンポーネント」(ページ 120)を参照してください。

カスタムオブジェクトの関連リストの表示

Visualforce を使用してカスタムオブジェクトとその関連リストを表示するのはとても簡単です。

MyChildObject、MyMasterObject、および MyLookupObject という 3 つのカスタムオブジェクトがあったとします。MyChildObject には MyMasterObject との主従関係があります(後者が主)。MyLookupObject にも MyChildObject との参照関係があります。

MyMasterObject の関連リストを表示する Visualforce ページを作成する場合は、次のマークアップを使用します。

```
t n tegndprd pørtlæs C t a r b M Mj a a c c ='' ys a o " " >
t r lt d ple t <ld b M k E á s i s ='' y o s " _ > /
```

このページに関連リストデータを表示するには、カスタムリレーションのある有効なカスタムオブジェクトトレコードの ID を、`h l : prø eot fnctco l m Q/ sa sá . /ys R j L á s` などのように、`00000000` ページの URL のクエリパラメータとして指定する必要があります。

MyLookupObject は、別の種別のリレーションを使用しますが、構文は同じです。

```
t n tegndprd por læs C t ao ub M j a apck c ='' y o " " >
t r lt d ple t <ld b M k E á s i s ='' y o s " _ > /
```

オンライン編集の有効化

バージョン 21.0 以降の Visualforce ページは、オンライン編集をサポートしています。オンライン編集では、レコードの詳細ページで直接、項目値をすばやく編集できます。編集可能なセルには、その上にマウスを置くと鉛筆アイコン (✎) が表示され、編集できないセルの場合は、錠アイコン (🔒) が表示されます。

コンポーネントには、オンライン編集を有効にする属性があり、`t d p u r o E < pp a i` コンポーネントには、さまざまなコンテナコンポーネントのオンライン編集機能が用意されています。

オンライン編集の効果を確認するために、次のコードを使用して `l` というページを作成します。

```
t n tegndprd por læs C au acc a a ='' " " >
x t e dt l bpt j ne e <du d teceli ds ='' ! a f. " L á s ='' s 'a > /
```



メモ: このページに取引先データを表示するには、有効な取引先レコードの ID をページの URL のクエリパラメータとして指定する必要があります。次に例を示します。

```
h e : p Salesforce_instance m xp x bJ / a /y Ral= 00 000 s
```

レコードの ID の取得についての詳細は、[Visualforce での項目値の表示](#)(ページ 20)を参照してください。

`取引先番号` などの項目をダブルクリックしてみてください。何も起こりません。

ここで、ページを次のコードで置き換えます。

```
t n tegndprd por h An C au acc a a " " >
e xt e dt l bpt jne e<du d tec ei ds e f"l m et a f.d" rL u s=" s'ai i i ="
g p p < /a >
```

任意の項目にマウスポインタを重ねると、コンテンツを直接編集できることがわかります。セクションの上部にある [保存] をクリックすると、すべての変更された情報が保持されます。オンライン編集をサポートするコンポーネントは、必ず `m p f` タグの子孫である必要があります。ただし、`l p` コンポーネントは、オンライン編集をサポートするために `m p f` の子孫である必要はありません。

コンポーネントは、次のコンポーネントの子孫である必要があります。

```
t d . p < aL iasa >
e d . bl p T< a a a a >
ro m p f < a >
e ou u lph p < a >
B lgo p p c < ka a >
e tB lga p poc & ka a s >
e B lgo pbl p c < Tk a a a >
t r . p p < a > a
```

次は、オンライン編集を利用する `l go pbl p c <` を使用してページを作成する方法を示すサンプルです。

```
t n tegndprd por h An Ce au er accd a ar cert erd=" d c " v g h S =" a s'i =" "Pa>
x et ro d m p fro h < a i =" F " >
e ex B t lgo pd p B k gdm k a ca i ="k Pa " >
e ex e B el{go epl p lc u < re}T krda arcer erda d at B l gdm bl ="a c " i k Pa
e ex x n lo u mpc < a >
e ex xt et eo u ue lph lep r< rd a tem& v ou a" cbm N Mai =" D aO"
e ex ex te ene e p and rex &t m a a fa' 'a < a /a a >
e ex x n lo u mpc < a >
e ex xt et eo u ue lph lep r< rd a te & v qm a" cb . yMi =" D y O"
e ex ex te ene e p and rex &t a a fa' 'a < y /a a >
e ex x n lo u mpc < a >
e ex xt et eo u ue lph l ip t& rd a ur & I v a" ! . s "y
t An tn ou d or I i M" B yO" > /
e ex ex te ene p findr dex urt aI a =a p fa > s< y /a a >
e ex x n lt e d p unroEn < pblal i i C cs v k" i "
w tne o td heBn Entto eBnl su&nOcidt="sd ha Bn Eu a " i O i =" i
e ex e B lgo pbl p c < Tk/a a a >
e ex tB lBgo pu p c < ka a a s >
e ex tt Bndmm w to lt u d & a & } a tt d B& i u o ia =" k "ai =" i "
e ex tt Bndmm w o lt u < & a o & a tt d B& v &ao ia =" k "ai =" & a "
e ex tt Bndmm w o lt {< lne a Cno a) e vl ntde" c Bnd" uice e' ! a" i =" a
e ex tB lBgo pu p c < k/a a a >
B lgo p p c < k/a a >
ro m p f < /a >
g p p < /a >
```

オンライン編集がサポートされない場合を次に示します。

- 次の場合は、オンライン編集できません。
 - ◊ アクセシビリティモード

◊ 設定ページ

◊ ダッシュボード

◊ カスタマーポータル

◊ HTML ソリューションの説明

- ・ ケース、リード編集ページにある次の標準チェックボックスは、オンライン編集できません。

◊ ケース割り当て（有効な割り当てルールによりケースを割り当てる）

◊ ケースメール通知（メールで取引先責任者に通知する）

◊ リード割り当て（有効な割り当てルールによりリードを割り当てる）

- ・ 次の標準オブジェクトの項目はオンライン編集できません。

◊ ドキュメントおよび価格表のすべての項目

◊ 件名] および [コメント] を除く ToDo のすべての項目

◊ 件名]、[説明]、および [場所] を除く行動のすべての項目

◊ 個人取引先、取引先責任者、およびリードの氏名項目。ただし、コンポーネント項目は、姓] や [名] などです。

- ・ 項目レベルセキュリティまたは組織の共有モデルによるオンライン編集を使用して、読み取り専用の権限しか持っていないレコードの項目の値を変更できます。ただし、変更内容を保存することはできません。保存しようとすると、権限が不十分であるため保存できない旨のエラーメッセージが表示されます。

- ・ Visualforce ページが salesforce.com ドメインではなく、別のドメインから配信される場合にバイパスされるなどの標準リッチテキストエリア(RTA)項目のオンライン編集はサポートされません。デフォルトでは、Visualforce ページは、システム管理者がデフォルト設定を無効にしない限り、別のドメインから配信されます。カスタム RTA 項目は、この制限の影響を受けないため、オンライン編集がサポートされます。

- ・ インライン編集は、連動選択リストでサポートされます。
- ・ 連動選択リストを使用する場合、ページに制御項目を含める必要があります。ページに制御項目が含まれていない場合、ページを表示するときにランタイムエラーが発生します。
- ・ インライン編集を有効にした項目と同じ連動関係グループの通常の入力項目を混合しないでください。たとえば、制御項目の標準の入力項目とインライン編集を有効にした連動項目は混合しないでください。

- ・ インライン編集を有効にした連動選択リストと Ajax スタイルの部分ページ更新を組み合わせる場合は、相互に連動または制御関係にあるすべての項目を 1 つのグループとして更新します。項目を個別に更新することはお勧めしません。元に戻す動作またはやり直す動作の一貫性がなくなる場合があります。インライン編集を有効にした連動選択リストがあるフォームを部分更新する方法の推奨例を次に示します。

```

x t te nou u t l p & i m po < o ca c Pá ="k i ai p " >
e xt et eoù u l p tl p n t o n oa } ocr E v ! i a . i "y >
e xe n it le te d p unroEn < pblál i i C S v k" >
e xt et ou u l p p < /a >
e xe n it le te d p unroEn < pblál i i C S v k" >
e xt et eoù u l p tl p < /a >
e xe n it le te d p unroEn < pblál i i C S v k" >
e xt et ou u l p p < /a >
x t te nou u l p p < /a Pa>
x t te < ! -->
x ro < /a >

```

オンライン編集を有効にした選択リストのすべてがコンポーネントでラップされています。アクションメソッドが起動すると表示されます。

PDF としてのページの表示

任意のページを PDF として表示するには属性を g p p コンポーネントに追加し、表示サービスとして「pdf」を指定します。次に例を示します。

```
rg& p pd < p a s=" " >
```

PDF として表示された Visualforce ページは、ブラウザ設定に応じて、ブラウザに表示されるか PDF ファイルとしてダウンロードされます。

前のチュートリアルでは、Visualforce ページを使用して会社名を変更しました。新しい社名の発表を PDF として生成するとします。次の例では、そのページを現在の日時で作成します。

```

n t egndprd por h M C nau r a dcr a al Bel = do pf " p s=" Tf " a y y=" a s "a >
d h < >a
l < s y >
t b do o l amf nr fl : do } y c i Ma y ij ' á ; s '
e{ n o t m n c m qpx b lde f : . loro Mayda : 3 ;0 ;
l < s /y >
d h < >a
t n r c < >y
w t eAn h ou cc m <1 h N N < 1h >
ex n l r d p l p m c< d a aIG 1s=" " i 1 = " >
e x t e t x e fu u p l t e u p < T e t o u a c c m l vle n a' ol C ma cN m "ap s y ss="a
e x t e t x e fu u p l uep } < T e t e t a t x ou vu W p a' N p & T /a >
e n l r d p p < /a aIG >
e t n r c < >/y
b do g p p < /a a

```

このページについては、次の点に留意してください。

- ・ は CSS マークアップであり、Visualforce マークアップではありません。CSS マークアップでは、ページ全体に使用するフォントファミリと、会社名用の特定のスタイルを定義します。
- ・ 出力テキストの一部は、コンポーネントに含まれます。パネルグリッドは HTML テーブルとして表示されます。コンポーネントの本文内の各コンポーネントは、列数に達する

まで、最初の行の対応するセルに配置されます。セルは 1 つしかないため、各出力テキストは別個の行に表示されます。

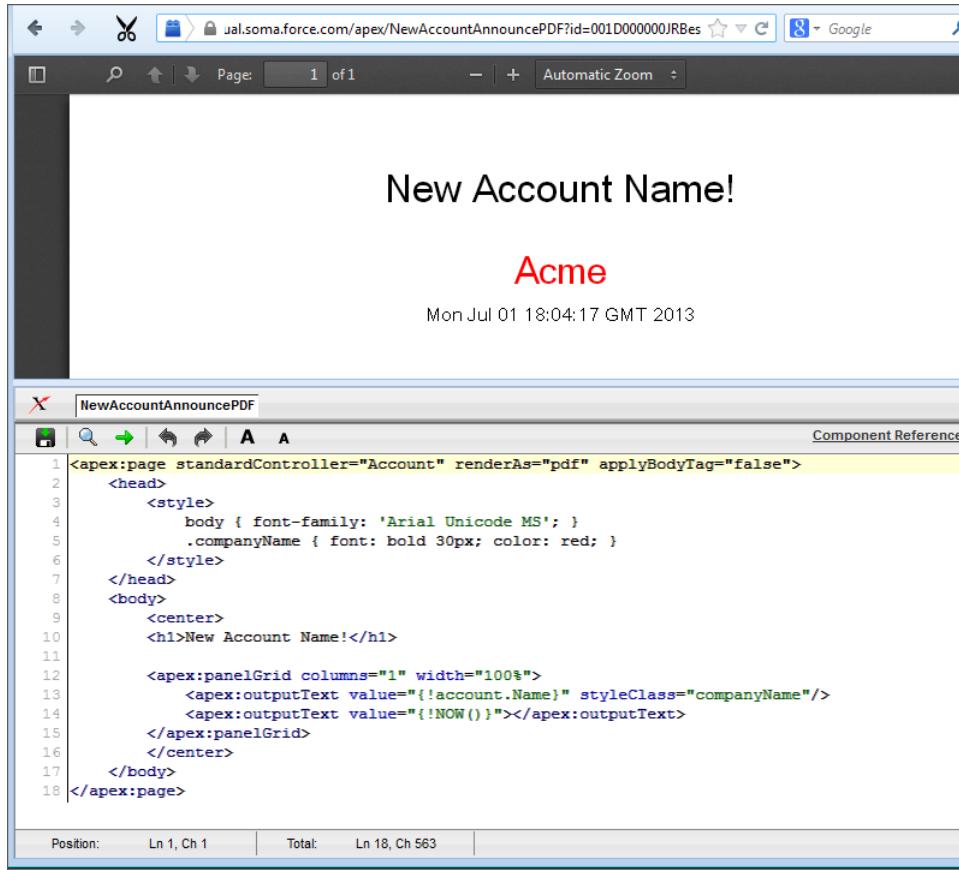


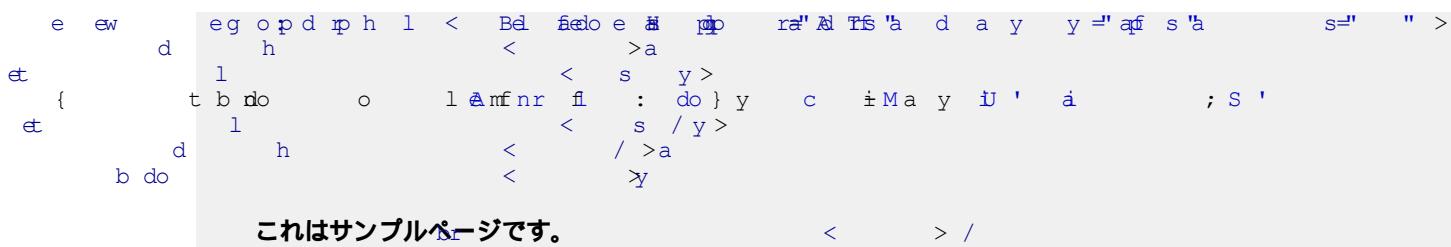
図 13 : PDF として表示された Visualforce ページ

リリースする前に、表示されるページのフォーマットを確認してください。

を使用する場合、次の点

に留意してください。

- ・ サポートされている表示サービスは PDF のみです。
- ・ Visualforce ページを PDF として表示する機能は、印刷用にデザインされ、最適化されたページのためのものです。
- ・ 印刷用の書式設定が容易ではないか、入力やボタンなどのフォーム要素が含まれる標準コンポーネント、または書式設定に JavaScript が必要なコンポーネントは使用しないでください。これには、フォーム要素が必要なコンポーネントなどが含まれますが、これに限定されません。
- ・ PDF 表示では、JavaScript で表示されるコンテンツはサポートされていません。
- ・ PDF でページのすべてのテキスト(特に日本語などのマルチバイト文字やアクセント記号付きの国際文字)が表示されない場合は、CSS のフォントを調整してそれに対応するフォントを使用します。次に例を示します。



```
e   e      hel  A  g npr pi si ls  sa  a2      a v  P $      8  0
b do      g  p  p      <      /  y
          g  p  p      <      /  y
```

選択したフォントは、Visualforce PDF 表示サービスで使用できる必要があります。現在、マルチバイト文字を含む拡張文字でサポートされているフォントは「Arial Unicode MS」のみです。

- PDF 作成時の最大応答サイズは、PDF として表示される前で 15 MB 未満です。これは Visualforce 要求の標準制限です。
- 生成される PDF の最大ファイルサイズは、60 MB です。
- 生成された PDF に含まれるすべての画像の最大合計サイズは 30 MB です。
- PDF 表示では、URI スキーム形式で符号化された画像はサポートされていません。
- 次のコンポーネントは、PDF として表示するときに 2 バイトのフォントをサポートしません。

```
B  l go p  p c  <  k a  a  >
o  o  rc  <  as  a>
```

PDF として表示するページでこのようなコンポーネントを使用することはお勧めしません。

関連リンク

[Visualforce ページのスタイル設定](#)

[PDF を表示するためのベストプラクティス](#)

ページでのデータのテーブルの作成

または `l go pbl px t e < a>` などの一部の Visualforce コンポーネントでは、レコードのコレクションを反復することによって、一度に複数のレコードの情報を表示できます。この概念を説明するために次のページでは、`l go pbl p c < a>` コンポーネントを使用して、現在コンテキストにある取引先に関連付けられた取引先責任者をリストします。

```
n  tegndyrd por hñ n C  au  acc  a  a  =''  "  >
t  te B el go p d p cll <de  kra  r)a  i mH$=  U ! s  E  s N  a!  >
e  w@turn  tegn n  dñ tY  ac mñ oiu  cc  !  a  .  a  a  .
x  B  l go p  p c  <  k/a  a  >
te B  l gñ ptnlp co <  kCa  c  a  i  =''  sñ  >
e  B  el{go pbl p lñ u st Trkoai  joac t  trñar  c  a!  !  a  .  c  sa"v  ='' a  " a>
x  e ñ  lo ut mñau  o  ñ  c  amc  v  a!  !  a  .  a  a  > /
x  e ñ  lo ut mñau  o  ts  ñ  a  c  gñ  a!  !  ai  i  a  i  "y  > /
x  e ñ  lo ut mñau  on <}  co a c  hv  a!  !  a  .  P  > /
ex  e  B  l go pbl p  c  <  T k/a  a  a>
B  l go p  p c  <  k/a  a  >
g  p  p  <  /a  a>
```



メモ: このページに取引先データを表示するには、有効な取引先レコードの ID をページの URL のエクスパラメータとして指定する必要があります。次に例を示します。

```
h  e :p Salesforce_instance m  xp  x  bJ  / a  /y  Ra1?  00  008 s
```

レコードの ID の取得についての詳細は、[Visualforce での項目値の表示](#) (ページ 20)を参照してください。

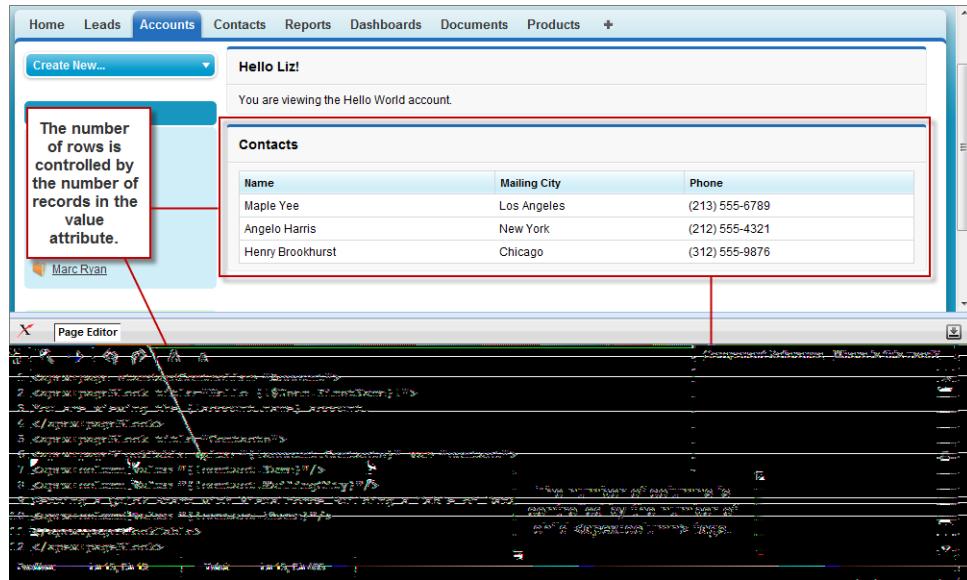


図 14:<apex:pageBlockTable> コンポーネント

他の反復コンポーネントと同様に、`l go epl p c <1 tneu` には、`a` と `l` という 2 つの必須属性が含まれます。

`l u .` は、`sObject` レコードまたは他のすべての Apex 型の値のリストを取ります。上記の例では、
`nt tn ou jbcc c c` は現在コンテキストにある取引先の ID を取得してから、リレーションをトラバースして、関連付けられた取引先責任者のリストを取得します。

`r .` は、反復変数の名前を指定します。この変数は、各取引先責任者の項目にアクセスするために、
`e B l go pbl p c < tneu` に、`oe }` が `mc vn !mpc` タグで使用されています。

`e B l go pbl p c <` コンポーネントは、1 つ以上の子 `u mpx` コンポーネントを使用します。テーブルの行数は、`l` 属性を使用して返されるレコード数によって制御されます。

 **メモ:** `l go pbl p c <` コンポーネントは、標準の Salesforce リストのスタイルを自動的に適用します。独自のスタイルでリストを表示するには、代わりに `bl p` を使用します。

ページでのデータのテーブルの編集

最後のチュートリアルでは、データのテーブルを作成しました。データテーブルの列に `u ldp p` を使用することによって、編集可能項目を含むテーブルを作成できます。`Bndmm w o <` を使用すると、変更したデータを保存できます。メッセージ (`g w i spal <`) は、`g w i spal <` タグで自動的に表示されます。

次のページでは、複数の業種を同時に編集できるページを作成します。

```

t n tegndprd por hAnCe au r accd at anc o w cc " v s = a a s"
b t len oue dcbr l say f a " $ = a s 'a >
ro m p f < a >
B l go p p c < ka a >
g p opM < a ass sa > /
ttB lBgo p u po c < ka a s >
ttB eBndmme w o lt u < a a o } a v a v s a i a = l a > /

```

```

t tB lBgo p u p c < k/a a s >
e B el{go pbl p lat u s T k o a v a va a! a s "v =! a! >a
e n lo ue mpx } < m a v a! ! . a 'a > /
e e n do u mertnl hu d ur a I V a a! s "y > /
et n e { u ldp l tmp < d ur a I V a a! s "y > /
n lo u mpx < /a >
e B l go pbl p c < T k/a a a>
B l go p p c < k/a a >
ro m p f < /a >
g p p < /a >

```

 メモ: URL に ID 属性がある場合、このページは正しく表示されません。たとえば、
n h e :pdu eo emax g fnc//aliasredpsal . a . HW/a / I では重複が発生します。URL から ID を削除する必要があります。

ページマークアップについては、次の点に注意してください。

- このページは、[コントローラの標準セット](#)を活用して、テーブルのデータを生成します。使用するデータセットの名前を指定するには、`rc` 属性を使用します。次に、`l go pbl p c <` の値に、そのセットの名前を使用してテーブルにデータを入力します。
- `u ldp p` タグは、項目の正しい表示を自動的に生成します。この場合は、ドロップダウンリストとして表示を行います。
- `oBndmm uc o < タグを使用するには、ページが m p f タグで囲まれている必要があります。フォームによって、Visualforce ページ内の、ユーザが操作できる部分が指定されます。`

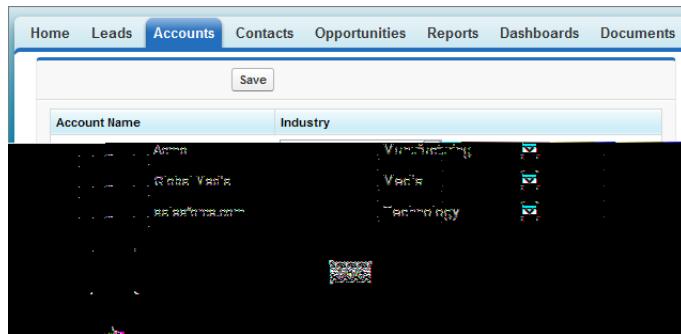


図 15: データテーブルの編集例

ページでのクエリ文字列パラメータの使用

前述の例で示すように、デフォルトのページコンテキスト（つまり、ページに表示されるデータのソースを提供するレコード）は、ページ URL 内の `__` という名前のクエリ文字列パラメータで制御されます。クエリ文字列パラメータを取得して Visualforce マークアップに設定することもできます。次のトピックで例を参照してください。

- クエリ文字列パラメータの取得
- リンクでのクエリ文字列パラメータの設定
- 単一ページでのクエリ文字列パラメータの取得と設定

クエリ文字列パラメータの取得

Visualforce マークアップでクエリ文字列パラメータを参照するには、`#{g}` グローバル変数を使用します。`#{egेत}` を使用すると、`g` 属性を指定してページのクエリ文字列パラメータにアクセスし、その後、個別の各パラメータにアクセスできます。

C q \$rm p parameterPanama as .

たとえば、特定の取引先責任者に関する詳細情報を [取引先] ページに追加します。取引先レコード ID はデフォルトの クエリ文字列パラメータで指定され、取引先責任者レコード ID は。 というクエリ文字列パラメータで指定されます。

egndprd por tå Nc eu acc a a = " " >
e l go p d p cll <de kra r)a i mH\\$" U ! s E s N a" >
ur edn l et lg upr oemmn nā shj t yace mn sœu t d cct !tm a .o ap ca c a sa a
eh d b cp eq tr i srg i rm py a y i a a .
B l go p p c < k/a a >
l gat ptlnlp co < kCa c a i = " sa >
ed bl plt unt thkou)act aenra c a" ! mle dd c sa bo rd # a " a iPa = " 4

n lo u mp < a >
te ene e p fnd rex ht m a a =p f 'a N< a /a a>
tre o } c nc ! a
n lo u mp < /a >
n lo u mp < a >
te ene e p fndnr ex h a ah a =p f 'a > < P /a a>
e on } co c h ! a P
n lo u mp < /a >
d bl p & /a a a a>
B l go p p c < k/a a >
l e bu te j n erct a gie d p' rel cd te t P a als f l L afs = s " a i =

この例が正しく表示されるためには、Visualforce ページを URL 内の有効な取引先 ID および取引先責任者 ID に関連付ける必要があります。たとえば、
I が取引先 ID で 000B003 5 IE が取引先責任者 000000 0
ID の場合、URL は次のようにになります。

h e : p **Salesforce instance** // pM dg t / a My s Bal b 00 000B00DE i5&D 3 0Q 00000

レコードの ID の取得についての詳細は、Visualforce での項目値の表示 (ページ 20) を参照してください。



メモ: URL に パラメータを使用する場合、そのパラメータは、標準コントローラで参照されるエンティティと同じエンティティを参照する必要があります。



図 16: ページでのクエリ文字列パラメータの使用

リンクでのクエリ文字列パラメータの設定

クエリ文字列パラメータをページへのリンクに設定するには、手動でリンク URL を作成するか、タグ内で使用します。たとえば、次の例はどちらも外部ページへの同一のリンクを作成します。

```

<a href="http://www.salesforce.com/test/test?c=1&p=1">Link 1</a>
<a href="http://www.salesforce.com/test/test?p=1&c=1">Link 2</a>

```

2つ目のメソッドでは、URL を手動で作成するのではなくタグを使用します。スタイル上の理由から、この方法をお勧めします。

xt t  のほか、`Imp i p` を使用して `dmm p` と `ka L ia` の要求パラメータを設定します。

単一ページでのクエリ文字列パラメータの取得と設定

前述のクエリ文字列パラメータの取得と設定の例に続き、この例では、単一ページ上で2つのアクションを組み合わせてより興味深い結果を作成する方法を示します。「クエリ文字列パラメータの取得」の例に基づいて、次のページではリストに含まれる各取引先責任者の名前をハイパーリンクにし、その下に表示される詳細コンポーネントのコンテキストを制御します。

これは、次の操作により実行できます。

- `データテーブル` を `m p f` タグでラップする
- 各取引先責任者名を、`m p p` タグで適切な `ca` パラメータを設定する

標準コントローラと一緒に使用されると、コマンドリンクは常に現在のページをページに新しく追加された情報で完全に更新します。この場合は、更新された `c` が取引先責任者詳細コンポーネントを更新します。

```
t n tegndprd por hñ An C au acc a a = " " >
x teBelgo p d p cll <de kra r} a i mH $! U ! s ñ s N a! >
e our dnt ltn etg p rcYot emn áns diij ya ac m seu cc ! a . a a .
ex t ltn e ñCec ñkemo e t ro ránd ñh ' av i is i sa .
x teB l go p p c < k/a a >
x ro m p f < a >
e x t e ed{ bl plt unt Th ou )acc t avar c a! ! al dd c saly = a " a ipa
e x n lo u m < a >
e ex t e ene e p d rex at m a a ñf 'a ñ< a / a a>
e x n no dmm < ka L ia >
{ t tne o } c mc ! al a
e x e n r e { mp mp t ltnac oa) daca ñai 'v a! ! ia " > /
e x n no dmm < /ka L ia >
e x n lo u m < /a >
e ex t e ene e p fe drn ex a ah a ñf 'a < P / a a>
{ t tne on } co c h ! a P
e x n lo u m < /a >
e x t e d bl p & /a a a a>
x ro m p f < /a >
ex B l go p p c < k/a a >
x t e {dt lebup tej n erect ñgi e ñm d p' rel cd te t P a als f I" L áfs = s'a i
```

このマークアップを保存した後、ブラウザを クエリ文字列パラメータで更新します。ただし、次のように URL に `c` パラメータは指定しません。

```
h e :p Salesforce_instance // pM dg t / a ñy ñ ñal ñ 00 000R003 5
```

最初は、取引先責任者詳細ページは表示されませんが、取引先責任者名をクリックすると、ページに該当する詳細ビューが表示されます。



メモ: URL に パラメータを使用する場合、そのパラメータは、標準コントローラで参照されるエンティティと同じエンティティを参照する必要があります。

関連リンク

[コントローラメソッド](#)

ページでの Ajax の使用

一部の Visualforce コンポーネントは Ajax を認識するコンポーネントであり、JavaScript をまったく記述することなくページに Ajax の動作を追加できます。次のトピックで例を参照してください。

- コマンドリンクとボタンによる部分ページ更新の実装
- 非同期操作のための状況の提供
- 任意のコンポーネントでのイベントへの Ajax 動作の適用

コマンドリンクとボタンによる部分ページ更新の実装

最も広く使用されている Ajax 動作の 1 つに部分ページ更新があります。ページ全体を再読み込みするのではなく、ユーザの何らかのアクションに従って特定の部分のページのみを更新する動作です。

部分ページ更新を実装する最も単純な方法は、更新する必要のあるコンポーネントを識別するために、

または `<apex:commandLink>` タグで属性を使用する方法です。ユーザがボタンまたはリンクをクリックすると、識別されたコンポーネントとそのすべての子コンポーネントのみが更新されます。

たとえば、「[単一ページでのクエリ文字列パラメータの取得と設定](#)」(ページ 46)で説明されている取引先責任者リストの例を考えてください。この例では、ユーザがリストの取引先責任者の名前をクリックしてその詳細を表示すると、このアクションの結果としてページ全体が更新されます。そのマークアップに 2 つの変更を適用することによって、リストの下の領域のみが更新されるようにページの動作を変更できます。

- まず、再表示するページの部分を作成または特定します。これを行うには、`<apex:repeat>` タグを `<apex:repeat>` タグでラップし、出力パネルに パラメータを指定します。この値は、この領域を `id` 参照する名前で、ページのあらゆる場所で使用できます。この値は、ページ内で一意である必要があります。
- 次に、定義した領域の部分ページ更新を実行するために使用する、呼び出しポイント(コマンドリンク)を示します。これを行うには、`<apex:commandLink id="listItem">` タグに `action` 属性を追加し、出力パネルの `action` に割り当てられた値と同じ値をその属性に指定します。

最終的なマークアップは次のようにになります。

```

<apex:repeat id="list" value="{!accounts}" var="a">
    <apex:commandLink id="listItem" action="updateList({!a.id})">
        <apex:outputText value=" {!a.name} " />
    </apex:commandLink>
    <apex:outputPanel id="listContent" rendered="false">
        <apex:repeat id="item" value="a.details" var="d">
            <apex:outputText value=" {!d.name} " />
        </apex:repeat>
    </apex:outputPanel>
</apex:repeat>

```

```

    {   t   tre   o }   c   mc   !   a
      e   x   e   n   r   e { mp   mp   t ltn&c   oa}   daca   =ai   v   a!   ia   "   >   /
e   e   x   n   no   dmm   &c   <   /a   L   ia   >
ex   ex   d   bl   p   &t   /a   a   a>
ext   ex   ro   m   p   f   <   /a   a   a>
ext   ex   B   lgo   p   p   c   <   k/a   a   >
ext   ex   neu   t   b   d   dp   l<   a   Pa   =   i   "a>
ext   ex   dt   l   e   b   p   tej   n   &e   t   aC   gei   &ejt   d   t   r   cd   Pal   a   a   s   if   "   L   a   s   =   s   'a>
ext   ex   te   e   l   l   f   i   =   s   'a   >   /
ext   ex   nou   u   b   p   <   /a   Pa>
ext   ex   g   p   p   <   /a   a

```

ページを保存したら、任意の取引先責任者をクリックし、ページ全体を更新しなくても詳細コンポーネントが表示されることを確認します。



メモ: テーブルのコンテンツを更新するために

属性を使用することはできません。

非同期操作のための状況の提供

部分ページ更新などの Ajax 動作は、ページユーザが作業を進める間にバックグラウンドで発生する非同期イベントです。使い勝手をよくするために、デザイナーは、現在進行中のバックグラウンドのアクティビティについてユーザに警告する状況要素を追加することがよくあります。

Visualforce は、`<apex:facet>` タグを使った状況更新をサポートしています。このタグを使用すると、`sp` 属性によって、バックグラウンドのイベントの開始時または終了時にテキストを表示できます。また、さらに上級の開発者であれば、画像やその他のコンポーネントを表示することができます。

この例では、取引先責任者リストのページに、開発段階であることを示す状況テキストを追加します。ユーザが取引先責任者の名前をクリックすると、詳細領域が表示される間、その領域に [要求中...] というテキストが表示されます。

メッセージを実装するには、`<apex:facet name="stop">` を非同期的に更新されるコンポーネントの周りにラップします。2つのタグの間に、「`stop`」という名前のコンポーネントを追加します。

`facet` は、コンポーネントに示されるデータに関するコンテキスト情報を提供する、Visualforce コンポーネント内の1つの領域のコンテンツで構成されます。たとえば、`tbl p` はテーブルのヘッダー、`tbl f` は列のヘッダーまたはフッターの facet のみをサポートします。`tbl f` コンポーネントを使用すると、Visualforce コンポーネントのデフォルトの facet を独自のコンテンツで上書きできます。facet の開始タグと終了タグ内で使用できるのは1つの子のみです。



メモ:

すべてのコンポーネントが facet をサポートしているわけではありません。facet をサポートしているコンポーネントは「[標準のコンポーネントの参照](#)」に記載されています。

次の例では、`<apex:facet name="stop">` は、アクションが完了するとすぐに表示されるコンポーネントを含む、「stop」という facet (この例では詳細領域) をサポートします。

```

n   tegndprd por h&n C   au   acc   a   a   =   "   >
te   Belgo p   d   p   cll <&e   k   ra   r   > a   i   mH $="   U   !   s   &   s   N   a!   >
our   dnt   ltn   e   tg   p   rcYot   emn   &ns   diij   ja   ac   m   sau   cc   !   a   .   a   a   .
t ltn   e   &Cec   &w   k   mo   e   t   ro   rhd   &h   '   av   i   i   s   i   sa   .
B   lgo   p   p   c   <   k/a   a   >

```

x x te tB l g@ ptnlp c o < kCa c a i ='' s@ >
e x x te e ro m p f < a >
e x x te e ed{ bl plt u nt Th ou)cc t a t@var c a! al dd c saly ='' a " a
e x x n lo u m@ < a >
{ t tre o } c mc ! N a
e x x e n r e { mp mb t ltn@ oa} daca =bi "v a! ia " > /
e x x n no dmm @ < /a L ia >
e x x d bl p T /a a a a>
ex x ro m p f < /a >
xt te nou u b p < a Pa

```
e x t te nou u lp p < a Pa>
e xt re te dt upre e eopeum en itar dSr vl l=" sv " =! ia " > i
e xt x e n r e {mp mφ t ltn&c oa} daca =bi "v a! !
{ t tne o } c mc ! a a
e xt te nou u lp p < /a a Pa>
e x n lo u mp < /a a a a>
ex ro m p f < /a a >
xt B l go p p c < k/a a >
xt neu u lp d dp l< a Pa =! i "a>
xt t tn teto tpeae r t < n uaqi g S sas a =! R si .".>
e ex te nt p fm o < pa a =as " >
e ext e dt l e b p tej n eret ac gei &rm d&t rld cd Pal a a s if " L a s
ext t tn o p a < /a ia S sa>
xt te nou u lp p < /a a Pa>
g p p < /a a
```

ページを保存したら、任意の取引先責任者にマウスポインタを移動し、クリックしなくても詳細領域が適切な情報で更新されることを確認します。

関連リンク

[Visualforce ページでの JavaScript の使用](#)

第4章

Visualforce ページの外観と HTML 出力のカスタマイズ

Visualforce ページおよびコンポーネントは、表示するブラウザに送信される HTML を出力します。Visualforce の HTML 生成機能は高度であり、自動的にページ構造、コンテンツ、スタイルを提供します。また、Visualforce には、Visualforce のデフォルトの HTML を変更し、ページを独自の HTML や、CSS スタイルシートまたは JavaScript ファイルなどの関連付けられた追加リソースに置き換える方法も多数用意されています。

- 個々のコンポーネントにカスタムスタイルを添付したり、ページに独自のスタイルやスタイルシートを含めたりして、[Visualforce ページのスタイルをカスタマイズ](#)できます。これにより、ページ上の個々の要素のデザインをカスタマイズできます。
- [Visualforce ページの「doctype」\(ドキュメント種別または DTD\) を変更](#)できます。これは、HTML5 を使用する場合に特に便利で、ブラウザの互換性問題について対処できる場合もあります。
- ブラウザで別の出力処理方法を強制するには、[Visualforce ページの特定のコンテンツタイプを設定](#)できます。たとえば、ブラウザウィンドウに表示する代わりに CSV ファイルとしてダウンロードする場合にこの機能を使用できます。
- [生成される HTML に「パススルー」するカスタム属性を Visualforce コンポーネントに添付](#)できます。この機能は、たとえば jQuery Mobile や Knockout.js などの JavaScript フレームワークと併用するために、ページ要素に 属性を添付する場合に便利です。
- [Visualforce ページで生成される HTML の自動「整理」を緩和](#)して、HTML5 アプリケーションを作成しやすくすることができます。
- 自分で追加する静的 HTML タグのために、[ページでの <html> および <body> タグの生成を手動で無効化](#)することができます。

Visualforce ページのスタイル設定

Visualforce ページのスタイルは、標準の Salesforce ページのデザインを模倣するか、独自のスタイルシートやコンテンツタイプを使用して容易に設定できます。

多くの Visualforce コンポーネントには `style` または `class` 属性が含まれています。¹ このいずれかの属性を定義すると、CSS コードをコンポーネントに関連付けることができます。カスタム CSS コードを使用すると、幅、高さ、色、フォントなど、コンポーネントのデフォルトのビジュアルスタイルを変更できます。

Salesforce スタイルの使用

詳細ページの関連リストやセクションヘッダーなど、多くの Visualforce コンポーネントのデザインは、Salesforce の同じコンポーネントと共にしています。配色など、こうしたコンポーネントのスタイルの一部は、コンポーネントが表示されるタブに基づいています。コンポーネントのスタイル設定に使用するタブスタイルを指定するに

は、ページを標準コントローラに関連付けるか、`exg p p` または `get p ap da` タグの `ka` 属性を設定します。

- Visualforce ページに標準コントローラを使用する場合、新しいページには、関連付けられたオブジェクトの Salesforce での標準タブのスタイルが適用されます。これにより、関連オブジェクトに関連付けられたメソッドやレコードにアクセスすることもできます。
- カスタムコントローラを使用する場合、`g p b p` タグの `< ka` 属性により、関連付けられた Salesforce ページのデザインを模倣できます。ページを部分的に Salesforce ページに似せる場合、`l go p p c` タグの `ka` 属性を使用できます。[getter メソッドの定義](#) (ページ 107) の例を参照してください。

Salesforce ユーザインターフェースのカスタマイズの詳細は、Salesforce ヘルプの「ユーザインターフェース設定のカスタマイズ」を参照してください。

Salesforce スタイルの拡張

ページにスタイルシートを追加するには、`p h` タグを使用します。スタイルシートのスタイル定義に Visualforce コンポーネントを接続するには、これらのほとんどのコンポーネントで使用できる `c` 属性または `l` 属性を使用します。このように、独自の Salesforce スタイルを拡張できます。

次のマークアップは非常に基本的なページを示します。`p h` タグは、[設定] の [開発] > [静的リソース] で TestStyles という名前の静的リソースとして保存された、CSS スタイルシートを参照します。これは、`l e p e h le` タグの `as` 属性の `> cv` グローバル変数によって参照されます。`u p p` タグの `c` 属性は、スタイルシートに定義されたサンプルスタイルクラスを使用しています。

```
g p p < a >
t e {l e p d a t<et ou as }yslc w !R s .s Sys" > /
txe ou et e l tp l<the aet lv m el a" l by elc il s a dC sm y ss'pas y ss=as
g p p < /a >
```

この例に使用されているスタイルシートは、次のとおりです。

```
1 m p s a i- ;
```

ユーザに表示する Salesforce スタイルの識別

Visualforce ページを作成する場合、期待される Salesforce のデザインを知っておくと、そのスタイルにマッチしたページを表示するのに役立ちます。たとえば、ユーザがデザインをカスタマイズするかどうか選択できるとします。Visualforce ページは、その違いを考慮して設計する必要があります。

ユーザに表示するスタイルを識別するには、`m Ih$` と `U mUsIh$d.` `T U p U` という 2 つのグローバル変数が役立ちます。これら 2 つの変数は、`m Ih$` がユーザに表示すべきデザインを返し、`ml Ih$d T U p U` が実際のデザインを返すという点で異なります。たとえば、ユーザは [新しいユーザインターフェースのテーマ] のデザインを表示するよう設定された権限を持っている場合がありますが、そのデザインをサポートしていないブラウザを使用していると、Internet Explorer 6 などで `ml Ih$d T U p Us` が異なる値を返す場合があります。

どちらの変数も、次の値のいずれかを返します。

`m h` — 古い Salesforce テーマ

`m h` — Spring '10 以前に使用されていた Salesforce テーマ

`t ro l` • `lu f` — Salesforce カスタマー ポータルのテーマ

`b ro•` — Salesforce AppExchange のテーマ

Spring '10 で導入された、現在の Salesforce テーマ

開発者が Salesforce に似た CSS スタイルをハードコードしたとします。Visualforce ページで新しいスタイルについても同じデザインを維持するために、開発者はユーザの設定に合わせるために複数のスタイルシートの中から選択する必要があります。次の例に、それを実現する方法の 1 つを示します。

```
t      n t egndprd por id An C   au acc a a   " " >
x e     e r bln epr e   wnl un Ma i ava ja " v k   a"   i SO"
e {n er d r de re e eml Ih$ T   p Us m h D Ti s ya= ' 3 " >
x ee    t e {l p le h e < et as o uys et in c US L! R E Q } R s   c y Sys , ' Sys ss " "
x e     r bl p < /a i a a>
e {n er d r de re e eml Ih$ T   p Us m h D Ti s ya= ' 3 " >
x ee    t e {l p le h e < et as o uys et in c US L! R E Q } R s   c y Sys , ' Sys ss " "
x e     r bl p < /a i a a>
te nene o u G d g   ! -- i a $ >-
g p p < /a a
```

この例では、次の点に留意してください。

- 表示するセクションを `<h1>` に替えるには 属性を使用する。
- `<h1>` タグには `as sys` 属性がないため、この属性があるコンポーネントでラップする必要がある。

ユーザ向けに新しいデザインを有効にしても、ユーザがそれを表示するための適切なブラウザやアクセシビリティの設定を行っていない可能性があります。次のコード例では、`m Ih$` 変数を使用して代替情報をユーザに表示します。

```
e ew   et g o ptd pth r e< b la   C a"   "   sa y=" s " a>
xe e e e   te p gM r e e <fm ro ea d r es   se ei   =y   m Ih$   T   ! UsT .   = ' 3   '&
et n t et e d w n ceu hn ledov ad 'ni   blfka ro tn o z ro f ag i s   a   y   i ai a .
o t r en ou et n cdH v eg   et yell efa ' l w ac a a i sch ic i h ak B a   i
t n u d mre ro o bln fty epo de so t an f ih s m   ps   i si   i .
exe e   g p opM < /a ass a
x et wet e e p e {n   et d p zedai s i erse   y=" sm Ih$   T   ! UsT .   = ' 3   '&
g p p < /a a
```

`m Ih$` は `m Ih$` と同じですが、`m Ih$` は異なります。そのため、ページは最大限まで表示されません。

Salesforce スタイルシートの使用

 **警告:** Salesforce スタイルシートはバージョン管理されておらず、コンポーネントの外観やクラス名は予告なしに変更される可能性があります。Salesforce.com では、Salesforce スタイルシートを直接参照し、それに依存するのではなく、Salesforce スタイルのデザインを模倣した Visualforce コンポーネントを使用することをお勧めします。

Salesforce では、アプリケーション全体でさまざまなスタイルシート (.css ファイル) を使用して、すべてのタブを Salesforce のデザインに準拠させています。これらのスタイルシートは、`ew g p po` タグの `<h1>` 属性に `style` を指定しない限り、Visualforce ページに自動的に含まれます。

Salesforce スタイルシートが含まれないようにすると、独自のカスタムスタイルシートのみがページのスタイルに影響します。Salesforce のデザインに部分的または完全にマッチするスタイルを構築するという目的では、デフォルトのスタイルシートから選択したコンテンツを参照し、使用することをお勧めします。

次のスタイルシートには、参照可能なスタイルクラスが含まれています。これらは、salesforce.com インスタンス
の c ディレクトリにあります。[ss /](#)

c - 標準のオブジェクトとタブのスタイル定義の大半が含まれる。

c - カスタムタブのスタイル定義が含まれる。

重要: Salesforce.com は、組み込みスタイルの変更を通知したり、マニュアルを提供したりしていません。

HTML を作成するすべての Visualforce コンポーネントには、**css** 属性があります。これらによって、独自のスタイルとスタイルクラスを使用して HTML タグのデザインを制御できます。

たとえば次のコードは **u p p** のクラスを設定し、スタイルを適用します。

```
<tx>
<tx>
<tx>
<tx>
```

DOM ID を使用してスタイルを適用する場合、スタイル定義に CSS 属性セレクタを使用する必要があります。属性セレクタは、HTML タグではなく属性の定義を利用して CSS スタイルを適用します。任意の Visualforce コンポーネントの **id** 属性を設定できますが、場合によってはその先頭に親コンポーネントの **id** が付加されます。たとえば、次のコードの **id** は **m_i_0** です。

```
<tx>
<tx>
<tx>
```

CSS にこれを反映するには、属性セレクタを使用します。

```
<tx>
<tx>
<tx>
```

スタイルシートで画像を使用する場合は、画像を CSS ファイルと一緒に zip に圧縮して、1 つの静的リソースとしてアップロードします。たとえば、CSS ファイルに次のような行があるとします。

```
background-color: #000000;
```

注意: ディレクトリ全体と親 CSS ファイルを 1 つの zip ファイルに追加します。たとえば、zip ファイルリソース名が **style.css** の場合、次のように参照します。

```
style="background-color: #000000; font-size: 14px; color: white; border-radius: 50%; width: 40px; height: 40px; display: flex; align-items: center; justify-content: center; margin-right: 10px;">
```



警告: スタイルシートの **style** 属性が空の文字列である場合、そのページを PDF として表示することはできません。たとえば、スタイルルール **background-color: #000000; font-size: 14px; color: white; border-radius: 50%; width: 40px; height: 40px; display: flex; align-items: center; justify-content: center; margin-right: 10px;** がページに含まれる場合、そのページは PDF として表示されません。

コンポーネントの DOM ID を使用したスタイルの定義

DOM ID を使用してスタイルを適用する場合、スタイル定義に CSS 属性セレクタを使用します。属性セレクタは、HTML タグではなく属性の定義を利用して CSS スタイルを適用します。どの Visualforce コンポーネントでも **id** 属性をその DOM ID に設定できます。ただし、表示される HTML には通常、Visualforce の自

動 ID 生成プロセスの一環として、親コンポーネントの `key` が先頭に付加されます。たとえば、次のコードの実際の HTML は `key="1001"` になります。

CSS にこれを反映するには、属性セレクタを使用します。

et e t tx g p p < a a
 { d et n mt o I] * <c s y y" ssy >
 l < d ih E y i- ;
 x t et tx ou u ep d pd m <T w I a n hi =T yv fc a" i si s ya ay > /
 g p p < /a a

このセレクタでは、ID 内のどこかに "myId" が含まれる DOM ID が照合されるため、Visualforce コンポーネントで設定する `id` をスタイル設定に使用する場合は、その ID がページ上で一意である必要があります。

HTMLコメントとIE条件付きコメント

Visualforce は、内容を処理することなく、表示前にほとんどの HTML コメントと XML コメントをページから削除します。ただし、Internet Explorer 条件付きコメントは表示されるため、IE 固有のリソースおよびメタタグを含めることができます。

Internet Explorer の条件付きコメントは、一般に古いバージョンの IE との間で発生するブラウザの互換性問題に 対処するために使用されます。条件付きコメントはページ上のどこで使用されても動作しますが、ページの `<h1>` タグの内側に配置されることがよくあり、その場合は、バージョン固有のスタイルシートまたは JavaScript 互換性の「shim」を含めるために使用できます。

ページの `h` タグの内側に条件付きコメントを配置するには、標準の Salesforce ヘッダー、サイドバー、およびスタイルシートを無効にし、`d` 独自の `h b do` タグと `<` タグを追加します。

e t d go p ep dc m< h o adtr ha lnt y< dtfrd & \$100 a'l h s'a s f a a Sys s=" s'a >
e et B d h < >a
st ee t e { l l < ! --s as ys >-
t t et e tr tx c t p jpr < sic py" v / saia "
{ r e ce w & tr o tsc US L h & R sp bjo la s } i ian h y, s ' s/ i -si- .
e t e n et le e r l et le t tx k h< i p" s cys " y" ss"
e { re e h ef w & tr o t c US Le h & R sp ld cs l } i iai cy, ssi' / - s - ys .s
n d]f < ! il >-
e e n et le e r l et le t tx k h< i p" s cys " y" ss"
e { re e h ef w & tr o t c US Le h & R sp . ds } i iai c y, ssi' / s - ys .ss
n d]f < ! il >-
d h < / >a
b do < / >a
e w B t o rh o in l c < p > hs i iai < 1 y >
t t n t o t ipj e bo nj et < n d I s r u s a . sp ' va a < . >/
b do < / >a
g p p < / a >

Visualforce では、標準 HTML コメント内の `<rp>` などの Visualforce タグをサポートせず、評価もしません。ただし、IE 条件付きコメント内の次の表記は評価します。

α u[•] r c や 関数 などのダローバル変数
• U L R F Q ()

条件付きコメントの使用方法の詳細は、Internet Explorer の条件付きコメントに関する Microsoft のドキュメントを参照してください。

Visualforce で追加または変更される HTML タグ

デフォルトでは、結果が確実に有効な HTML(および XML) ドキュメントとなるようにするために、必要な HTML タグが Visualforce によってページに自動的に追加されます。

この自動的な動作を使用するページでは、Visualforce により、比較的単純な `E` 要求コンテキスト (ページが最初に読み込まれて表示されるとき) と、`x` `C` コンテキスト (`O` `MP` `S` `f` `K` が返送されるときや、`>` `O` `p` `c` `xxx` タグを使用して Ajax 要求が行われるときなど) の 2 つのコンテキストで、HTML タグが追加されます。

E コンテキストでは、Visualforce で表示される HTML はやや緩和されています。ページをラップする `h` タグや、ページのタイトルおよび `e` `px` `ehn` または `dusrpye` `>` `<` を使用してページに `S` `>` 追加されたスタイルシートまたはスクリプトをラップする `h` タグ、ページのコンテンツをラップする `ab` タグが追加されます。

他の Visualforce タグで生成される HTML は完全で有効な HTML になり、無効な静的 XML を含む Visualforce ページは保存できません。ただし、コントローラメソッドにアクセスする式によって追加された HTML、sObject 項目、他の非 Visualforce ソースなどは、それが返される前に Visualforce で検証されません。このため、E 要求 T を介して無効な XML ドキュメントが返される可能性があります。

□ コンテキストでは、Visualforce はより厳密です。場合によっては要求のコンテンツを既存の DOM に挿入する必要になるため、応答 HTML は有効になるように後処理されます。この「整理」により、欠落したタグや片方しかないタグの修復、無効なタグや属性の削除、それ以外にも無効な HTML のクリーンアップが行われ、コンテンツがクリーンな状態で返送先のページの DOM に挿入されます。この動作の目的は、
□ `<div>` などの既存の DOM を更新するタグが確実に動作することです。

HTML5 文書型の整理の緩和

問題の原因となる、HTML5 アプリケーションでのデフォルトの HTML の整理を緩和するには、
「html-5.0」に設定し、API バージョンを 28.0 以降に設定します。

API バージョン 28.0 以降では、@ コンテキストについて `LOCK_MODE` - を使用した Visualforce ! 5 0 ページの整理動作が変更され、HTML5 タグと属性が削除されなくなりました。Visualforce では常に、すべてのページで保存時に XML が正確であるかどうかが検証され、ページの XML 形式が適切であることが要求されますが、後処理での整理では @ 要求の不明なタグまたは属性は削除されなくなりました。これにより、HTML 属性を幅広く使用する HTML5 および JavaScript フレームワークの操作が非常に簡単になります。

最新のブラウザは独自の整理を非常に効果的に行いますが、有効なマークアップの表示に比べて動作の一貫性に欠けます。`h1 - h6` モードでの HTML の整理の削減はセーフティネットの減少を表しますが、その代わりに柔軟性が大幅に向上します。この緩和された整理モードは、このモードが必要な HTML5 ページでのみ、HTML 検証およびデバッグツールと共に使用することをお勧めします。



メモ: API バージョン 28.0 以降では、ページに対する `c` の判定方法の範囲が異なります。
x e n l du p c を使用して子ページをルートページに追加する場合、階層内のいずれかのページが
t d o l c m h - に設定され、ルートページがAPI バージョン 28.0 以降に設定されていると、ペー
t ジ階層全体が h - モードで表示されます。. 5 0

<html> および<body> タグの自動生成の手動による無効化

t g p b タグの `<m pop Bl` および `H pa y` 属性を使用して、`yh bydo` および `<` タグの自動 <
生成を抑制し、手動で静的マークアップをページに追加します。

次の例で、この方法を示します。

```
eg ope a p h l < d bta l h t ee dt ed e s 'a s =l a s 'a s f a a Sys s=" s 'a
l m h < >
b do d r h < a>
tn t or n h u C o <1 h > ai ass1!
d r h < / a>
r l c < i a >
e tle g o o T p o m < > ik s aM lsT p iH L < ! 5 >/
r l c < / i/a >
l m h < / >
g p p < / a >
```

これらの属性の動作は互いに独立して機能し、`f` 、`f` 、設定なしを組み合わせて使用できます。両方の属性がデフォルトの `m` に設定されると、`h b do` および `<` タグの自動生成が保持されます。どちらかが `f` に設定されると、対応するタグをマークアップに自分で追加する必要があります。このモードでは、ユーザが無効なタグの組み合わせや、最新のブラウザでも処理できない属性を作成するのを Visualforce で防止できません。

1 メモ: `m pop Bl` および `H pa y` の値に関係なく、`yh y` セクションは必要に応じて常に生
e 成されます。たとえば `<du erp c st ee <t` または `p ish >` タグを使用したり、ペー
l ジの `e` を設定したりすると、`h` タグが生成されます。>a

ただし、これには 1 つの例外があります。 `m pop` が `T f` に設定され、ページに `s a`
x e n t l du r p c c e < 以外の要素がない場合、`>h` は生成されません。たとえば、次のコードで
b do は `タグは自動的に追加されますが、h セクションは追加されません。`

```
eg ope d p h tl < el al m h &" Tis 'a H a y =l a s 'a >
```

属性は、ページについては `f` に設定する必要があります (例: `g p p < a`
`f).g" s "a >`

属性は "text/html" (デフォルト) に設定する必要があります。

まり最も外側の `g p p` タグの値が使用されます `du p c` タグを使用して追加さ `>`
上の `m pp Bl および H pa y` 属性は無視されます。 `y a`

カスタム doctype の使用

デフォルトでは、Visualforce ページは、HTML 4.01 Transitional という doctype で配信されます。特に、ページはこの doctype 命令で開始します。

B C E T M T D I C ! O H t L n R U L r n M P " T o l - B / D H / L E A O S Á a N Y /

Visualforce ページに異なる `doctype` を指定するには、`g` `pd` `φ` タグの `< c` 属性を使用します。
属性は、ドキュメント種別を表す文字列を取ります。文字列の形式は次のとおりです。

各項目は次のとあります。

tx・ fc m は_{txt} h のいずれか
o・ d o は c で有効な10進数のバージョン番号
・ は次のいずれか(含まれる場合)
◊ すべての_{txt} ドキュメント種別と h ドキ_{txt}
r または f sa

無効なドキュメント種別が指定されている場合、デフォルトの `doctype` が使用されます。有効な HTML `doctype` の詳細は [W3C Web サイトに掲載されているリストを参照してください](#)。

メモ: Sumatra '13 では、ページの `c` の判定方法の範囲が異なります。 `du` `p` `c` タグを使
用してページをメインページに追加する場合、階層内のいずれかのページが `lc` `m` `lt` - に設
定されていると、ページ階層全体がそのモードで表示されます。

カスタム doctype の例

XHTML 1.0 Strict ドキュメント種別の Visualforce ページを作成するには、[g](#) [pd](#) [φ](#) タグの [< c](#) 属性を [`](#) 使用し、[hr](#) [c](#) という値を指定します。

t̪d̪ go p̪ t̪p̪ t̪d̪ t̪s̪ ph̪r t̪a t̪ la c̪ry̪ l̪l̪ -cs̪t̪ " M̪ T̪" i̪s̪ H̪ Y̪ "
 d̪r h̪ l̪ e̪ d̪b̪fr l̪s̪ f̪" s̪'a̪ s̪ =a̪ s̪'a̪ >
 h̪ rh̪ T̪ c̪<1 M̪ i̪h̪s̪ i̪h̪ K̪ < 1! >
 p̪ < >
 b̪ m̪ m̪ tol̪ oe̪ otc̪ g̪l̪ br̪ c̪s̪ : y̪ s̪a̪ < y̪ > /
 n̪d̪i̪ p̪n̪ m̪g̪o <te̪ oc̪á gl̪p̪na f̪hif̪o A̪ o/ -s̪c̪i̪ " =a̪ s̪p̪i̪ " > /
 q̪ p̪ p̪ < /a̪ >



メモ: Visualforce では、doctype と一致させるためにコンポーネントが生成したマークアップを変更したり、ヘッダー や サイドバー のような標準の Salesforce 要素のマークアップを変更したりすることはありません。Salesforce 要素は、大半の doctype で有効であり、どの doctype でも適切に機能しますが、厳格な doctype を選択して HTML 検証 テストを通過させる場合、標準の Salesforce 要素の停止か置き換えが必要になる場合があります。

カスタム ContentType の使用

Visualforce ページに異なる形式を指定するには、`text-align` タグの属性を使用します。これは、

- 応答の HTTP ヘッダーをページの 属性の値に設定します。
属性は、`x-content-type-options` など、Multipurpose Internet Mail Extension (MIME) メディアタイプを値として取ります。無効な MIME メディアタイプを設定すると、ブラウザが予期しない動作をすることがあります。有効な MIME メディアタイプの詳細は、<http://www.iana.org/assignments/media-types/> を参照してください。

Microsoft Excel ContentType の例

Visualforce ページデータを Microsoft Excel スプレッドシートに表示するには、`encoding="UTF-8"` タグの `contentType` 属性を使用し、`content-type: application/vnd.ms-excel` という値を指定します。

たとえば、次のページは取引先責任者の簡単なリストを作成します。これは、「[ページでのデータのテーブルの作成](#)」(ページ 41)で示した例の簡易版です。

n tegndprd por ld An C au acc a a = " " >
 te B lgæ ptñlp co < kCa c a i = " sa >
 e B el(go þbl p lt u t Tnkoai joac t trñvar c æ! a c sav = a " a>
 x e ñ lo u t tnkoai o 4 c amc v æ! a " a > /
 x e ñ lo u t tnku o tñ d ac qñ Gæ! a.i ia i "y > /
 x e ñ lo u t tnue on <} co a c hv æ! a. P " > /
 e B l go þbl p c < T k/a a a >
 B l go p p c < k/a a >
 g p p < / a æ



メモ: このページに取引先データを表示するには、有効な取引先レコードの ID をページの URL のクエリパラメータとして指定する必要があります。次に例を示します。

h e : p **Salesforce instance** m /exp x bJ / a y Eal? 00 000 s

レコードの ID の取得についての詳細は、Visualforce での項目値の表示 (ページ 20) を参照してください。

このページを Excel で表示するには、次のように [属性](#) をタグに追加します。

n tegndprd por ld AnC au acc a **contentType="application/vnd.ms-excel"**
te TB lg@ ptnlp co < kCa c a i =" sa >
e B el(go pbl p l@ u st Trkoau joac t trñar c @" ! a .c sa"v = a " a>
x e h lo ut mpo o 4 c amc v @" ! AL 'a > /
x e h lo ut mpo o tr d ac gw @" ! ai ia i "y > /
x e h lo ut mpo on <} co ac hv @" ! a p " > /
e B l go pbl p c < T k/a a a>
B l go p p c < k/a a >
g p p < /a >

Excel でページが正しく表示されない場合、

など、別の MIME タイプを試してみてください)。

Visualforce コンポーネントのカスタム HTML 属性の設定

任意の属性を多くの Visualforce コンポーネントに追加し、表示される HTML に「パススルー」することができます。たとえば、Visualforce と jQuery Mobile、AngularJS、Knockout などの JavaScript フレームワークを併用するとき、またはその他の属性をフレームワーク関数を有効化するフックとして使用する場合に、この機能は便利です。また、`[ゴースト](#) テキスト、[クライアント側検証](#)、[ヘルプテキスト](#) 属性などの HTML5 機能の使いやすさを向上させるためにも使用できます。

パススルー属性をコンポーネントなどに追加するには、属性に「html-」のプレフィックスを付け、通常どおりに属性値を設定します。

```
eg op dtph lnst< ee dtfd el& et a'l h tsd& s f a ac Sym ps=" s 'a y="
nou tu pl qu bl< o a c htmladata-role="panel" html-data-id="menu"
en r n p m m < á s =a " > /
nou u p p < /a Pa>
nou tu pl qu bl< o a c htmladata-role="panel" html-data-id="main"
n r n p m m < á s =a i "a > /
nou u p p < /a Pa>
g p p < /a a
```

これにより、次のような HTML 出力が作成されます。

```
C E T M TD ! O M L P >
h < >
h d h < >a<.. />a
data-id="#menu" data-role="panel"
o n o m & ! -- s >-
< & />

data-id="#main" data-role="panel"
o n o m & ! -- s i a >-
< & />
l m h < />
```

「html-」で始まる属性はすべて、「html-」を削除して、表示される HTML にパススルーされます。



メモ: コンポーネントの組み込みの属性と競合するパススルー属性を使用すると、コンパイルエラーが発生します。

パススルー属性は、次の Visualforce コンポーネントでサポートされています。

```
n lo u mp < a >
t oBndmm pc o < a a >
n no dmm pc < k L ia >
t n n o om pc p < a >
e d b l p T< a a a a>
ro m p f < a >
r m p f < a a
m g p < a a
t l du rp c c < p a i S >
t n u p p < a >
```

```

et n    • u   b o  p c <     ká           >
et n    • u ldp   p <     á   É   >
t e  • n u dd p   p <     á H   i   >
et e t  • u r p   pc <     á           S   >
et n tx • u   p   p <     á           >
et n ex • u   p   p <     á           a> a
e      • m   p   <     a ss   sa>
et     c u u l b   p <     a   É   >
te     c u u b b   p <     a L   a>
t     nc u u p   p <     ka L   i   >
te     nc u u b   p <     a           Pa>
et tx c u u p   p <T a   >
      • g   p   p <     a   à
      B • l go p   p c <     k a   a   >
ttB • lBgo p u p c <     k a   a   s >
e     tB • l g a p   p o c & k a   a   s   >
e     tB e l g a p   p o c & m k a   a   s   >
e     B • l go p b l p   c <     T k a   a   a>
n     B l r p   p <     a   a   >a
at    B l r p   p m I <     a   a   a   >
n     • l r d p   p <     a   aiG   >
t e en • o   p rc   <     as   É   a>
et e lx  pb oc K c <     as   >s >
t t l  p   c   <     as L   i s   >
t t l n  p o c   <     as   Oi   >
t t l n  p o c   <     as   Oi   s >
t l d p o c   <     as   R i   a>
ee t  • l   p   h   <     as   ys   >
b   p   <     a   >a
n b   l p   <     a   a Pa>

```

表示された HTML へのパスルート属性の追加場所など、個々のコンポーネントについての詳細は、「[標準のコンポーネントの参照](#)」(ページ 288)を参照してください。

パスルート属性をサポートするコンポーネントを使用して生成できない HTML マークアップを作成するには、Visualforce タグと静的 HTML を組み合わせます。たとえば、jQuery Mobile の `<div>` を作成するには、`<div>` と `<div>` タグと必要な HTML タグを組み合わせます。

```

x lud et r low t e t ne < t   e rta uad" i w i r" r   iñ a = " "   á a = " " >
x t e { r e l p te p o < m è ta mvr ñ a = ! ñ i s o s "v = "ia " >
ex e { l e t e h È m< m È l a = "# > !i N < a < >/á >
ex t r p p < /a >a
lu

```

パスルート属性は、動的 Visualforce ではサポートされません。

HTML5 manifest 属性を使用したオフラインキャッシュ

`g p p` タグの `<manifest>` 属性を使用して、ページの重要なリソースをオフラインでキャッシュするための HTML5 キャッシュマニフェストを設定できます。

`f` 属性の値は、生成された HTML に渡されます。例:

```
eg ope dph l & d bta lat ee dt d e $'a $' =l a s f a a Sys s=" s "a
e lct ne h m e at y=" pGt5 M a sf =" / a / a i as " >
d r h < a>
tn t or n h u C o <1 h > ai as1! >
d r h < / a>
r l c < i a >
t lehoo T po m < iks aM ksT p iH L < ! 5 >/
r l c < i/a >
g p p < / a >
```

タグは次のようにになります。

```
g p p < / a >
```

`f` 属性は、API バージョン 28.0 以降に設定された Visualforce ページの `g p p` タグで使用できます。また、`m app u` が `T` (デフォルト) に設定されている必要があります。

Visualforce を使用して、ページのキャッシュマニフェストを指定できます。たとえば、上記の例で参照される `Cdn M f` ページは、次のようにになります。

```
t en ne tgotpe pc e < g p p / a > i as " H a y =" a s "a
net ee dtrd el e ew lh e so fda ha Sys s=" f s "a H a" s "a >
A C C E M I E N F S
d l m h c i .
l mlg o o g g p i s ys ss
r n cm jp s i sa / .
g p p < / a >
```

第5章

標準コントローラ

Visualforce コントローラは、関連付けられた Visualforce マークアップで指定されたコンポーネントをユーザが操作(ボタンやリンクのクリックなど)したときの動作を指定する命令のセットです。コントローラを使用すると、ページに表示されるデータにアクセスでき、また、コンポーネントの動作を変更できます。

Force.com プラットフォームでは、標準の Salesforce ページで使用されるものと同じ機能とロジックを持つ多くの標準コントローラが提供されます。たとえば、標準取引先コントローラを使用する場合、Visualforce ページで [保存] ボタンをクリックした場合、標準の取引先編集ページで [保存] をクリックした場合と同じ動作が行われます。

標準コントローラは、Force.com API を使用してクエリできる Salesforce オブジェクトのすべてに存在します。

標準コントローラの使用に関する詳細は、次のトピックを参照してください。

- Visualforce ページへの標準コントローラの関連付け
- 標準コントローラによるデータへのアクセス
- 標準コントローラアクションの使用
- 入力規則と標準コントローラ
- 標準コントローラを使用するページのスタイル設定
- オブジェクトのアクセシビリティの確認
- カスタムコントローラおよびコントローラ拡張

Visualforce ページへの標準コントローラの関連付け

標準コントローラを Visualforce ページに関連付けるには、`<apex:controler>` 属性を使 a a 用し、Force.com API を使用してクエリ可能な Salesforce オブジェクトの名前をそれに割り当てます。

たとえば、ページを `MyCustomObject` という名前のカスタムオブジェクトの標準コントローラに関連付けるには、次のマークアップを使用します。

```
tn t egndprd portlde C t au ob M Cj a a c c = ys o " _>
```

 **メモ:** `<apex:controler>` 属性を使用する場合、同時に `controller` 属性を使用することはできません。

標準コントローラによるデータへのアクセス

すべての標準コントローラには、ページ URL の クエリ文字列パラメータで指定されたレコードを返す getter メソッドが含まれます。このメソッドを使用すると、関連付けられたページマークアップが `object` 構文 (`object` はコントローラに関連付けられたオブジェクトの小文字の名前) を使用して、コンテキストレコードの項目を参照できます。たとえば、標準取引先コントローラを使用するページは、`cc_m` を使用して、`a` 現在コンテキストにある取引先の `項目の値` を返すことができます。



メモ: getter メソッドを正常に実行するには、URL の クエリ文字列パラメータで指定されたレコードが標準コントローラと同じ型である必要があります。たとえば、標準取引先コントローラを使用するページが返せるのは取引先レコードのみです。取引先責任者レコード ID が クエリ文字列パラメータで指定されている場合、`cc` 式ではデータは返されません。

Force.com API でのクエリと同様に、差し込み項目の構文を使用して関連するレコードからデータを取得できます。

- 子-親リレーションは、最大 5 レベルまでトラバースできます。たとえば、取引先責任者標準コントローラを使用している場合、`cc_m_a` (3 レベルの子-親リレーション) を使用して、その取引先責任者に関連付けられている取引先レコードの所有者名を返すことができます。
- 親-子リレーションは、1 レベルをトラバースできます。たとえば、標準取引先コントローラを使用している場合、`cc_c` を使用して、現在コンテキストにある取引先に関連付けられているすべての取引先責任者の配列を返すことができます。

標準コントローラアクションの使用

`action` メソッドは、ユーザがボタンをクリックしたり、ページ内のある領域にマウスポインタを移動したりするなどのページイベントが発生すると、ロジックまたはナビゲーションを実行します。次のいずれかのタグの `action` パラメータに 表記を使用することによって、ページマークアップから `action` メソッドをコールできます。

- `onBind` o < はアクションをコールするボタンを作成する
- `noBind` o < はアクションをコールするリンクを作成する
- `onLoad` o < は定期的にアクションをコールする
- `onUpdate` o < pp は、別の名前つきのコンポーネントにイベント (「`onclick`」、「`onmouseover`」など) を作成し、アクションをコールする
- `onClick` o c < は、アクションをコールする新しい JavaScript 関数を定義する
- g p p はページが読み込まれると、アクションをコールする

次の表に、すべての標準コントローラでサポートされる `action` メソッドの一覧を示します。これらのアクションは、`c` 属性が含まれる Visualforce コンポーネントに関連付けることができます。

アクション	説明
new	新規レコードを挿入するか、既存のレコードが現在コンテキストにある場合はそれを更新します。この操作が完了した後、アクションは、元のページ(わかつて)

アクション	説明
u q	i いる場合) にユーザを戻すか、保存したレコードの詳細ページにユーザを移動します。
c k e	i 新規レコードを挿入するか、既存のレコードが現在コンテキストにある場合はそれを更新します。 アクションとは異なり、このページはユーザを別のページにリダイレクトしません。
d e t	i 現在コンテキストにあるレコードの編集ページにユーザを移動します。この操作が完了した後、 アクションは、最初にユーザがアクションを呼び出したページにユーザを戻します。
d l	現在コンテキストにあるレコードを削除します。この操作が完了した後、 アクションは、ページを更新するか、関連付けられたオブジェクトのタブにユーザを移動します。
l c c	編集操作を中止します。この操作が完了した後、 c c アクションは、最初にユーザが編集を呼び出したページにユーザを戻します。
t tn o	i そのオブジェクトに対して最近使用されたリスト検索条件に基づいて、標準リストページの PageReference オブジェクトを返します。たとえば、標準コントローラが c c で、ユーザが表示した最近検索されたリストが New Last Week (先週に新規作成) の場合、先週作成された取引先責任者が表示されます。

たとえば、次のページでは取引先を更新できます。[保存] をクリックすると、 アクションが標準コントローラでトリガされ、取引先が更新されます。

```

t n tegndprd por hñ nC   au   acc a a      " " >
x ro m p f      <     a >
ex te tB l gøe pt l pne a e < o tkMado m al i      " y      "      " i " >
x ttB lbgo p u p c < k a a      s >
ex tit obnñmm pñno   e   a l u a      ia "   k   "   a" v   sa > /
ex tB lbgo p u p c < k/a a      s >
ex tb hegn p pø d leta ntk a nñ C aS locu " c y      2   s "      s=" " >
x etn e { u ldp ltepn n< ou ácc mñ v      ! a . " a > /
x etn e { u ldp ltepn < ou ácc ñ v      ! a $ " > /
x etn e { u ldp ltepn < ou ácc ñ v      p a! ! a . y " > /
x etn e { u ldp ltepn & eom ácou b wcm a! ! a . a N " > /
ex e tb l gø p po c <c k/a a s >
x B l go p p c < k/a a >
ro m p f < /a a >
g p p < /a a >

```



メモ: このページに取引先データを表示するには、有効な取引先レコードの ID をページの URL のエリパラメータとして指定する必要があります。次に例を示します。

```
h e :p Salesforce_instance m xp x bJ /a /y Real? 00 008 s
```

レコードの ID の取得についての詳細は、 [Visualforce での項目値の表示](#) (ページ 20)を参照してください。



メモ: 標準コントローラの q e、e ct 、 dklx または w アクションに関連付けられたコマンドボタンとリンクは、ユーザに適切な権限がある場合にのみ表示されます。同様に、特定のレコード

がページに関連付けられていない場合、`id` および `label` は表示されません。

アクションに関連付けられたコマンドボタンとリンクは表示されません。

入力規則と標準コントローラ

標準コントローラを使用する Visualforce ページにユーザがデータを入力し、そのデータが入力規則エラーになつた場合、エラーが Visualforce ページに表示されることがあります。入力規則エラーの場所がコンポーネントに関連付けられた項目の場合、エラーはそこに表示されます。入力規則エラーの場所がページ上部に設定されている場合は `executeScript` タグ内の属性 `action` または `onSuccess` コンポーネントを使用してエラーを表示します。

標準コントローラを使用するページのスタイル設定

標準コントローラに関連付けられたページは、指定されたオブジェクトに関連付けられた標準の Salesforce ページに使用されているスタイルを自動的に継承します。つまり、指定されたオブジェクトのタブが選択された状態で表示され、関連付けられたタブ色がすべてのページ要素のスタイルに使用されます。

標準コントローラを使用するページのスタイルは `apex:page` タグの `tabStyle` 属性を使用して上書きできます。たとえば、次のページでは標準取引先コントローラを使用しますが、[商談] タブを強調表示し、[商談] タブの黄色を使用するページを表示します。

```
<apex:page tabStyle="Account">
```

MyCustomObject に関連付けられたスタイルを使用するには、次のように指定します。

```
<apex:page tabStyle="MyCustomObject">
```

カスタム Visualforce タブに関連付けられたスタイルを使用するには、属性をタブ名(表示ラベルではない) + アンダースコア 2 個(_) + 単語「tab」に設定します。たとえば、名前が Source で表示ラベルが Sources の Visualforce タブのスタイルを使用するには、次の設定を使用します。

```
<apex:page tabStyle="Source_Sources">
```

または、標準コントローラページのスタイルを独自のカスタムスタイルシートとインラインスタイルで上書きできます。

関連リンク

[Visualforce ページのスタイル設定](#)

オブジェクトのアクセシビリティの確認

オブジェクトを表示するにはユーザの権限が不十分な場合、コントローラを使用してそのオブジェクトを表示する Visualforce ページにはアクセスできなくなります。このエラーを回避するには、Visualforce コンポーネントが、

ユーザがコントローラに関連付けられたオブジェクトに対するアクセス権を持つ場合にのみ表示されるようになります。

オブジェクトのアクセシビリティは次のように確認できます。

```
b j e oe $ objectnameO cc y . a $
```

この式は true または false の値を返します。

たとえば、標準の Lead オブジェクトへのアクセス権があるかどうかを確認する場合、次のコードを使用します。

```
e b e j e dc $} bl cc! o L y . a a $
```

カスタムオブジェクトの場合、コードは似ています。

```
b t t c ue $b M Cj ! o )c blc ycc ys o . a $
```

t u ob M Cj c c は、カスタムオブジェクトの名前です。

ユーザにオブジェクトへのアクセス権がある場合のみページの一部が表示されるようにするには、コンポーネントの render 属性を使用します。たとえば、ユーザに Lead オブジェクトへのアクセス権がある場合にページブロックを表示するには、次のように実行します。

```
t n tegnd rd por ld s C da a a L " " a
e e B { h go p d r t c e < b ek ja e adc $} bl = cc! o L y . a a " >
tx w hpl d T lee ateo ip se f ti di bni iya y c sa L p a < . >/
B l go p p c < k/a a >
g p p < /a a
```

ユーザがオブジェクトにアクセスできない場合に、代替メッセージを表示することをお勧めします。次に例を示します。

```
t n tegnd rd por ld s C da a a L " " a
e e B { h go p d r t c e < b ek ja e adc $} bl = cc! o L y . a a " >
tx w hpl d T lee ateo ip se f ti di bni iya y c sa L p a < . >/
B l go p p c < k/a a >
e e B h go p d r dc te < e k ae j ae T dc $} bl o cd! o L y . a a " >)
t ro he mepou eo ec d > bS hyt en ydc o atse te le a atcs a d yboh j v a c ss L p
B l go p p c < k/a a >
g p p < /a a
```

第6章

標準リストコントローラ

標準リストコントローラを使用すると、レコードセットの表示や操作が行える Visualforce ページを作成できます。レコードセットを使用する既存の Salesforce ページの例として、リストページ、関連リスト、一括アクションページなどがあります。標準リストコントローラは、次のオブジェクトで使用できます。

- Account
- Asset
- Campaign
- Case
- Contact
- Contract
- Idea
- Lead
- Opportunity
- Order
- Product2
- Solution
- User
- カスタムオブジェクト

標準リストコントローラの使用に関する詳細は、次のトピックを参照してください。

- Visualforce ページへの標準リストコントローラの関連付け
- リストコントローラによるデータへのアクセス
- 標準リストコントローラアクションの使用
- 標準リストコントローラによるリストビューの使用
- 標準リストコントローラを使用したタブの上書き
- 標準リストコントローラを使用したカスタムリストボタンの追加

関連リンク

[カスタムコントローラの作成](#)

Visualforce ページへの標準リストコントローラの関連付け

標準リストコントローラの使用方法は、標準コントローラの使用方法によく似ています。最初にコンポーネントの `or lā r C` 属性を設定し、次に同じコンポーネントの `rc` 属性を設定します。

たとえば、ページを取引先の標準リストコントローラに関連付けるには、次のマークアップを使用します。

```
<apex:page standardController="Account">
```

 メモ: `g p e m d` タグで `or lā r ac` と `属性を使用する場合` 同時に `c` 属性を使用することはできません。

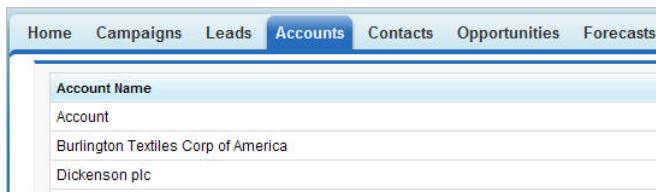
`rd` 属性は、ページがリストコントローラを使用することだけでなく、レコードコレクションの変数名も示すことができます。この変数は、レコードコレクションのデータにアクセスする場合に使用できます。

リストコントローラによるデータへのアクセス

ページをリストコントローラに関連付けたら、式言語の構文を使用してレコードのセットを参照できます。たとえば、取引先の単純なテーブルを作成するには、次のマークアップを使用してページを作成します。

```
<apex:page standardController="Account">
<apex:form>
<apex:repeat value="{!accounts}" index="i">
    <apex:outputText value="Name: {!accounts[i].Name}"/>
    <apex:outputText value="Phone: {!accounts[i].Phone}"/>
</apex:repeat>
</apex:form>
</apex:page>
```

これによって作成されたページには、組織のすべての取引先名のリストが表示されます。



 メモ: このページでは要求で検索条件が指定されないため、ページは最後に使用した検索条件で表示されます。リストコントローラでの検索条件の使用方法については、「[標準リストコントローラによるリストビューの使用](#)」(ページ 72)を参照してください。

Force.com API でのクエリと同様に、式言語の構文を使用して関連するレコードからデータを取得できます。標準コントローラと同様に、子-親リレーションは最大 5 レベルまで、親-子リレーションは 1 レベルをトラバースできます。

標準リストコントローラを使用する場合、返されるレコードは、現在のビューの定義に従って最初のデータ列を基準に並び替えられます。これは、最初のデータ列が表示されない場合でも同様です。拡張またはカスタムリストコントローラを使用する場合、並び替えメソッドを制御できます。



メモ: 標準リストコントローラによって返すことができるレコードは 10,000 個以下です。カスタムコントローラは、より大きな結果セットに使用できます。「[大量のデータセットを使用した作業](#)」(ページ 85)を参照してください。

標準リストコントローラアクションの使用

action メソッドは、ユーザがボタンをクリックしたり、ページ内のある領域にマウスポインタを移動したりするなどのページイベントが発生すると、ロジックまたはナビゲーションを実行します。次のいずれかのタグの { o c } パラメータに 表記を使用することによって、ページマークアップから action メソッドをコールできます。

button	• oBndmm pc o	< はアクションをコールするボタンを作成する
button	n no dmm pc	< はアクションをコールするリンクを作成する
button	• o ldp pc	< は定期的にアクションをコールする
button	n t • o upo	pp は、別の名前つきのコンポーネントにイベント(「onclick」、「onmouseover」など)を作成し、アクションをコールする
button	n • o upc o c <	は、アクションをコールする新しい JavaScript 関数を定義する
	• g p p	はページが読み込まれると、アクションをコールする

次の表では、すべての標準リストコントローラでサポートされる action メソッドについて説明します。これらのアクションは、c 属性が含まれる Visualforce コンポーネントに関連付けることができます。

アクション	説明
e	v 新しいレコードを挿入するか、変更された既存のレコードを更新します。この操作が完了した後、c アクションは、元のページ(わかっている場合)またはホームページにユーザを戻します。
c k	i 新しいレコードを挿入するか、変更された既存のレコードを更新します。c アクションとは異なり、c はユーザを別のページにリダイレクトしません。
e t	i1 ユーザが f I を指定していない場合は、そのオブジェクトに対して最近使用されたリスト検索条件に基づいて、標準リストページの PageReference オブジェクトを返します。
c c	編集操作を中止します。この操作が完了した後、c c アクションは、最初にユーザが編集を呼び出したページにユーザを戻します。
f	i セットにあるレコードの最初のページを表示します。
	i セットにあるレコードの最後のページを表示します。
	i セットにあるレコードの次のページを表示します。
o u p	v セットにあるレコードの前のページを表示します。

次の例では、ユーザが、取引先レコードを表示するための検索条件を指定します。ユーザが [Go] をクリックすると、選択された検索条件を使用して標準リストページが表示されます。

```
t      n tegndprd por hAnCe   au r accd at anc   o# cc " v     s =" a a s" >
e ex t ro m p f < a >
e ex t te l eptl cu e q rzd as fl i sv   a# li i " $1 = " >
e ex t t t len{ po etcl uwpl nasj ooi  s p a# li wi i s" > /
x      t t l p c < /as L is >
x      tm eBndmm w o tl{ u <no t ao } lca v   a# "G ia =" hi s" > /
x      ro m p f < /a >
g p p < /a >
```

リストコントローラによるページネーション

リストコントローラを使用してページにページネーションを追加するには、`r` および `p` アクションを利用します。たとえば、次のマークアップを使用してページを作成したとします。

```
t      n tegndprd por hAnCe   au r accd at anc   o# cc " v     s =" a a s" >
t      teB el go p lpm ct An lg au acc vi =" i i s" >
x      et ro d m p fro h < a i = " F " >
x      e tB l gn p poc < ka a is >
x      xt t d e { pr lt a net } adu iasc = " a! a ! a s" 1y" >
e      en } m ! .a a
x      xt t d p < /aL iasa >
ex      e tB l gn p poc < c k/a a is >
x      n l r d p lo pu m c < a2 aiG s=" " >
x      n mto{ dmne w o e <rc> ou l piaou ina =ho ! vdmn ws" >v ip< s /a ia >
x      n mto{ dmne w t m o t c xla L ma no idmne w ! " N< /a ia >
ex      n l r d p p < /a aiG >
x      ro m p f < /a >
B      l go p p c < k/a a >
g p p < /a >
```

デフォルトでは、リストコントローラはページに 20 レコードを返します。ページごとに表示するレコード数を制御するには、コントローラ拡張を使用して `p` を設定します。コントローラ拡張の詳細は、「[コントローラ拡張の作成](#)」(ページ 79)を参照してください。



メモ: ページネーションを使用する場合、コレクションに変更された行があると例外が発生します。これには、拡張アクションでコレクションに追加された新しい行も含まれます。この場合のエラーメッセージの処理は、標準動作に従って行われ、ページ上に表示できます。たとえば、`g p gpM < m` または `m g p` コンポーネントを使用してエラーメッセージをユーザに表示できます。

標準リストコントローラによるリストビューの使用

多くの Salesforce ページに含まれるリストビューを使用して、ページに表示されるレコードを絞り込むことができます。たとえば、商談ホームページでは、リストビュードロップダウンから「私の商談」を選択して、自分が所有する商談のみのリストを表示できます。リストコントローラに関連付けられたページでは、リストビューを使用することもできます。

たとえば、リストビューを使用して取引先の単純なリストを作成するには、次のマークアップを使用してページを作成します。

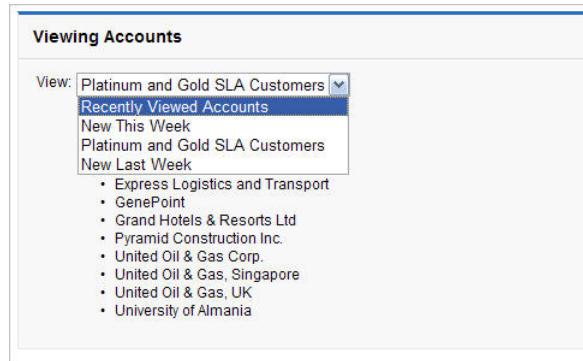
```
t      n tegndprd por hAnCe   au r accd at anc   o# cc " v     s =" a a s" >
t      teB el go p lpm ct An lg au acc vi =" i i s" >
```

```

x et ro d m p fro < a i = F " >
x n l r d p lo pu m c < a2 aiG s=" " >
x t eou eu b p w l < a L: vaV a" i " > /
ex t t e l ept l cu e & rzd as f il sv a" li " $1 = " >
x t ne te & upr oce ne en epp n ar g i da d r Sl v = " a = " i s " > /
e ex t t len{ po et cl uwpxl nasjo ooi vs p a" li vi i s " > /
e ex t t l r d p p < /as L is >
ex n l r d p p < /a aiG >
x e tB l ga p poc < k a a >
x t t d e { pr lt an } a a t i a s a l = " a v a a" ! a s" i = " i s " >
e n } m ! . a a
x t t d p < /aL i a s a >
ex e tB l ga p poc < c k/a a >
ro m p f < /a >
B l go p p c < k/a a >
g p p < /a a >

```

そのページを開くと、次のように表示されます。



このページは、標準取引先コントローラに関連付けられており、コンポーネントはコンポーネントによって入力されます。ユーザが表示できるリストビューが得られます。ユーザがドロップダウンリストから値を選択すると、その値はコントローラのプロパティにバインドされます。プロパティが変更されると、ページで使用できるレコードが変更されるため、が更新されると、その値を使用して、ページで使用できるレコードのリストが更新されます。

次のように編集ページでピューリストを使用することもできます。

```

t n tegndprd portld keda re ut s r p yad a et cn e ro u i "pp v s=" a i i s"
t et bt lt n ro u pp a Sy" O i " y
e d br l f < a s " a >
x ex ro m p f < a >
e exe e B l go p p c < k a a >
e ex B l go p p c < k a a >
e ex n l r d p lo pu m c < a2 aiG s=" " >
e xt eou eu b p w l < a L: vaV a" i " > /
e ex t t e l ept l cu e & rzd as f il sv a" li " $1 = " >
e ex ne te & upr oce ne en epp n ar g i da d r Sl v = " a = " i s " > /
e ex t t len{ po et cl uwpxl nasjo ooi vs p a" li vi i s " > /
e ex n l r d p p < /a aiG >
e ex B l go p p c < k/a a >
e ex tB l go p p c < k a a >
e ex tB l go p p c < k/a a >
e ex e B el{go pbl p tc aut< n kroa ju a ppr vate a" ! o hpi i s "pp = " a " i = "

```

```

e      x      e   n   lo u e h p un   o   <   m app v   a" !   .   "a > /
e      x      e   en do u endor l hu   <   g a v   a   a" !   S" a>
e      x      e   etn e { u ldp e   & } a g p v   a" !   s   Na   "a > /
e      x      n   lo u mpc   <   /a   >
e      x      e   en do u endrelthu l< o   aC v   a   a" Ds   "a>
e      x      e   etn e { u ldp d   & t & l jo a   o p v   a" !   .Ds   "a > /
e      x      n   lo u mpc   <   /a   >
e      ex      e   B   l go pbl p c   <   T k/a   a   a>
ex      B   l go p   p c   <   k/a   a   >
x      ro      m p f   <   /a   >
g      p      p   <   /a   >

```



メモ: ユーザがリストビューを変更した場合、コレクションに変更された行があると例外が発生します。この場合のエラーメッセージの処理は、標準動作に従って行われ、ページ上に表示できます。たとえば、`try { xpM < または m pass sa>` コンポーネントを使用してエラーメッセージをユーザに表示できます。

リストコントローラによるレコードの編集

リストコントローラを使用してレコードのセットを編集することもできます。たとえば、次のマークアップを使用してページを作成したとします。

```

t      n   tegndprd portld *eG   re   ut s r pxd at et cn e" ro   u i   "pp v   s =a   i i s"
bt      lt   ne   ro   u e   p p r   la Sy" O   f   i   " y$   =a   s 'a >
x      ro   m p f   <   a   >
ex      B   l go p   p c   <   k a   a   >
e      ex      e   g   p   gpM   <   a   ass   sa > /
e      ex      tB   lBgo p u p c   <   k a   a   s >
e      ex      tB   lBgo p u p c   <   k a   a   s >
e      ex      e   B   el{go pbl p tc aut< n kroa ju a   ppr va o   a" !   ppi i   s "v   =a   " >
e      x      e   n   lo u e h p un   o   <   m app v   a" !   .   "a > /
e      x      e   en do u endor l hu   <   g a v   a   a" !   S" a>
e      x      e   etn e { u ldp e   & } a g p v   a" !   s   Na   "a > /
e      x      n   lo u mpc   <   /a   >
e      x      e   en do u endrelthu l< o   aC v   a   a" Ds   "a>
e      x      e   etn e { u ldp d   & t & l jo a   o p v   a" !   .Ds   "a > /
e      x      n   lo u mpc   <   /a   >
e      ex      e   B   l go pbl p c   <   T k/a   a   a>
ex      B   l go p   p c   <   k/a   a   >
x      ro      m p f   <   /a   >
g      p      p   <   /a   >

```

次のような、商談の [フェーズ] と [完了予定日] を更新して保存できるページが表示されます。

Opportunity Name	Stage	Close Date
asdasdads	Closed Won	11/11/2009 [1/13/2010]
Burlington Textiles Weaving Plant Generator	Closed Won	8/5/2007 [1/13/2010]
Dickenson Mobile Generators	Qualification	8/5/2007 [1/13/2010]
Edge Emergency Generator	Closed Won	8/5/2007 [1/13/2010]
Edge Emergency Generator	Id. Decision Makers	8/5/2007 [1/13/2010]
Edge Installation	Closed Won	8/5/2007 [1/13/2010]
Edge SLA	Closed Won	8/5/2007 [1/13/2010]
express Logistics Portable Truck Generators	Value Proposition	8/5/2007 [1/13/2010] E
express Logistics SLA	Perception Analysis	8/5/2007 [1/13/2010] E
express Logistics Standby Generator	Closed Won	8/5/2007 [1/13/2010] G
enePoint Lab Generators	Id. Decision Makers	8/5/2007 [1/13/2010] G
enePoint SLA	Closed Won	8/5/2007 [1/13/2010] G
enePoint Standby Generator	Closed Won	8/5/2007 [1/13/2010]

詳細は、「カスタムリストコントローラによるレコードの一括更新」(ページ 125)を参照してください。



メモ: リストコントローラの **q** **t**、**c** **d**、または **a** アクションに関連付けられたコマンドボタンとリンクは、ユーザに適切な権限がない場合は表示されません。同様に、特定のレコードがページに関連付けられていない場合、**q** **t** アクションに関連付けられたコマンドボタンとリンクは表示されません。

第7章

カスタムコントローラおよびコントローラ拡張

標準コントローラは、標準ページに使用されているものと同一のロジックを含むため、Visualforce ページに必要なすべての機能を提供できます。たとえば、標準取引先コントローラを使用する場合、Visualforce ページで [保存] ボタンをクリックした場合、標準の取引先編集ページで [保存] をクリックした場合と同じ動作が行われます。

ただし、既存の機能の上書き、アプリケーションによるナビゲーションのカスタマイズ、コールアウトまたは Web サービスを使用する場合、またはページの情報にアクセスする方法についてより詳細な制御が必要な場合は、Apex を使用して、カスタムコントローラまたはコントローラ拡張を記述できます。

- ・ カスタムコントローラおよびコントローラ拡張とは?
- ・ カスタムコントローラの作成
- ・ コントローラ拡張の作成
- ・ コントローラメソッド
- ・ コントローラクラスのセキュリティ
- ・ カスタムコントローラおよびコントローラ拡張の作成に関する考慮事項
- ・ Visualforce ページ内の実行順序
- ・ カスタムコントローラおよびコントローラ拡張のテスト
- ・ 入力規則とカスタムコントローラ
- ・ キーワードの使用

nr

カスタムコントローラおよびコントローラ拡張とは?

カスタムコントローラは、標準コントローラを使用せずにページのすべてのロジックを実装する Apex クラスです。Visualforce ページを完全にシステムモードで実行する場合に、カスタムコントローラを使用します。システムモードでは現在のユーザの権限と項目レベルのセキュリティが適用されません。

コントローラ拡張は、標準コントローラまたはカスタムコントローラの機能を拡張する Apex クラスです。次の場合にコントローラ拡張を使用します。

- ・ 標準コントローラの組み込み機能を使用するが、編集、参照、または削除などの 1 つ以上のアクションを上書きする。
- ・ 新しいアクションを追加する。
- ・ ユーザ権限が適用される Visualforce ページを作成する。コントローラ拡張クラスはシステムモードで実行されますが、コントローラ拡張が標準コントローラを拡張する場合、標準コントローラのロジックは、システムモードで実行されません。代わりに、ユーザモードで実行され、現在のユーザの権限、項目レベルのセキュリティ、および共有ルールが適用されます。



メモ: カスタムコントローラとコントローラ拡張クラスはシステムモードで実行されるため、ユーザ権限や項目レベルのセキュリティを無視しますが、クラス定義に `h` `sh` キーワードを使用することによって、ユーザの組織の共有設定、ロール階層、および共有ルールを使用するかどうかを選択できます。詳細は、[Force.com Apex コード開発者ガイド](#)の「`th` `sh` または `r` `sh` `s` `sh` `a` キーワードの使用」を参照してください。

s i

カスタムコントローラの作成

カスタムコントローラは、引数をとらないデフォルトのコンストラクタを外部の最上位クラスに使用する Apex クラスです。パラメータを含むカスタムコントローラコンストラクタを作成することはできません。

カスタムコントローラを作成する手順は、次のとおりです。

1. [設定] で、[開発] > [Apex クラス] をクリックします。
2. [新規] をクリックします。
3. [バージョン設定] をクリックして、このクラスで使用する Apex および API のバージョンを指定します。組織が AppExchange から管理パッケージをインストールした場合、このクラスで使用する各管理パッケージのバージョンも指定できます。すべてのバージョンでデフォルト値を使用します。デフォルト値では、Apex および API についても、各管理パッケージについても、クラスを最新バージョンに関連付けます。最新バージョンのパッケージのものとは異なるコンポーネントや機能にアクセスする場合は、管理パッケージの古いバージョンを指定することもできます。特定の動作を維持するには、Apex および API の古いバージョンを指定できます。
4. クラスエディタで、クラスの Apex コードを入力します。1 つのクラスの長さは、最大 1,000,000 文字です。を使用して定義したコメント、テストメソッド、またはクラスは含みません。
5. [Save (保存)] をクリックし、変更を保存してクラスの詳細画面に戻るか、[Quick Save (適用)] をクリックし、変更を保存してクラスの編集を続行します。作成した Apex クラスは、クラスに保存する前に正しくコンパイルする必要があります。

次のクラスは、カスタムコントローラの単純な例です。

```
bl t el n pc or ld M C i ss a y
r n t A h p t onf cou vcc a a ;
t n t oln { or pl d M C i y ( )
ou e cc e td E EC m tT An =L ou S M cc N a S F R O
e ex d A e E Etel n ed grnt pHe e dR g e =gt rm Ba . gd Pa ( . ) ] Pa a s ( . ) i ( '
}
etblAt n rt rm ou cc i a ;
e e e e bie en { grc f c i Ra a a ( )
t d n ou p cc a a ;
n n ru lu ;
```

次の Visualforce マークアップは、上記のカスタムコントローラをページ内で使用する方法を示します。

```
x te n gor pl d mp c t set ld r aC b talAn e'u y cc " a Sy=" " >
ro m p f < a >
```

```

ex      teB l go ptnlpt {c or<@ukCao te a r i r }=" m $ai a! s 直 s N "a>
e      t n o b l toeAn g e ou x &cnm Y e {: u ldp lt&nN n<aoü ácc mä v a! a . "a>
e      ex      t@ oBndmm w@o < @ e a l u a ia=" k "ä" v "a> /
x      ex      B l go p p c < k/a a >
x      ro m p f < /a >
x      g p p < /a >

```

- e カスタムコントローラは、`g ptepn`コンポーネントの `c` 属性が使用されているため、ページと関連付けられています。

標準コントローラおよびコントローラ拡張と同様に、カスタムコントローラのメソッドは、関連付けられたページマークアップで表記を使って参照できます。上記の例では、`xcoet n` メソッドは `u ldp p <` タグの `e` 属性で参照されており、`Bandmm w@o n <` タグは `c` 属性が設定された `ia` メソッドを参照しています。



メモ: 他の Apex クラスと同様に、すべてのカスタムコントローラはシステムモードで実行されます。したがって、現在のユーザの資格情報はコントローラのロジックの実行に使用されていないため、ユーザの権限と項目レベルのセキュリティは適用されません。

w クラスの定義で `h g` キーワードを使用することによって、カスタムコントローラでユーザの組織の共有設定、ロール階層、および共有ルールを適用するかどうかを選択できます。詳細は、[Force.com Apex コード開発者ガイド](#)の「`th g` または `rih sghia` キーワードの使用」を参照してください。

カスタムコントローラは、新しいレコードを作成する場合にも使用できます。次に例を示します。

```

e blu w l nkt A p c de f dyl dr C ss N i si
e      blnxtA nptc coau te ct gr i } p a ; 立 w ;
w      blnxtA nptc coau te ct gr i } p a ; 立 w ;
e      eck da e It negrtp e dt gie gt rm Ba . gd Pa(.) Pa a s (.) i ( ; )
t      ou ndc w@li t An oau = ac( == : ) ? ( ) H R = i ;
e      e n EtECo thur tl An [S Nou a CCPd , E E IYR O W ] H R = i ;
}
e e e e ble en { grc f c i Ra w a ( )
e      t tu rn ou p cc y s ( a ; )
t      ext e x t e { d mn d M Ec pa (s@L . i )
e      ex A e e e gld p g M Ba . ass sa ( ; )
t      n ru li ;
}
e      e A ernt t et@ gteo dew//luhv Sa@ i g a p: vai ( a v i ) ; ( )
e      ne mn erx A t n t@ ndprd ortlde(C wou Bacc S a a
}

```

次の Visualforce マークアップは、上記のカスタムコントローラをページ内で使用する方法を示します。

```

x      te n e gompld ptcA de n at dyl dr C b=tNlAn i s@u cc " s a y" "
x      ro m p f < a >
ex      e B l go p do pm dc < ka a = i " >
e      ex      e g p gpM < a ass sa > /
e      ex      e B l go p po c < ka a & >
e      ex      e { u ldp lt@An n< oü ácc mä v ä! ! . "a > /
e      ex      e { u ldp lt@An < nou} áoc f hv p ä! ! . " > /
e      ex      e { u ldp lt@An tr ou &car F v ä! ! i. s "y > /
e      ex      e B l go p po c < c k/a a & >
e      ex      t@ lBgotp u p & o < o lk e o am s i a=" " >
e      ex      t@ eBndmm w@o lt@ < @ a o } a v ä! v Sa i a=" k "a > /

```

e	ex	tB	lBgo	p	u	p	c	<	k/a	a	s >
ex		B	l go	p	p	c	<	k/a	a	>	
x		ro	m p f		<		/a	/a			
		g	p p	<		/a	à				

コントローラ拡張の作成

e e コントローラ拡張は、`CustomControllerName` 型の单一の引数を取る
e e コンストラクタが含まれる Apex クラスです。`CustomControllerName` は、拡張するカスタムコントローラの名
e e 前です。

次のクラスは、コントローラ拡張の単純な例です。

bl t d n e ptx or l d r nc ó E ss a y s
r n t A m p ouf cc v cc a a a
tx n nt ntho ñ rem u netz Á te es r sh b mimi i i as v a
tr bl etabet uce gt v gina ta moar qish m fl rdR s a a
n or ld rc // .
t ñln e tx or pl a r n ex A & E in t gynd prdt ortla ñ{ Cd(or lar CS a a s
t t Am t e come dcor ldirs C. g=rd (cs) . R ;()
e bie t { p ñg rg g i is G i ()
e n r ru tld o n t mc H + 'd tc a . + a t ' a . ; ' '

次の Visualforce マークアップは、上記のコントローラ拡張をページ内で使用する方法を示します。

n t e g n d p r d p o r d d x n t C x a m n a c c t e a n a e t x o r t h r n C " o E s s = y \$ " >
n r } g g p ! i < > /
x et n e { u l d p l t e p n n < o u a c c m v a " ! p a . " a < > / > /
x t t e B n d m m e p o l t u < a a o } a v a " v S a i a = " k " a > /
ro m p f < / a >
g p p < / a a

拡張は、コンポーネントの属性を使用してページに関連付けられます。

すべてのコントローラメソッドと同様に、ページマークアップで表記を使用して、コントローラ拡張メソッドを参照できます。上記の例では、ページ上部の `g e` 式は、コントローラ拡張の `g` メソッドを参照しています。

この拡張は、Account 標準コントローラと共に機能するため、標準コントローラメソッドも利用できます。たとえば、`ue ldp p l` タグの `a` 属性は、標準コントローラ機能を使用して取引先の名前を取得します。同様に `OBndmm wto n <` タグは `a c e` 属性のある標準 Account の `メソッド` を参照します。

カンマ区切りのリストを使って、単一のページに対し複数のコントローラ拡張を定義できます。これにより、同じ名前のメソッドを上書きできます。たとえば、次のようなページがあったとします。

t n tegnd prd por hñ C au ecc a a " "
n netx ñx w e ew E o e ñ d s htst" o ,f " H a" s "a >
xtet txe ñou u p l up <ñ o a f v a" ! " > /
a p p < / a >

さらに、次の拡張があったとします。

```
keb(x l n pcc E i ss a o
etx e bln ex A tpc En t g ndprd {ior l& rO C }o Bac S a a a )
t e blnt { r p c g goo f i is ' F ( )
} ne r run oo o
```

```
bt(x wl pcc o E Ti ss a
tx ewbli ex A topc En t g ldprd {ior l& r C }o Bac S a a a )
t e blnt { r p c g goo f i is ' F ( )
} n r wru oo o
```

tx ou u p p e コンポーネントの値は、> f として表示されます。上書きは、一番左の拡張、つまりカンマ区切りリストの最初の拡張で定義されているメソッドによって定義されます。したがって、E の good x w メソッドは、E E のメソッドを上書きします。



メモ: 他の Apex クラスと同様に、コントローラ拡張はシステムモードで実行します。したがって、現在のユーザの資格情報はコントローラのロジックの実行に使用されていないため、ユーザの権限と項目レベルのセキュリティは適用されません。ただし、コントローラ拡張が標準コントローラを拡張する場合、標準コントローラのロジックはシステムモードで実行されません。代わりに、ユーザモードで実行されます。このモードでは現在のユーザの権限、項目レベルのセキュリティ、共有ルールが適用されます。

クラスの定義で h gn キーワードを使用することによって、コントローラ拡張でユーザの組織の共有設定、ロール階層、および共有ルールを適用するかを選択できます。詳細は、[Force.com Apex コード開発者ガイド](#)の「th gn または r h sgn ia キーワードの使用」を参照してください。

カスタムリストコントローラの作成

カスタムリストコントローラは、標準リストコントローラと似ています。カスタムリストコントローラには、レコードセットの表示や操作を行うために定義した Apex ロジックを実装できます。

たとえば、SOQL クエリに基づいて次のカスタムリストコントローラを作成できます。

```
blt t l t n pcc fo u n pp o i 2Gss a l iys
ex A t e n t g t edpr d etor l& rt/ctau mBa . S a adS s i s i a a
t n rot dtred fl or l& rc // s a a s s
ex biA t e n pcc g t edpr d t(or l& m C oBa . S @ a S s
e t n { n o fl hC i s( == )
e t e newn ex A t C e n t g t edpr d t e z ed ret C b Bat S g a a s ro D c ( as@ a L y
e e e e e t E mCo tT ct nL ro[S uMN apD s ] ER O O i y; )
}
t e tn r rm o C s ;
}
}
t blt ten pcc fo ut et p g ro l u i s p i y o i s ( )
ntr rut t en to eu etn pp o L q esd o ci y s ) . R s ;( )
```

 メモ: によって返される sObject のリストは、不变です。たとえば、それに対して c をコールすることはできません。リストに含まれる sObject に変更を適用することはできますが、リストそのものに項目を追加したり、リストから項目を削除したりすることはできません。

次の Visualforce マークアップは、上記のカスタムコントローラをページ内で使用する方法を示します。

```

x t e n      g o r p l d p c n & r s u n a pp a    2 C =''     i l i y s " >
x ex      B l go p p c < k a a >
e e x      e B el{go p b l p h c aut< n @ kroa }u a p p r v a o a" ! i i s " v ='' a" >
e e x      e h lo ue h p u }o < m a v a" ! N " a > /
ex      e B l go p b l p c < T k/a a a >
ex      B l go p p c < k/a a a >
g p p < / a a

```

また、SOQL クエリの一部として反結合および準結合を使用するカスタムリストコントローラを作成することもできます。次のコードは、標準取引先コントローラの拡張として実装されます。

```

blu n p r c b l g n t{ nou in acg s i a s s a i Pai a
r n t A n t p ou f cc v cc a a a
t nt e o r h u e t o l t c n t p e / n e d s e or l a n r s d s i d s f a a i
c n e wr m u h l o p k // a
t b l An t n o u p e x c g A d e n t i g t e l p d i t e r i m a r C o r l a n c S a a s )
t t A m t e n c e u t o x c l d r a s g r a = ( c ) . R ;( )
}

e x b l A t e n p t c g t e l p d o e t l d m { C o u B a . r d c a a s a R s
g e t n o u { f d c d h i ( a R s == )
e t n e o u e w n r d e x A d e n t g t e l p d s o z l d r C B a . S a a s (
t e e t b t g n o e r o D c a s e A n C L @ y M & C ( d [ S N E I E R O W I H R R
t t A n d u R d c d c l I r o L u e M ( S p p t l o o F R E O r C u W i H y ] R s s =
}

t n e r t r m o u r d c c a R s ;
e te t r p 立 s ;
} t b l t e A n t p c o t A n c d g f o u n l c c g < s o > i Pai a ( )
t n t r u t A n e t o u n e c c e t u r d c i s g r d > c ) a R s . R s ;( )
}

```

これらのレコードを表示するページでは、標準リストコントローラアクションを組み合わせて使用しますが、カスタムリストコントローラから返されるレコードの反復処理に基づいています。

```

t n t e g n d p d p o r h A n C e a u r a c c d a n c o " cc " v s ='' a a s "
x t A n t nou n c c g o $ s ='' i Pai a" >
x t B el go p l m c t A n k g e u a c c V i ='' i i s" >
e et ro d m p fro h < a i ='' F " >
e ex e t B l g a p p o c & k a a a s >
{ t e t e n } a c ! a . a
e ex e t B l g a p p o c < c k/a a a s >
e ex n l r d p l o p u m c < a2 a i G s ='' " >
e ex n m o { d m m e p o e < r c } o u h p i a o u i n a = " h o ! v i m m p s > v i p < s / k ia
e ex n m o { d m m e p o t e o t k c x h L m a n o i d h m = p ! " N < / k ia >
x ro m p f < / a a i G >
B l go p p c < k/a a a >
g p p < / a a

```

コントローラメソッド

Visualforce マークアップは、次の種別のコントローラ拡張とカスタムコントローラメソッドを使用できます。

- action
- getter
- setter

action メソッド

action メソッドは、ユーザがボタンをクリックしたり、ページ内のある領域にマウスポインタを移動したりするなどのページイベントが発生すると、ロジックまたはナビゲーションを実行します。次のいずれかのタグの { o c } パラメータに 表記を使用することによって、ページマークアップから action メソッドをコールできます。

七丸 • oBndmm p o < はアクションをコールするボタンを作成する
n n no dmm p < はアクションをコールするリンクを作成する
e n • o ldp p < は定期的にアクションをコールする
n t • o upr p &pp; は、別の名前つきのコンポーネントにイベント（「onclick」、「onmouseover」など）を作成し、アクションをコールする
丸 n • o upc o c < は、アクションをコールする新しいJavaScript 関数を定義する
• g p p はページが読み込まれると、アクションをコールする
たとえば、[カスタムコントローラの作成](#)（ページ 77）のサンプルページでは、oBndmm p o < タグの
o c パラメータによって、コントローラの action メソッドがコールされます。その他の action メソッドの例は、[「action メソッドの定義」](#)（ページ 109）を参照してください。

getter メソッド

getter メソッドはコントローラの値を返します。コントローラによって計算され、ページに表示される各値には、boolean 変数など、対応する getter メソッドが含まれる必要があります。たとえば、[カスタムコントローラの作成](#)（ページ 77）のサンプルページでは、コントローラに cc メソッドが含まれます。このメソッドによって、ページのマークアップは、 表記のあるコントローラクラスの cc メンバー変数を参照することができます。ue ldp p l タグの a パラメータは、この表記を使用して取引先にアクセスし、ドット表記を使用して取引先の名前を表示します。getter メソッドの名前は、 Variable にする必要があります。

重要: getter メソッドを 等にする、つまり副次的影響がないようにすることがベツ顛ズ害垂だ 鮎。Óペ fi à。寇

ストの setter メソッドを明示的にはコールしませんが、ユーザがコマンドボタンをクリックすると、action メソッドの前に setter メソッドが実行されます。

```

t e n   e t g o r p d e p n c < o r l a h r a C a = " " >
e x   e B l g o p d o p m d c < d b l k a a c = " i k " i = " " >
e x   e t B l g a p p o c & k a a s >
e ex   e t B e l g a p p o c & m k a a s >
e x t   t e o a u u h e l p t e x p r e f t e a d L x t T e a d h = " s o u T a b p > S p a < / a L a>
e ex   n l r o p p < p a a G >
e x e t n t x e u e p d x e p t e k e l d a u t x T r j i = " s d a " v T a " ! s a " > /
e x t t t e B n d m m p o l { u < a o d c p r v a " d h i a = " ! S a " >
e ex   e e e n r r d t r b l & t u c u k = " s s a " s s a > /
e ex   n l r o p p < p / a a G >
e ex   e t B e l g a p p o c < c m k / a a s >
x t t t n & t p a d e t t e x e r a t i a r n S u s a q = " T s g d a s a = " s i . . . > /
e ex   e t B l g a p p o c l & t d a r l u i n l o u = " R s s " i = " s s " l s = " " >
e ex   e B e l { g o e p b l p t c u < r ) t k l u a x l v a a " ! s s " v = " a " >
e x   e n r d r d e t r ) l u T = " N U L I N S ( s s " ) >
e x   e n l o u e m p u l } < m a v a " ! . " a > /
e x   e n l o u e m p u l } & m a v a " ! . i " a > /
e x   e n l o u e m p u n l } & h a p v a " ! . " > /
e ex   e B l g o p b l p c < T k / a a a >
e ex   e t B l g a p p o c < c k / a a a >
x ro m p f < / a a >
g p p < / a a >

```

次のクラスは、上記のページマークアップに使用するコントローラです。

```

e t b l h t l e n p c c o r l a h r C i s s a
n e r t x g d h T i s s a ; 
e t d r l u L i k s > a s s ;
t e k d t e r t p x g g r d h i T s s a ( )
e n r t x r d h T s a ; 
e e b l h e d t x p t c r { n r d h i g T i s a = s ; s a ( i s s )
e t x r d h T s a = s ; s a ( i s s )
e t b l e e t t p c g l u L i k s > a R s s ( )
e n t r u r l u e d o f c i d R a s a ( )
t r e t l u e d e t x r s E s L = i d R a s a ( )
n m r u l u ; 

```

getter メソッドは、コントローラから値にアクセスする場合に必ず必要なメソッドですが、値をコントローラに渡す場合、必ずしも setter メソッドを含める必要はありません。Visualforce コンポーネントが、コントローラに保存されている sObject にバインドされている場合、sObject が対応する action メソッドによって保存または更新されるかぎり、sObject 項目はユーザによって変更されると自動的に設定されます。この動作の例は、[カスタムコントローラの作成](#) (ページ 77) のサンプルページにあります。

e setter メソッドの名前は、variable にする必要があります。



重要: setter メソッドを繰り返す、つまり副次的影響がないようにすることがベストプラクティスです。たとえば、変数の増分、ログメッセージの書き込み、データベースへの新規レコードの追加を行わないようになります。Visualforce では、要求の処理の過程でコールされる可能性のある setter メソッドのコール順序および回数を定義しません。1つのページリクエストでの setter メソッドのコール回数に関係なく、同じ結果を生成するように setter メソッドをデザインしてください。

カスタム拡張またはコントローラを使用したデータの取得と設定

Apex メソッドと変数がコントローラ拡張またはカスタムコントローラによって処理される順序は保証されません。このため、コントローラと拡張クラスは、実行している別のメソッドに依存するのではなく、直接そのメソッドをコールする必要があります。これは、変数の設定とデータベースのデータへのアクセスに特に当てはまります。

たとえば、次のカスタムコントローラの最初のメソッド `do_c_M_h_1` は、contact 变数 `c` がすでに存在することを前提としているため、正しい値を常に返します。一方、`do_c_M_h_2` は、正しい値を返すことがあります、`c` がまだ設定されていない場合は正しい値を返すとは限りません。

```
blu { h Bo c d c i V ss a s a
      o C c c a;
      t etabl t toe tn ptc Qgo c do á M h a a1 ( )
      n l h f c e c d i E EtG= a T o #LM C[S c IMI ER Ø L a 1 ;
      n r ru c
      t etabl t toe tn ptc Qgo c do á M h 2 a a ( )
      n r ru c ;
```

次のカスタムコントローラにはまったく同じメソッドがあります。ただし、`do_c_M_h_2` は `c` をコールするため、変数 `c` は常に設定され、返されるときには正しい値が常に含まれます。

```
blu { h p o c do c i V ss a s G
      o C c c a;
      t etabl t toe tn ptc Qgo c do á M h a a1 ( )
      n l h f e c d i E EtG= a T o #LM C[S c IMI ER Ø L a 1 ;
      n r ru c
      t etabl t toe tn ptc Qgo c do á M h 2 a a1 a ;( )
      n r t e t m t go c M h a1 ;( )
```

次のマークアップでは、これらのコントローラをコールする 2 つのページを示します。Visualforce マークアップは同じものであり、コントローラの名前のみが変更されています。

```
t e n gorplä pc o < do c a a V =! s G" >
t e t n t g o t c t M h ⊕ 2 :n c do} c M h b m 2 ( )! a . < a > /
t e t n t g o t c t M h ⊕ :n c do} c M h m ( )! a1 . a
g p p < / a à
```

```
t e n gorplä pc Bo < d c a a V =! s " à
t e t n t g o t c t M h ⊕ 2 :n c do} c M h b m 2 ( )! a . < a > /
t e t n t g o t c t M h ⊕ :n c do} c M h m ( )! a1 . a
g p p < / a à
```

コントローラクラスのセキュリティ

他の Apex クラスと同様、ユーザが自分のプロファイルに基づいてメソッドをカスタムコントローラまたはコントローラ拡張クラスで実行できるかどうかを指定できます。



メモ: 組織に管理パッケージをインストールした場合、セキュリティの設定は、パッケージ内の `Apex` クラス、または `c` と宣言されたメソッドを含むクラスに対してのみ設定できます。

ユーザに「Apex 開発」権限がある場合、そのユーザには、個々のクラスのセキュリティ設定に関係なく、関連する組織のすべての Apex クラスへのアクセス権があります。

Apex クラスに対する権限は、最上位でのみ確認されます。たとえば、クラス A がクラス B を呼び出す場合、ユーザプロファイルにクラス A へのアクセス権のみがあってクラス B へのアクセス権はなくても、ユーザはクラス A のコードを実行することができます。同様に、Visualforce ページでカスタムコンポーネントと関連コントローラが併用されている場合は、そのページと関連付けられているコントローラに対してのみセキュリティがチェックされます。カスタムコンポーネントと関連付けられているコントローラは、権限に関係なく実行されます。

クラス一覧ページから Apex クラスのセキュリティを設定する手順は、次のとおりです。

1. [設定] で、[開発] > [Apex クラス] をクリックします。
2. 制限するクラス名の横にある [セキュリティ] をクリックします。
3. [選択可能なプロファイル] リストから有効にするプロファイルを選択して [追加] をクリックするか、[有効にされたプロファイル] リストから無効にするプロファイルを選択して [削除] をクリックします。
4. [保存] をクリックします。

クラスの詳細ページから Apex クラスのセキュリティを設定する手順は、次のとおりです。

1. [設定] で、[開発] > [Apex クラス] をクリックします。
2. 制限するクラス名をクリックします。
3. [セキュリティ] をクリックします。
4. [選択可能なプロファイル] リストから有効にするプロファイルを選択して [追加] をクリックするか、[有効にされたプロファイル] リストから無効にするプロファイルを選択して [削除] をクリックします。
5. [保存] をクリックします。

関連リンク

[Apex および Visualforce 開発のセキュリティのヒント](#)

大量のデータセットを使用した作業

Visualforce カスタムコントローラおよびコントローラ拡張には、Apex ガバナ制限が適用されます。ガバナ制限についての詳細は、「[実行ガバナと制限について](#)」(ページ 63)を参照してください。また、`loop` `for` `in` `list` `size` `1000` などの Visualforce 反復コンポーネントは、反復を行うコレクションの項目数が最大 1,000 項目に制限されています。

Visualforce ページでは、(カスタムレポーティングおよびカスタム分析を行うときなどに) これより大きなデータのセットの処理または表示が必要になることがあります、そのデータを変更する必要はありません。Visualforce が開発者のために用意している「参照のみモード」を使用すると、1つの要求でクエリできる行数の制限を緩和し、ページ内で反復できるコレクションの項目数に関する制限を引き上げることができます。

参照のみモードは、ページ全体、または特定の制限付きで個別のコンポーネントまたはメソッドに対して指定できます。



メモ: ページ全体に参照のみモードを指定する場合、大きなデータセットのみを反復処理することができます。

関連リンク

[ページ全体での参照のみモードの設定](#)

[コントローラメソッドでの参照のみモードの設定](#)

ページ全体での参照のみモードの設定

ページ全体で参照のみモードを有効にするには、`repeat` コンポーネントの `list` 属性を `list="true"` に設定します。

たとえば、次の例は、参照のみモードで処理される単純なページです。

```
ten gorplätiptc tæmme aor läretgr d="1" Sr uay Ss a " Oa ="y " >
t tr t{ e p e k Htc D g $ mms ai si v ll ypa s ay< S a >/
g p p < /a a
```

このページのコントローラも単純ではありますが、ページに表示する概要統計を計算できる方法を示しています。

```
ble lte t pter fm or läri C ss a s ay Ss a
e et able t e rpg I g trt gr mm V L y a s ay S a )
c n e rt g todit t ro u pp s o i y Ss a
} e n r rut l todit t ro u pp s o i y Ss a
```

通常、単一の Visualforce ページ要求のクエリでは 50,000 行を超える行数を取得できないことがあります。参照のみモードでは、この制限が緩和され、最大 1,000,000 行をクエリできます。

属性は、より多くの行をクエリできるようになるだけでなく、`list="true"` および `repeat` などのコンポーネントを使用して反復処理できるコレクション内の最大項目数も引き上げることができます。この制限は、1,000 項目から 10,000 項目に増加しました。次の例は、これを示す単純なコントローラとページです。

```
ble el n ptecd M dr läri C ss a i a
et ble e n ep c tde M dn Ae lg i<sd M dn i a > i a ( )
e n rete d M dn n rL c dks M dn i a > i a = i a ( )
e e EEC E nc E c [S M NM dia,i P c FRI T i a L _ 1 000
} e te n r reu n r dh M dn ( i a ; )
```

```

t e n e   g o r p l a p a   t e < n d e   M a d r l d a r e t o r   d   "   r   u i s   "   "   O a = " y   "   >
e t e   e d(   l l e p e n   m   m < H   d p   i s   h a :   p i s   v < a   >/
x t e   e d{   b l e   p l   u A e   l < r   } da   M t h a v a r   d o e " u   !   p c   i a   " v   = " a   "   >
x e   e x t e   e n e   p   f n t d e r   r e d o   t   a   a   = " a   p   f   " a   >   R   / a   a >
e e   e x t e t   t x e   f u   u   p   l t e p   r < d o   u   a   p m c   v   a   !   N   " a   >   /
x x   n   l o   u   m p x   <   a   >
x x   n   l o   u   m p x   <   / a   >
e e   e x t e   e n e   e p   f n e d   r   e x r t   a   c a   = " b   f   " a   > i < P   / a   a >
e e   e x t e t   t x e   f u   u   p   l t e p   r < d o   u   r a   p c   v   c   c a   !   . i   P   "   >   /
x x   n   l o   u   m p x   <   / a   >
x t e   d   b l   p   &   / a   a   a   a >
g   p   p   <   / a   a

```

ページ全体に参照のみモードを使用する Visualforce ページではデータ操作言語(DML)操作を使用できませんが、ページのフォームやその他のユーザインターフェース要素に影響する getter、setter、および action メソッドのコール、その他の参照のみクエリの実行などを実行できます。

コントローラメソッドでの参照のみモードの設定

Visualforce コントローラメソッドでは、一部の重要な制限付きで、ページ自体が参照のみモードでなくとも Apex アノテーションを使用できます。

アノテーションを使用した Visualforce コントローラメソッドは、自動的に参照のみモードを使用します。ただし、Visualforce コントローラメソッドのアノテーションの制限のため、参照のみメソッドにもも○○アノテーションが必要となります。○○アノテーションでは、次のメソッドであることが必要です。

l b o d • b l または p c a i
• c s i a

アノテーションを使用した参照のみモードの有効化は、トップレベルメソッドのコールで行う必要があります。トップレベルメソッドのコールにアノテーションが存在しないと、セカンダリメソッドにアノテーションが存在する場合でも、クエリされる最大行数に関する通常の制限が要求全体に適用されます。

e @ コントローラメソッドでアノテーションを使用すると、Visualforce 式の結果としてレコードの大きなコレクションを取得できます。ただし、反復コンポーネントのコレクションの最大項目数が引き上げられることはありません。より大きな結果のコレクションを反復処理する場合は、ページ全体で参照のみモードを有効にする必要があります。

関連リンク

[ページ全体での参照のみモードの設定](#)

カスタムコントローラおよびコントローラ拡張の作成に関する考慮事項

以下に示すのは、コントローラ拡張およびカスタムコントローラの作成時における考慮事項です。

- として定義されたメソッドがクラスにない限り、カスタム拡張とコントローラクラスおよびコントローラメソッドは、通常として定義されます。i クラスに Web サービスマソッドが含まれる場合、として定義されている必要があります。
- データベースからデータを返すときは sets、maps、または lists を使用します。こうすることで、コードとデータベース間の通信回数を削減できるため、コードの効率が高まります。
- Visualforce コントローラ拡張とカスタムコントローラの Apex ガバナ制限は、匿名ブロックまたは WSDL メソッドの制限と同じです。ガバナ制限についての詳細は、付録の「[実行ガバナと制限について](#)」を参照してください。
- カスタムコントローラまたはコントローラ拡張を作成する場合、通常はユーザに表示されない機密データが不注意で公開されないように注意してください。権限を適用するには、クラス定義に hgh キーワードを使用することをお勧めします。また、Web サービスの使用に注意してください。プロファイルによって最上位のエントリポイントとして確保されていますが、一旦初期化されるとシステムのコンテキストで実行されます。
- Apex メソッドと変数のインスタンス化の順序は、必ずしも指定されたとおりではありません。詳細は、「[カスタム拡張またはコントローラを使用したデータの取得と設定](#)」(ページ 84)を参照してください。
- コントローラの「getxxx」メソッドでは、データ操作言語 (DML) の操作を使用できません。たとえば、コントローラに n メソッドが含まれている場合、オブジェクトを作成するメソッドで d または p i s を使用することはできません。
- コントローラのコンストラクタメソッドでデータ操作言語 (DML) の操作を使用することはできません。
- コントローラ、またはコントローラのコンストラクタでは、「getxxx」または「setxxx」メソッドに f アノテーションを使用することはできません。
- string 型または integer 型などの Apex のプリミティブデータ型は、値によってコンポーネントのコントローラに渡されます。
- list や sObject などの Apex の非プリミティブデータ型は、参照によってコンポーネントのコントローラに渡されます。つまり、コンポーネントのコントローラによって取引先の名前が変更されると、その変更内容はページのコントローラで使用できるようになります。
- 個人取引先を使用する組織の場合は、次の事項が適用されます。
 - ◊ カスタムコントローラを使用する取引先レコードの n 項目を u ldp p a コンポーネントを 使用して参照する場合、t クエリに ou cc を指定する必要があります。
 - ◊ 新しい取引先を作成して m を設定すると、レコードは法人取引先になります。新しい取引先を作成して m を設定すると、個人取引先になります。
 - ◊ ベストプラクティスとして、個人取引先と法人取引先の両方で正しく表示されるカスタム名数式項目を作成し、Visualforce ページで、標準項目の代わりにその項目を使用します。
 - ◊ Force.com AppExchange パッケージに Visualforce ページを含める場合は、コントローラまたはコントローラ拡張で、個人取引先にのみ存在する項目を明示的に参照することはできません。

Visualforce ページ内の実行順序

ユーザが Visualforce ページを参照すると、ページに関連付けられたコントローラ、拡張、およびコンポーネントのインスタンスがサーバによって作成されます。これらの要素が実行される順序によって、ユーザに表示されるページの状態が変化することがあります。

Visualforce ページの要素の実行順序を理解するには、はじめに、ページのライフサイクル、つまり、ユーザセッションにおいてページがどのように作成されて処分されるのかを理解しておく必要があります。ページのライフ

サイクルは、ページのコンテンツだけではなく、ページの要求方法によっても決定されます。Visualforce ページの要求には 2 つの種類があります。

- *get* 要求は、ユーザが URL を入力したとき、またはユーザを新しいページに移動するリンクやボタンがクリックされたときに発行される最初のページ要求です。
- *postback* 要求は、[保存] ボタンをクリックし、*save* アクションをトリガするなど、ユーザの操作によってページの更新が必要になった場合に発行されます。

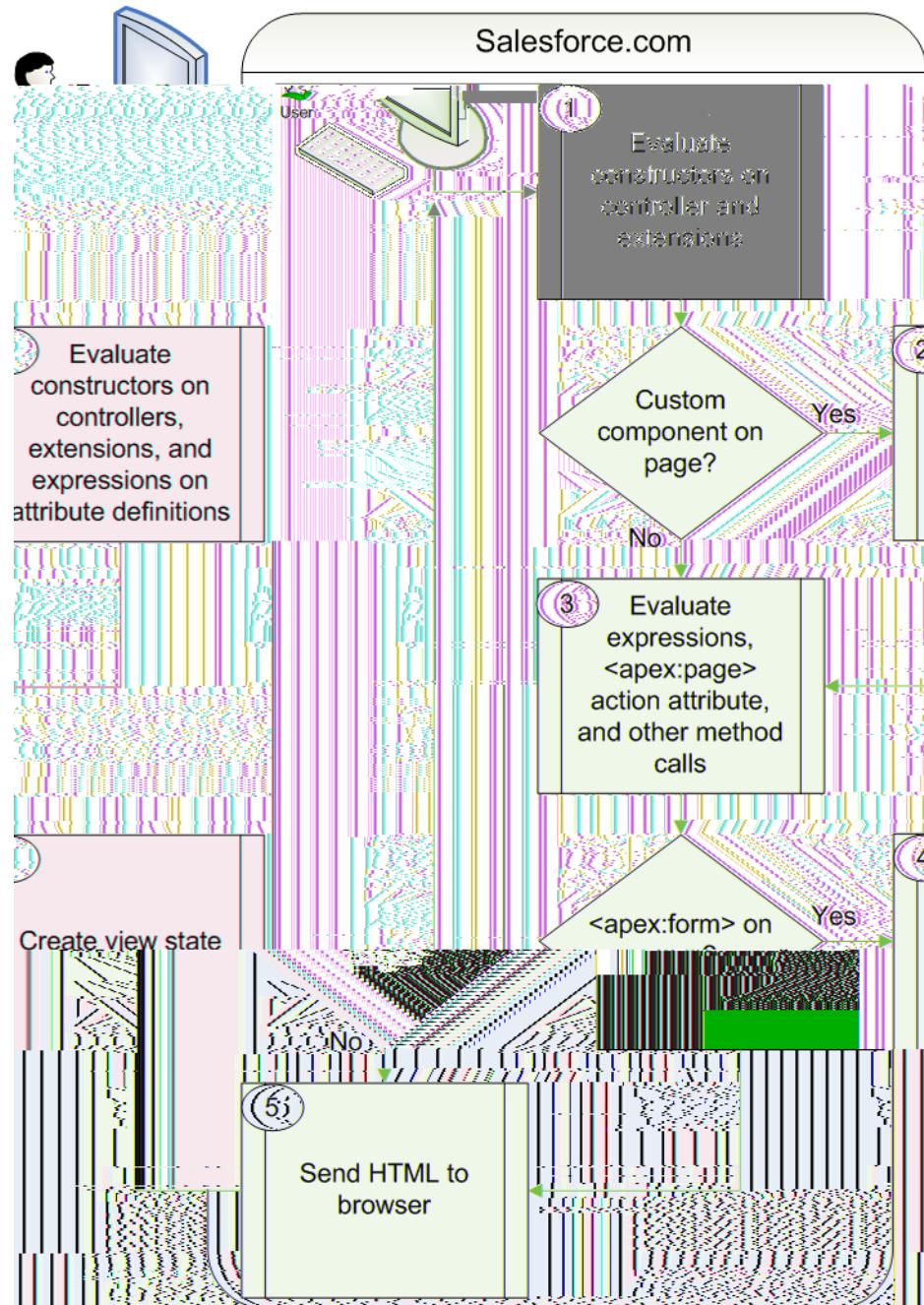
2 種類の要求の具体的な詳細、ページのライフサイクルを説明した例、および独自のカスタムコントローラおよびコントローラ拡張を記述するときの実行順序のヒントについては、次を参照してください。

- Visualforce ページの *get* 要求の実行順序
- Visualforce ページの *postback* 要求の実行順序
- Visualforce ページの実行順序の例

 メモ: Visualforce ページ要求の最大レスポンスサイズは、15 MB 未満である必要があります。

Visualforce ページの *get* 要求の実行順序

get 要求は、ユーザが URL を入力したとき、またはユーザを新しいページに移動するリンクやボタンがクリックされたときに発行される最初のページ要求です。次の図は *get* 要求時の、Visualforce ページとコントローラ拡張またはカスタムコントローラクラスとのやり取りを示します。



上の図では、はじめにユーザが URL を入力するか、リンクまたはボタンをクリックするかのいずれかの操作を行ってページを要求します。この最初のページ要求は *get* 要求と呼ばれます。

- 関連するカスタムコントローラまたはコントローラ拡張クラスのコンストラクタメソッドがコールされ、コントローラオブジェクトがインスタンス化されます。
- ページにカスタムコンポーネントが含まれる場合、そのコンポーネントが作成され、関連するカスタムコントローラまたはコントローラ拡張のコンストラクタメソッドが実行されます。式を使用して属性がカスタムコンポーネントに設定されている場合、その式はコンストラクタが評価された後に評価されます。
- ページは、その後、そのページにあるすべてのカスタムコンポーネントのすべての属性を実行します。メソッドが実行されると、式が評価され、属性が更新されます。

評価されます。最後に、プロパティ値を取得または設定するなどのすべての他のメソッドコールが実行されます。

- x 4. ページに `m p f` コンポーネントが含まれる場合、ページ要求間でデータベースの状態を維持するために必要なすべての情報は、暗号化されたビューステートとして保存されます。ビューステートは、ページが更新されるたびに更新されます。
- 5. 生成された HTML がブラウザに送信されます。ページに JavaScript などのクライアント側技術が含まれる場合、ブラウザによって実行されます。

ユーザがページを操作すると、ページは、action、getter、および setter メソッドを実行するための必要に応じてコントローラオブジェクトにアクセスします。

ユーザによって新しい get 要求が発行されると、ビューステートとコントローラオブジェクトは削除されます。



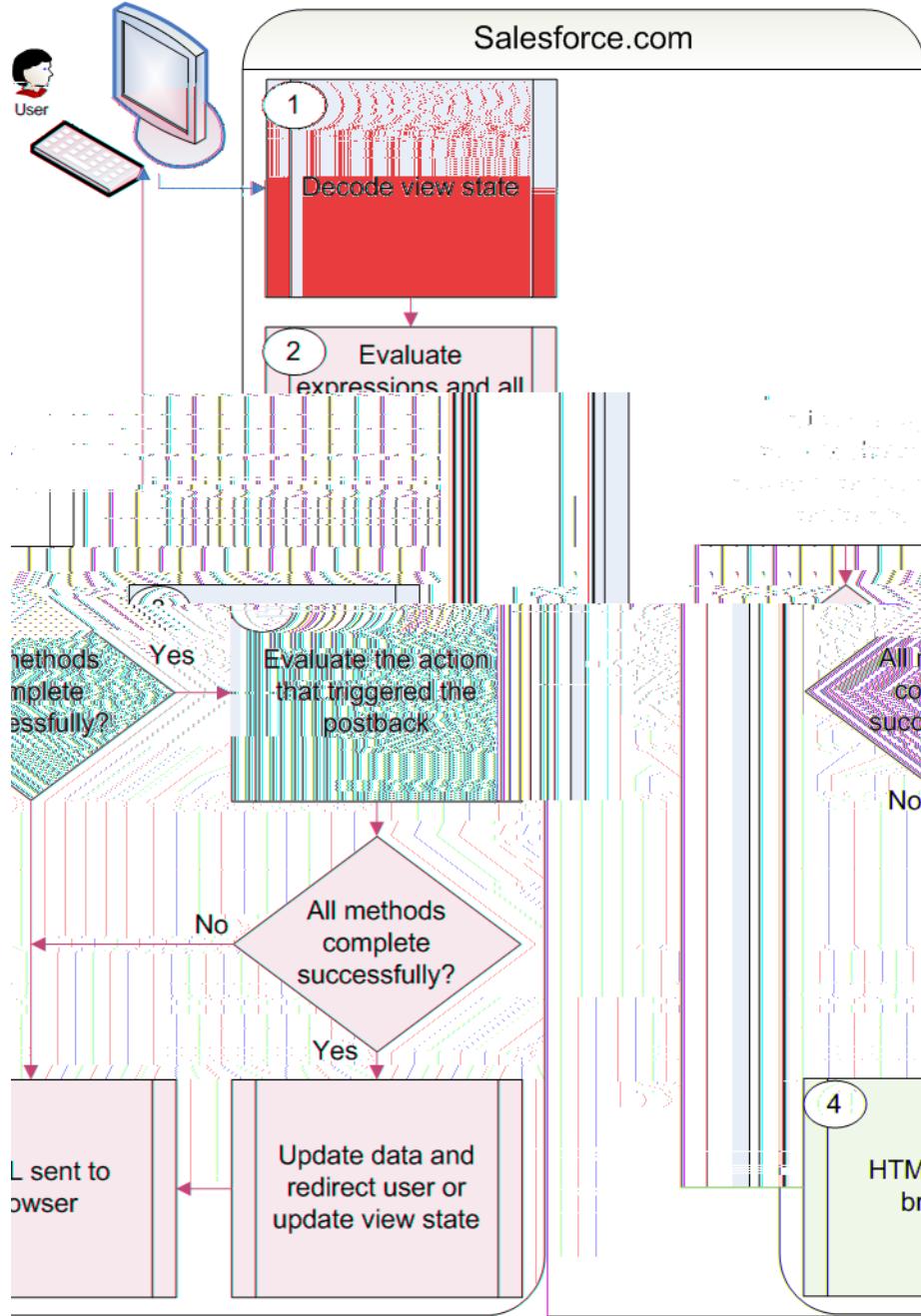
メモ: ユーザは、同じコントローラ、および同じまたは適切なコントローラ拡張のサブセットを使用するページにリダイレクトされ postback 要求が発行されます。postback 要求が発行されると、ビューステートが維持されます。

save アクションをトリガする [保存] ボタンをユーザがクリックするなど、ユーザの操作によってページの更新が必要となる場合、postback 要求が発行されます。postback 要求の詳細については、[Visualforce ページの postback 要求の実行順序 \(ページ 91\)](#)を参照してください。

get 要求の具体的な例については、[Visualforce ページの実行順序の例 \(ページ 93\)](#)を参照してください。

Visualforce ページの postback 要求の実行順序

postback 要求は、[保存] ボタンをクリックし、save アクションをトリガするなど、ユーザの操作によってページの更新が必要になった場合に発行されます。次の図は、postback 要求時の、Visualforce ページとコントローラ拡張またはカスタムコントローラクラスとのやり取りを示します。



- postback 要求時にビューステートは復号化され、ページの値を更新するための基盤として使用されます。



メモ: `action` 属性が `button` に設定されたコンポーネントでは、要求のこのフェーズが省略されます。つまり、アクションは実行されますが、入力に対する検証は実行されず、ページでのデータ変更は行われません。

- ビューステートが復号化された後、式が評価され、カスタムコンポーネント用に定義されたコントローラの `set` メソッドを含む、コントローラとすべてのコントローラ拡張の `set` メソッドが実行されます。

すべてのメソッドが正常に実行されない限り、これらのメソッドのコールによってデータが更新されることはありません。たとえば、メソッドの 1 つがプロパティを更新し、入力規則や不正なデータ型が原因でその更新が無効である場合、データは更新されず、ページが再表示されて該当のエラーメッセージが示されます。

3. postback 要求をトリガしたアクションが実行されます。アクションが正常に完了すると、データが更新されます。postback 要求によってユーザが同じページに戻ると、ビューステートが更新されています。



メモ: `g p p` コンポーネントの `c a` 属性は、postback 要求時には評価されません。get 要求時にのみ評価されます。

4. 生成された HTML がブラウザに送信されます。

postback 要求でページリダイレクトが指定されており、そのリダイレクト先が、リダイレクト元ページと同じコントローラ、およびリダイレクト元ページのコントローラ拡張の正しいサブセットを使用するページである場合、postback 要求がそのページに対して実行されます。リダイレクト先がそのようなページでない場合、そのページに対して get 要求が実行されます。postback 要求に `m p f` コンポーネントが含まれる場合、postback 要求の ID クエリパラメータのみが返されます。



ヒント: `tp f dcr` の `ta` 属性を使用して、postback 要求または get 要求のどちらが実行されるかを制御できます。`c` が `true` に設定されている場合、get 要求が実行されます。これを `false` に設定しても、ターゲットが同じコントローラと正しい拡張のサブセットを使用している場合にのみ postback 要求が実行されるという制限は無視されません。`c` を `false` に設定した場合、ターゲットがこれらの要件を満たさなければ get 要求が実行されます。

ユーザが別のページにリダイレクトされると、ビューステートとコントローラオブジェクトは削除されます。

postback 要求の具体的な例については、[Visualforce ページの実行順序の例](#) (ページ 93)を参照してください。

Visualforce ページの実行順序の例

次の例では、ユーザが Visualforce ページを操作するときのページのライフサイクルを説明します。例に使用されるページは、取引先に関する情報(ページの変数の値)を表示し、キー値が `false` 以外に設定されている場合にユーザが取引先の詳細を編集できるように設計されています。

この例で使用する Visualforce ページを設定する手順は、次のとおりです。

1. というカスタムコンポーネントのコントローラを作成します。

```
public class LineItemController {
    private Account account;
    private List<LineItem> lineItems;
    private Map<String, String> editModeMap;

    public void addLineItem() {
        LineItem lineItem = new LineItem();
        lineItem.setAccount(account);
        lineItem.setUnitPrice(100);
        lineItem.setQuantity(1);
        lineItem.setComments("Initial Comment");
        lineItems.add(lineItem);
    }

    public void updateLineItem(LineItem lineItem) {
        if (editModeMap.get(lineItem.getId()) != null) {
            lineItem.setComments(editModeMap.get(lineItem.getId()));
        }
    }

    public void deleteLineItem(LineItem lineItem) {
        lineItems.remove(lineItem);
    }
}
```

2. というカスタムコンポーネントを作成します。

```
public class LineItem {
    private Account account;
    private Integer quantity;
    private Double unitPrice;
    private String comments;
    private Boolean editMode;

    public void setEditMode(Boolean editMode) {
        editModeMap.put(this.id, editMode ? "Edit" : "View");
    }

    public void increaseQuantity() {
        quantity++;
    }

    public void decreaseQuantity() {
        quantity--;
    }
}
```

13. **or Ba r C** というカスタムコントローラを作成します。

```

w blh n prc tle g m or id ric s i a ss a y
t r n t A h p t ouf cou v cc á a a ;
t e n { or pld r C i y ( )
e ew tou e cd ne td e c m ee a b ms ltnm Ed ap t rano m ouf cosy , s y
} eew e re x d A eh te n egrntpe e ct q e =gt rm Ba . gd Pa (.) ] Pa a s (.)
e t etbilAn t quAn cg ou cc i ( )
} eet n rt rm ou cc a ;
e e e e b le en { prc f c i Ra a ; a ( )
e t n n ru lh ou p cc a ;
} e e e e b le en qrn fl c c ic Ra a ( )
e t n n ru lh
}

```

14. **fc** というコントローラ拡張を作成します。

```

w blh n ep rc e h l g l c li i f s i a ss a y
t r n t A h t p ouf cc v cc á a a ;
r Ag Id dm E p ;
} e t e blt l te ip c t or ch d r C or l d ir c y ( y
t Am te nc ou t or cl d tr An s g ea ( cc ) . ;( )
} t e blt t { p n g rg m g i nro i s G i ( )
t er rm e t nc mn i nro + md a f a ' i a ; '
} e e blt { d r p c l m E pi v i s ( ) yb = ; 0
e t t d ee b r mo l m f p a . a a
}

```

5m E というページを作成します。

```

e en te n gor pld ip c t se l d r aC b e talantx e m e y coe o" l a S y" f "b $ s=" i y"
e x e t o r c m E pia " ! s " >
e ex te te lego pt lp a r < g k ag a i " ! i " >
e ex xt te eo u ub e p lte p < b te jan L ecou es a h d a" ! l oin l y . a : i s N a a "
e ex xt et eo u u l p lte p n s ou a tecc m il v a" mcc a . " ai = " N " a > /
e ex xt te eo u u b p t ej n eo u w h dceer h m om l o y . l o y . l oin l ip s N : o ys . a "
e ex et et eo u u l p lte p < ou ea bcc m il v a" d ! p a m o p ys " i = " s" > /
e ex br B l go p p c < k/a a >

```

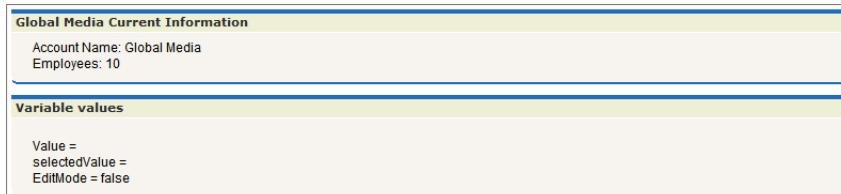
```

e ex te tB l g o e p l p cr bl k l a u a v i ='' i a v a s' >
e ex e { d doc:l t a u M n e r e < e C i g w f m a p ! P l a a a s . " y > /
e ex B l g o p p c < k/a a >
e ex e { n o r o e n p e n e r e t e g a g e $ r m p ! r u P l a a a s . y ' " >
e ex tB l g o t p t l p c d t A n k t a e p o u h a d c a U B ='' l g o n a c " i ='' k P a " >
e ex e tB l g n p p n o c l o & m k a a s l s ='' >
e ex x et n e u l d p l t y n < o m t e n a n a u i c e m N a v a a ! a . " a > /
e ex x et n e { u l d p l t y n < o m t e n a n a u i c e m N a v a a ! p a N O y s " > /
e ex e tB e l g a p p o c & m k a a s >
e ex x t te e o u u b e p l t e u p < b t e j a n L c o u t a h c a d ! u f l b l y . a i s . s
e ex t t r a d f u r c c I ='' a s " y > /
e ex x t e n n o p o g < a i a R i >
e ex x et n e { u l d p l t y n t s o u a c t f t m a d ! u r a c I . s " y i ='' a
e ex x t r e t e & u p r o e n e e n e p o p e n t a r e i d a r s v B l g o n a c ='' k
e ex x t t t t u u s s a' s d a > /
e ex x t e n n o p o g < / a i a R i >
e ex e tB e l g a p p o c < c m k/a a s >
e ex tB l g a p p o c < c m k/a a s >
e ex tB l B g o t p u p m o < o k k o o a m s i a ='' k a' v S a > /
e ex tB l B g o t p u p m o < o k k o o a m s i a ='' k a' v S a > /
e ex x t t t t B l g o p p c < k/a a s >
e ex x ro m p f < / a a >
e ex x g p p < / a a >

```

get 要求の例 1

最初の例では、`h e : p e S a l e s f o r c e _ i n s t a n c e / d p E p r e c o r d I d` のフォームの URL を使用してページに移動します。URL の `Salesforce_instance` はインスタンスの名前（など）で、`recordID` は組織の取引先レコードの ID（など）です。次のページに似たコンテンツのページが表示されます。



ライフサイクルを追跡して、なぜページがこのように表示されるのかを確認しましょう。URL を直接入力してページを要求したため、このページは、postback 要求ではなく、get 要求により生成されています。

1. get 要求では、はじめにカスタムコントローラとコントローラ拡張のコンストラクタメソッドがコールされますが、`c` メソッドは、コントローラのコンストラクタで、`f c` メソッドは、拡張のコンストラクタです。これらが実行され、2つのオブジェクトが生成されます。コントローラには `cc` という変数があります。これは、クエリする取引先オブジェクトを識別するために URL の パラメータを使用するクエリによって生成されたものです。拡張には `cc` という変数があります。これはコントローラの `c` メソッドをコールして作成されたものです。`cc` メソッドには副次的影響はありません。

2. get 要求における次のステップではカスタムコンポーネントが作成され、関連付けられたコントローラまたはコントローラ拡張のコンストラクタメソッドが実行されます。ページには、1つのカスタムコンポーネントが含まれます。

```
e { d doc:l tMn ex e Cig vñm ap ! Pla a a s . "y > /
```

このカスタムコンポーネントには、コントローラが関連付けられていますが、このコントローラには明示的なコンストラクタはありません。明示的なコンストラクタがないすべての Apex オブジェクトと同様に、このオブジェクトは引数をとらない暗黙的な公開コンストラクタを使用して作成されます。カスタムコンポーネントの作成の一部として、カスタムコンポーネントの value 属性が設定されます。この場合、式

の結果と同じになります。URL に属性を指定しなかったため、属性は null に設定されます。

3. カスタムコンポーネントが作成されたら、カスタムコンポーネントのすべての属性が実行されます。属性は、その値を関連付けられたカスタムコンポーネントコントローラのクラス変数に割り当てるsetter メソッドです。カスタムコンポーネントには、メソッドがあるため、これが実行されます。setter メソッドは、属性の value を属性に設定します。属性は null に設定されるため、value も null に設定されます。
4. get 要求における次のステップでは、コンポーネントの c_a 属性、式、および必要な getter メソッドと setter メソッドが評価されます。以下でこれらの手順を順に行いますが、これらの評価の順序は確定であり、次に示す順序とは異なる場合があります。

- コンポーネントには、拡張の m とメソッドをコールする c_a 属性があります。そのメソッドは、オブジェクトの l_m_a_f_p 項目を 10 に設定します。
- ページで評価する式は複数あります。次の 3 つに絞って説明します。

```
te g p lego pt lp ar< g k ag a i = ! i " >
```

tbl go p pl c の属性は、ライフサイクル拡張 g のgetter メソッドをコールします。これは、「Global Media の最新情報」としてページに表示されます。

no red n erkt e ag et\$rm y Jr u Pla a a s . y ! " >
mepl f d の < 属性は、> パラメータの値に基づいて設定されます。ページをコールするときに を設定しなかったため、フォームは表示されません。

この式は、カスタムコンポーネントに現れます。l_d と value は null に設定されることはない前に説明しましたが、E_M の値はまだ不明です。n は、cor l_a_p_c の boolean 变数です。 が null であるかどうかに基づいて設定されます。

```
eed led l uc l u s/ a/ h a/ - ' is !  
e ed do t m t file d o i // s =v ( a = ; )
```

l_u は null であるため、Elam は f に設定されます。ただし、E_M の setter メソッドには副次的影響があります。M を設定する作業の一部として、value を s に設定しました。 は null であるため、これによる変更はありませんが、この動作によって後の例に影響が出ます。

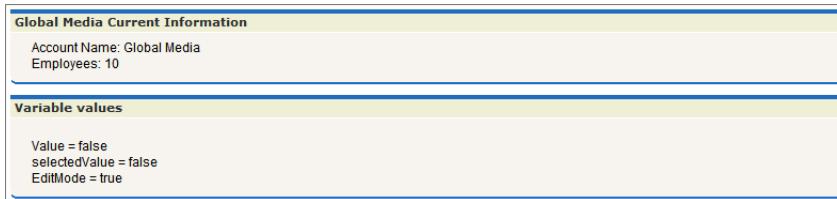
- 他の式とメソッドが同様に評価されます。

mpf コンポーネントは表示されないため、ビューステートは作成されません。

6. get 要求の最後のステップは、HTML をブラウザに送信して、その HTML を表示させることです。

get 要求の例 2

2つ目の例では、フォームで `peSalesforce_instance` / `recordID` の URL を使用して、ページに移動します。`Salesforce_instance` はインスタンスの名前（など）で、`recordID` は組織の取引先レコードの ID（など）です。最初の例とは異なり、この例には `fk` という 2 つ目のパラメータが含まれます。次のページに似たコンテンツのページが表示されます。



もう一度ライフサイクルを追跡しましょう。このページも、get 要求の結果生成されるページです。

1. get 要求では、はじめにカスタムコントローラとコントローラ拡張のコンストラクタメソッドがコールされま
す。コンストラクタメソッドは、コントローラのコンストラクタで、
トメソッドは、拡張のコンストラクタです。これらが実行されて、2つのオブジェクトが生成されています。コントローラには `cc`
という変数があり、これは、クエリする取引先レコードを識別するために URL から得る パラメータを使用するクエリによって生成されたものです。拡張には `cc` という変数があります。これはコントローラの
メソッドをコールして作成されたものです。
2. get 要求における次のステップではカスタムコンポーネントが作成され、関連付けられたコントローラまたは
コントローラ拡張のコンストラクタメソッドが実行されます。ページには、1つのカスタムコンポーネントが
含まれます。

このカスタムコンポーネントにはコンストラクタのない関連コントローラがあるため、コントローラオブジェクトは、引数をとらない暗黙的な公開コンストラクタを使用して作成されます。カスタムコンポーネントの作成の一部として、カスタムコンポーネントの `greeting` 属性が設定されます。この場合、式の結果と同じになります。`greeting` 属性を `false` に指定したため、`greeting` は `false` に設定されます。

3. カスタムコンポーネントが作成されたら、カスタムコンポーネントのすべての属性の `set` メソッドは、属性の `value` を属性に設定します。属性は `false` に設定されるため、`value` は `false` に設定されます。
4. get 要求における次のステップでは、コンポーネントの `greeting` 属性、式、および必要な getter メソッドと setter メソッドが評価されます。以下でこれらの手順を順に行いますが、これらの評価の順序は確定であり、次に示す順序とは異なる場合があります。
 - コンポーネントには、拡張の `get` メソッドをコールする `greeting` 属性があります。そのメソッドは、`cc` オブジェクトの `label` 項目を 10 に設定します。
 - ページの式のうち、ここで選択した 3 つの式が評価される方法を確認しましょう。

```
<apex:pageBlock title="{!greeting}">
```

コンポーネントの `greeting` 属性は、ライフサイクル拡張の `get` の getter メソッドをコールします。これは、「Global Media の最新情報」としてページに表示されます。

`<apex:form rendered="{$CurrentPage.parameters.key = 'true'}">`

この式は、カスタムコンポーネントに現れます。この属性は、ページをコールするときに `rendered` 属性の値に基づいて設定されます。ページをコールするときに `key` を `null` に設定したため、フォームは表示されません。

`Value = {!value}
 selectedValue = {!selectedValue}
 EditMode = {!EditMode}`

この式は、カスタムコンポーネントに現れます。この属性は、null ではないため、`Value` は `null` に設定されます。この時点で、`selectedValue` は `null` に設定されます。ただし、`EditMode` のsetter メソッドには副次的影響があります。この場合、副次的影響によって、`selectedValue` は、カスタムコンポーネントの `EditMode` 属性に設定されます。`EditMode` は `false` に設定されています。このことから、メソッドで副次的影響を使用すべきでない理由がわかります。評価順序が異なり、`EditMode` のsetter が評価される前に `selectedValue` の値が決定される場合でも、`selectedValue` は `null` になります。実行順序は保証されないため、`selectedValue` の結果は、このページに次回アクセスしたときに変化する可能性があります。



警告: getter または setter に副次的影響を使用しないでください。

このコンポーネントは表示されないため、ビューステートは作成されません。

6. get 要求の最後のステップは、HTML をブラウザに送信して、その HTML を表示させることです。

get 要求の例 3

3つ目の例では、`h : Salesforce_instance / apex:page recordId=" / si` フォームの URL を使用して `Account` ページに移動します。`Salesforce_instance` はインスタンスの名前(など)で、`recordID` は組織の取引先レコードの ID (など)です。2つ目の例とは異なり、この例では `recordID` が設定されます。次のページに似たコンテンツのページが表示されます。

もう一度 get 要求のライフサイクルを追跡しましょう。

1. get 要求では、はじめにカスタムコントローラとコントローラ拡張のコンストラクタメソッドがコールされま

す。このメソッドは、コントローラのコンストラクタで、`fc` メソッドは、拡張のコンストラクタです。これらが実行されて、2つのオブジェクトが生成されています。コントローラには `cc` という変数があり、これは、クエリする取引先レコードを識別するために URL から得る パラメータを使用するクエリによって生成されたものです。拡張には `cc` という変数があります。これはコントローラのメソッドをコールして作成されたものです。

2. get 要求における次のステップではカスタムコンポーネントが作成され、関連付けられたコントローラまたはコントローラ拡張のコンストラクタメソッドが実行されます。ページには、1つのカスタムコンポーネントが含まれます。

```
e { d doc:l tMn ere t e Cig vñm ap ! Pla a a s . "y > /
```

このカスタムコンポーネントにはコンストラクタのない関連コントローラがあるため、コントローラオブジェクトは、引数をとらない暗黙的な公開コンストラクタを使用して作成されます。カスタムコンポーネントの作成の一部として、カスタムコンポーネントの `ap` 属性が設定されます。この場合、式 `y` の結果と同じになります。`y` 属性を `true` に指定したため、`ap` は `true` に設定されます。

3. カスタムコンポーネントが作成されたら、カスタムコンポーネントのすべての属性のsetter メソッドは、属性の `ap` を `ap` 属性に設定します。`ap` 属性は `true` に設定されるため、`ap` は `true` に設定されます。
4. get 要求における次のステップでは、`g p p` コンポーネントの `c a` 属性、式、および必要なgetter メソッドとsetter メソッドが評価されます。以下でこれらの手順を順に行いますが、これらの評価の順序は不確定であり、次に示す順序とは異なる場合があります。
- `g p p` コンポーネントには、拡張の `m` 上の `ap` メソッドをコールする `c` 属性があります。そのメソッドは、`acc` オブジェクトの `l m a f p` 項目を 10 に設定します。
 - ページの式のうち、ここで選択した 3 つの式が評価される方法を確認しましょう。

`<apex:pageBlock title="{!!greeting}">`
`tbl go p pl c の ap 属性は、ライフサイクル拡張 g の getter メソッドをコールします。これは、「Global Media の最新情報」としてページに表示されます。`

`<apex:form rendered="{!!$CurrentPage.parameters.key = 'true'}">`
`me pl f d の ap 属性は、> パラメータの値に基づいて設定されます。ページをコールするときに ap を ap に設定したため、フォームが表示されます。`
`Value = {!value}
 selectedValue = {!selectedValue}
EditMode = {!EditMode}`
`この式は、カスタムコンポーネントに現れます。te が null ではないため、ap は ap に設定されます。以前の例と同様に、ap は null に設定されます。ap の setter メソッドの副次的影響によって、ap は ap に設定されます。`

コンポーネントが表示されるため、ビューステートが作成されます。

6. get 要求の最後のステップは、HTML をブラウザに送信して、その HTML を表示させることです。

postback 要求の例

最初の 2 つの例とは異なり、3 つ目の例では、編集可能項目のクリック可能なボタンと共に最終ページが表示されました。postback 要求がどのように機能するかを理解するために、例 3 の最終ページを使用して、取引先名を「Pan Galactic Media」、従業員数を「42」、および業種を「その他」に変更します。その後 [保存] をクリックします。これにより、postback 要求が開始されます。

- postback 要求では、はじめにビューステートが復号化されます。ビューステートには、ページを表示するためには必要な情報がすべて含まれます。postback 要求中に操作が失敗した場合、ページをユーザに表示するためにビューステートが使用されます。
 - 次に、すべての式が評価され、コントローラとコントローラ拡張のメソッドが実行されます。
- ページの式のうち、ここで選択した 3 つの式が評価される方法を確認しましょう。

```
<apex:pageBlock title=" {!greeting}">
```

tbl go p pl c の ke 属性は、[ライフサイクル拡張](#)ます。この編集では、取引先名の値を変更しました。したがって、Media の最新情報に変更されます。

の getter メソッドをコールし
の値は、「Pan Galactic

```
<apex:form rendered=" {!$CurrentPage.parameters.key = 'true'}">
```

me pl f d の < 属性は、パラメータの値に基づいて設定されます。パラメータを変更していないため、ビューステートの値が使用されます。ビューステートが作成されたときの値は true であったため、この値は true のままであり、フォームが表示されます。

```
Value = {!value}<br/> selectedValue = {!selectedValue}<br/> EditMode = {!EditMode}
```

これらの値を変更していないため、各式について、ビューステートの値が使用されます。

- 最後に、postback 要求をトリガした save アクションが評価されます。save アクションは、コントローラの次のメソッドです。

```
public void updateAccount() {
    Account account = new Account();
    account.name = 'Pan Galactic Media';
    account.industry = 'Other';
    account.employees = 42;
    update(account);
}
```

このメソッドによって、レコードは新しいデータで更新されます。このメソッドが失敗する場合(ユーザにレコードを更新する権限がない場合、または変更によってトリガされる入力規則がある場合)、エラーメッセージとそのエラーの説明と共にページが表示されます。ユーザが入力した値は失われません。これらの値は、ユーザが [保存] ボタンをクリックしたときの状態に維持されます。エラーが発生しなかった場合、オブジェクトのデータは更新され、ビューステートが更新されます。また、postback 要求をトリガしたアクションにページのリダイレクトが含まれていなかったため、ビューステートが更新されます。生成された HTML がブラウザに送信されます。

関連リンク

[開発モードフッターの使用](#)

カスタムコントローラおよびコントローラ拡張のテスト

すべての Apex スクリプトなど、コントローラ拡張やカスタムコントローラに対し、単体テストを実施する必要があります。単体テストは、コード内の特定の部分が正しく機能していることを確認するクラスメソッドです。

単体テストのメソッドは引数を取らず、データベースへのデータの送信を行うこともなく、メソッド定義に
do M h キーワードのフラグが立てられます。

コントローラ拡張およびカスタムコントローラクラスの単体テストを記述するときに、テストで使用できるクエリパラメータを設定できます。たとえば、次のカスタムコントローラとマークアップはコントローラメソッド（ページ 82）の例に基づいていますが、ページの URL にクエリパラメータ p が指定されていることを要求するように拡張されています。次のテストメソッドクラスは、このページの機能を実行します。

```
e t b l i t t e l e f p c c o r l a h r c i s s a
e t t r n t e r p r g m f i a i s i s N ; a
e t t r n t e r p l g m i a i s s N a ; a
e t t r n r n p g m c i a i s ; a
e t t e r n r r p g l m i a i s i ; a
e t t r n r p g q p i a i s ;
e t t k e l u n { o p d d h r c i ( )
t e e x A e q h t e n e g r n t p e e d t i g e . g t r m B a . g q P a ( . ) p P a a s ( . ) ( ; )
}
e t t e b l u t t e { p c g g m i i s i s N a ( )
e t t n r t e u r h m f i s i s N ; a
}
t e b l u t t e d t p c r n t e f i g v i s m f i s i s N a ( i s i s N a ) a
t t e b l u t t e d t p c r n t e h m f r i g v i s m f i s i s N a ( i s i s N a ) a
e t t e b l u t t e r { p c g g m i i s L s N a a ( )
e t t n r t e u l h m i s s N a a ( i s i s N a ) a
}
t e b l u t t e d t p c n t e { m l g v i s m s s N a a ( i s i s N a ) a
t t e b l u t r { m c g g o m c i p i s i s . a ( i s i s N a ) a
}
t e b l u t d t n p c o n m r C n g i v m i c s p a y ( i s ) a y ( )
t n o h m n c o i p c i p s . a y ( i s ) a y ( )
}
e t e b l u t t e r p c g g l m E i i s i s . i a ( i s i a ) a
e t t e b l u t d e p c n l m r E g l m v i s s i a ( i s i a ) a
}
e e e e e b l e n e n { g r c f c i R a = a ( )
e e e e e n g r n f l i c p i R a = a ( )
t e e e e n f g e l h p t i { h i g s | . | == q h ! p y y . ' s a i s . ) )
e t e e t g e t r n p f u r m r o p o r p a m s ( . ) ( ' , ' , ' P a a ' )
e e { } l
t e e e w n e d e w h d t e t d L t e m a l h a L m L a s N a = a i s . s N a a ,
t e e e w n e d e w h d t e t d L t e m a l h a L m L a s N a = a i s . s N a a ,
t e e e w n e d e w h d t e t d L t e m a l h a L m L a s N a = a i s . s N a a ,
t e e e w n e d e w h d t e t d L t e m a l h a L m L a s N a = a i s . s N a a ,
t e e e x t e n c d o E c p a ( i s ; a )
e e g l i p u f = a ( i s ; a )
e e g l i p u f = a ( i s ; a )
```

```

    e t e e t      get      rip      ue rm tn      p. o r Pa a s (.) ( ' , ' 's ; ')
}
}

e { n      lif p      cc      i      (==      )      ss;
e e      e t et dr      p u c      s R i      ( ; )
e t      n r ru      p      ;
}

```

コントローラは、成功ページと失敗ページの2つのページをコールします。この例では、これらのページのテキストは重要ではありません。ただし、テキストが存在することは必要です。

次のマークアップでは、上記のコントローラを使用しています。

```

ten et gor pde pnc t <et ldehrac bal l d="      " sa y=" " a
x B l go p p c < k a a >
e ro m p f < /a >
e te eh ngo ddT pl fgl > s h a ia < s a >
e te te hp d ngo dd plif gis a a p ia < s a > /
te ne r x et pntxe { : u < p d! up erg a x edmn tx v u a! p! s p! < a > p / a < >
te ne x et pntxe { : u < p l > up a k ax etan tx v u a! p ! s p! < a > p / a < > >
en ox etn tpe { p: u p l upne } k ax etn tpx p v u a! p < a > p / a < > >
e el mddix Ept n txe { : u > pi up } ass ent n Ex w p a! p i < a > p / a < > >
x tm oBndm pmo & ea el wua i ad=" ts " a & v S a < a > /
x ro m p f < /a >
x B l go p p c < k/a a >
g p p < /a a

```

次のクラスは、コントローラをテストします。

```

T is s
etbl tle ne p tcor l a hr c i ssTa ss
t te ble t p e t t do tdeM m { i sorild a M s v i s y ( )
e e e ene gr e ef gc p guf Ra cc Ra = Pa = ss;
et te te ne en T C g g sps f Pa ( Ra ; )
ten toer l d hre c oer l d r de n or l a hr c = ss a( . ) ;( )
e ntex r tge n eg oer l a tr c is rg Pa ss a( . ) ;( )
t e t ter w fl yle ev tp ofu / a h yrm ap ai sai a a s
e mr x ie q E l r a y rep ss a tex r s a n / a i / ga ? = Pa a, ' Pa )
ex A e ddt er rm op g / p a a s U L a R
te negr ntp e c g gt rm Ba . u q Pa p. ) p Pa a s ( . ) ( ' , ' y yyy ' )
ten e tn e wn t e n wor l d re ce / s lli at he ran p i g a p a a si a
ten e oer l d r de n or l a hr c = ss a( . ) ;( )
ten e tor l d r c te n lm m La s Na a( s ' a a; ' )
ten e tor l d r c r te n mr mf s s N a( s ' a a; ' )
ten e tor l d r en o m C mp c s a( s ' a ; ' )
ten e tor l d r c t tle m r El f mo sm d a( s ' a a. ; ' )
tex n t e n eg oer l a tr c rg Pa ss a( . ) ;( )
t te tr e e f hu N dcg / a p a ps ss a( s ' a a. ; ' )
e mr x ie q e E tex s p s s c a g s a( ' a s / ss , ' Pa )
e ede l tde ] l ed L ed e mr el ds a fr e [ oih m , ic a pm c a] a ; ' a ; '
t e t mr t he r E le s p s e mss am d c s a( s m s a a. , ' s a [ .oi a ) a
}
```



ヒント: コントローラをテストするときに、次のエラーメッセージが表示される場合があります。

このメッセージが表示される場合、 というクラスを作成しているかを確認してください。作成している場合は、そのクラスの名前を変更します。

入力規則とカスタムコントローラ

カスタムコントローラを使用する Visualforce ページにユーザがデータを入力し、そのデータが入力規則エラーになった場合、エラーが Visualforce ページに表示されることがあります。標準コントローラを使用するページと同様に、入力規則エラーの場所が `u ldp p` コンポーネントに関連付けられた項目の場合、エラーはそこに表示されます。入力規則エラーの場所がページ上部に設定されている場合は、`g p p` 内の `< m p` コンポーネントを使用してエラーを表示します。ただし、ページに情報を取得するには、カスタムコントローラが例外をキャッチする必要があります。

たとえば、次のページがあるとします。

メモ: このページを表示するには、有効な取引先レコードの ID が URL のクエリパラメータとして指定されている必要があります。たとえば、

次のようなカスタムコントローラを記述する必要があります。

c or ld M C i ss a y
ou cc a ;

grc f c i Ra w a ()
ou p cc y a a ;

cm d h o Ec p D a (i Ba ass)
e gld x p g M . a s s sa (;)

;

etblAt t n t ene w e n rt t ou e cd ne d n c tm br and s so[fcrAnq mauf cc i as Pa(.)] Pa a's (.) i(' ; ')
t ou fcdli i (a ==)
re x d A eh te negrntpe e t q e =gt rm Ba . gd Pa(.)
ou cc a ;

ユーザがページを保存したときに、入力エラーがトリガされると、標準コントローラの場合と同様に、例外がキャッチされ、ページに表示されます。

transient キーワードの使用

キーワードは、保存ができず、Visualforce ページのビューステートの一部として送信することもできないインスタンス変数の宣言に使用します。次に例を示します。

また、逐次化可能な Apex クラス(つまり、コントローラ、コントローラ拡張、エンティティまたはエンティティアソシエーション)でインターフェースを実装するクラス)でキーワードを使用できます。また、逐次化可能なクラスで宣言する項目の型を定義するクラスでキーワードを使用できます。`String`

変数を **レコード** として宣言すると、ビューステートのサイズが縮小されます。キーワードは、**レコード** です。

Visualforce ページでページ要求の間のみ必要な項目でよく使用されます。この項目は、ページのビューステートには含まれず、要求中に何度も再計算するには非常に大きなシステムリソースを使用します。

Apex オブジェクトの中には、自動的に transient と判断されるものもあります。つまり、その値はページのビュー・ステートの一部として保存されません。例として次のようなオブジェクトがあります。

- PageReferences
 - XmlStream クラス
 - コレクションが自動的に transient とマーキングされるのは、Savepoints のコレクションなど、コレクションに含まれているオブジェクトが自動的に transient とマーキングされている場合だけです。

・r クラスインスタンス。sPa

また、**静的な**変数はページのピューステートを使用して転送されません。

次の例には、Visualforce ページとカスタムコントローラの両方が含まれています。ページが更新されることに transient 日付は再作成されるため、[refresh] ボタンをクリックすると、日付が更新されます。非 transient 日付には、ピューステートが適用されません。

```

t e n      g e r p l o   p t c e   n   k   m o E a d   r   p   = "       a           "   >
}           br:   T           1     1   ! <           > / 
}           b2:   T 2          ! <           > / 
x           n   e n o   e d m e n   p c   l   u r   k   la   fl h   ia v   a"       s "   > / 
           ro   m   p   f   <           a           > 
           ro   m   p   f   <           /a           > 
           a   n   n   <           /a           > 

```

blex 1 te plic d moE lð r p i ss a a
t m T D ja1 :

```
tre t m T $ & ia ;  
t e blit { r p cg g T i S 1 ( )  
t n t ru e tif wn iol ( == 1 ) = s $ . ;( )  
} + 1 ' ;  
t e blit { r p cg g 2 T i S ( )  
t n t ru e tif 2 wn 2 iol ( == ) = s $ . ;( )  
} + 2 ' ;
```

第8章

高度な例

クイックスタートチュートリアルの例は、入門レベルの例として、Visualforce マークアップのみを主に使用しています。高度な例では、Visualforce マークアップのほか、Force.com Apex コードが使用されています。

初めてのカスタムコントローラの作成

これまで、このチュートリアルのすべての例では、標準取引先コントローラを使用して各ページの基盤のロジックを定義してきました。Visualforce では、カスタムコントローラを定義して独自のロジックやナビゲーションコントロールをページに追加できます。次のトピックでは、カスタムコントローラクラスを作成したり、Visualforce マークアップとやり取りできるクラスメソッドを定義したりするための基本を説明しています。

- カスタムコントローラクラスの作成
- getter メソッドの定義
- action メソッドの定義
- navigation メソッドの定義
- カスタムリストコントローラによるレコードの一括更新



メモ: Salesforce ユーザインターフェースを使用した Apex の追加、編集、または削除は、Developer Edition を使用している組織、Salesforce Enterprise Edition トライアル版を使用している組織、または Sandbox を使用している組織でのみ行えます。Salesforce の本番組織では、Force.com 移行ツール または Force.com API のいずれかを使用してのみ Apex に変更を加えることができます。

カスタムコントローラクラスの作成

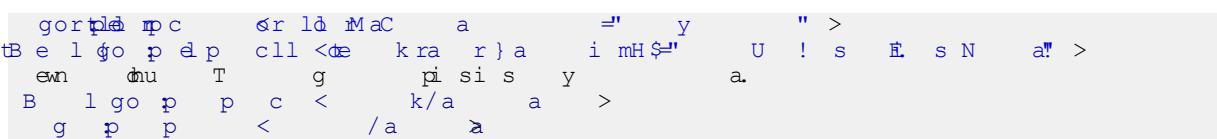
カスタムコントローラは、要するに Apex クラスです。たとえば、次のコードは非効率的ですが、有効なコントローラクラスです。

```
bill_tell_n_p_c_or_lde_M_C_i_ss_a_y
```

コントローラクラスを作成して次の 2 つの方法でページに追加できます。

- コントローラ属性をページに追加し、「クイック修正」を使用して、コントローラクラスをその場で作成します。

1. ページエディタで、コントローラ属性を `g p p` タグに追加します。次に例を示します。



2. クイック修正オプションを使用して、MyController という新しい Apex クラスを自動的に作成します。

- 任意の Apex エディタでコントローラクラスを作成して保存し、ページで参照します。
 - アプリケーションで、[設定] の [開発] > [Apex クラス] をクリックし、[新規] をクリックして新しいクラスを作成します。
 - ページに戻り、上記の例で説明されているとおりに `g p p` タグに `controller="MyController"` 属性を追加します。

 メモ: ページは、一度に 1 つのコントローラのみを参照できます。`g p p` タグで `controller="MyController"` 属性と `controller="Acme"` 属性の両方を使用することはできません。

有効なカスタムコントローラを参照するページを保存するとただちに、ページエディタの横に 2 つ目の [コントローラ] エディタタブが表示されます。このエディタでは、ページマークアップとページのロジックを定義する Apex とを切り替えることができます。

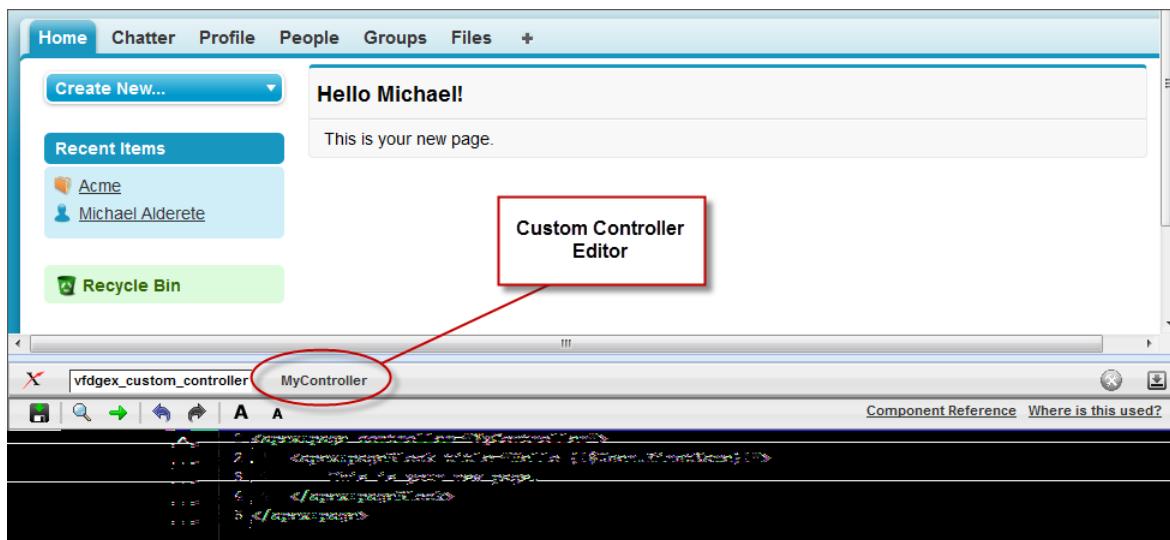


図 17: カスタムコントローラエディタ

getter メソッドの定義

Visualforce コントローラクラスに関する主要なタスクの 1 つに、データベースや他の計算値をページマークアップに表示する方法を開発者に提供するというタスクがあります。この種の機能を有効にするメソッドは *getter メソッド* と呼ばれ、一般的に *Identifier* と名づけられています。*Identifier* は、メソッドが返すレコードやプリミティブ値の名前です。

たとえば、次のコントローラには、コントローラの名前を文字列として返すための getter メソッドがあります。

```
blu t el n pc or l& M C i ss a y
t e blu { r p c g g m i is N a ( ) ;
n rterm or l& M C y ; }
```

getter メソッドの結果をページに表示するには、プレフィックスを使用しない getter メソッドの名前を式で使用します。たとえば、getter メソッドの結果をページマーカアップに表示するには、`<![CDATA[]>` を使用します。

```
x ten gorpl& mpc sr l& MaC a = y " >
x te Blgo p d p cll <de kra r> a imH$= U ! s E s N a! >
ex e ew due t T e gn t epinsfi sm gyrl& r c a ! a
B l go p p c < k/a a >
g p p < /a >
```

標準取引先コントローラを使用した前述の例では、ページには、`c$ <fieldName>` 式を使用して(クエリ文字列パラメータを使用した) URL に指定された、取引先レコードの値が表示されました。これは、標準取引先コントローラに、指定された取引先レコードを返す `cc` という getter メソッドが含まれるため実行できました。次のコードを使用して、カスタムコントローラでこの機能を模倣することができます。

```
blu t el n pc or l& M C i ss a y
t e blu { r p c g g m i is N a ( ) ;
n rterm or l& M C y ; }

etblAit cuAn cg ou cc i ( )
e en rt rue l n d d m o u f s c i , a
e ew e re x d A eh te n e g r n t p e e t g e = g t rm Ba . gd Pa( . ) ] Pa a s ( . ) i ( ' ; )
```



メモ:

この例が正しく機能するためには、Visualforce ページを URL の有効な取引先レコードに関連付ける必要があります。たとえば、`Id` が取引先 ID の場合、次の URL を使用します。

```
h e : p Salesforce_instance // pM dg t / a E/y s Eal 00 000B003 5
```

`get` メソッドは、埋め込み SOQL クエリを使用して、ページの URL の パラメータで指定した取引先を返します。たにアクセスするために、`get` メソッドは次のように名前空間を使用します。

- `get` メソッドが現在のページの `fc` インスタンスを返します。
は、クエリ文字列パラメータなど、Visualforce ページへの参照を返します。
- `get` ページ参照に基づいて、`rm` メソッドを使用して、指定されたクエリ文字列パラメータの名前と値の対応付けを返します。
- 次に、`get` を指定する メソッドのコールにより、パラメータ自体の値を返します。

MyController コントローラを使用するページは、`cc__n` または `cca__.` 式を使用して、それ `d` または 項目のいずれかを表示できます。コントローラの SOQL クエリによって返される項目はこれらの項目のみであるため、ページではこれらの項目のみを使用することができます。

標準取引先コントローラをさらに緻密に模倣するには `ctg__p_bp` タグに `< a` 属性を追加して他の取引先ページと同じスタイルをページに適用できます。ページのマークアップは、次のようにになります。

```
ten gortplä ppc t set lð Mac b talAn e'u y cc " a Sy" " >
x teB e lðgo p d p cll <ð k r a i mH $! U ! s E s N a! >
e ewn dneut T e gr a t ðpnsf ð m y lð r c bra ! a < . > /
ex w@torn tegn n ðñ tY ac m a o i cc ! a . a a .
B l go p p c < k/a a >
g p p < /a a
```

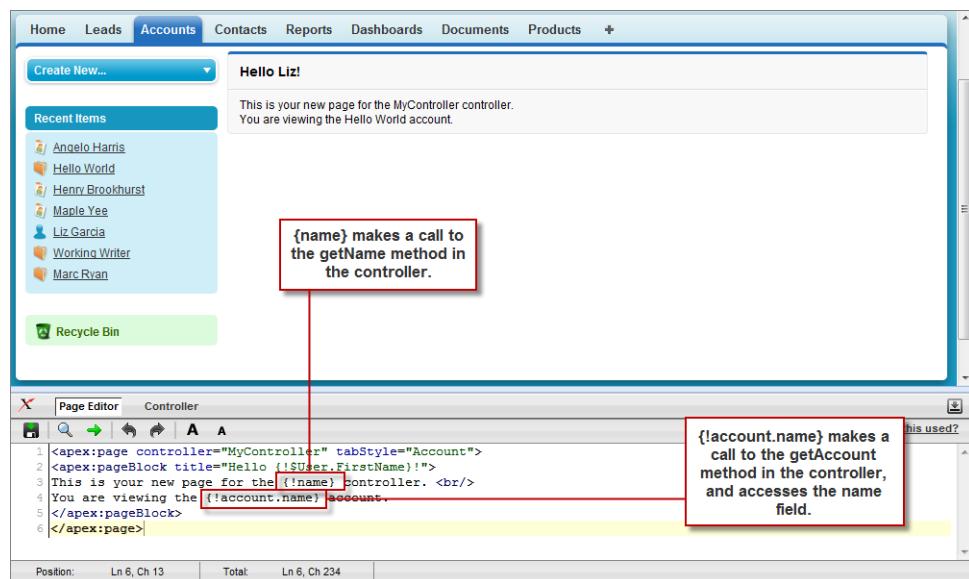


図 18: カスタムコントローラを使用してページで値を表示する

action メソッドの定義

action メソッドは、ユーザがボタンをクリックしたり、ページ内のある領域にマウスポインタを移動したりするなどのページイベントが発生すると、ロジックまたはナビゲーションを実行します。次のいずれかのタグの `{ o }` パラメータに `表記` を使用することによって、ページマークアップから action メソッドをコールできます。

- `oBndmm` `o` `< o` はアクションをコールするボタンを作成する
- `no dmm` `o` `< o` はアクションをコールするリンクを作成する
- `o lð prc` `< o` は定期的にアクションをコールする
- `o upro` `< o` は、別の名前つきのコンポーネントにイベント（「`onclick`」、「`onmouseover`」など）を作成し、アクションをコールする
- `a upc o c <` `o` は、アクションをコールする新しい JavaScript 関数を定義する
- `g p p` `o` はページが読み込まれると、アクションをコールする

たとえば、「ページでの入力コンポーネントの使用」(ページ 27)で説明されているサンプルページでは、コマンドボタンは、標準取引先コントローラのメソッドにバインドされています。MyController カスタムコントローラを使用するように、前の例を適応させることができます。

```

x     te n      gortplæ mp c t & t lâ MaC b talAn      e'u      y cc      "      a Sy="      " >
x     ro m p f      <      a      >
ex    teB e l gø p ðp cll <æ kra rja i mH$=      U ! s  ã s N      a! >
e     e      w@torn      tegn n      ðñ tY ac m à oii      cc      !      a      p.      a      a < .      > /
e     e      teAn g kðu      cc m      :      ða      N < a      > /
e     x      et n      e { u ldp ltep n      oþ      acc mñ v      a!      ! p a      .      'a< > /      > /
e     x      tñ      oñndmm wæo      e ð      e ea l wu a teAn ia =butñ      ðñ m      a! v      S ñ
ex    ro m p f      <      k/a      a      >
x     g p p      <      /a      a      >
N      " a >

```



メモ: このページに取引先データを表示するには、有効な取引先レコードの ID をページの URL のクエリパラメータとして指定する必要があります。次に例を示します。

```
h   e : p Salesforce_instance m xp x      bJ / a /y Ra1=? 00 008 s
```

レコードの ID の取得についての詳細は、[Visualforce での項目値の表示](#)(ページ 20)を参照してください。

上記のページを保存すると、Visualforce エディタで「クイック修正」オプションを使用して、MyController クラスにメソッドを追加できます。^a クイック修正リンクをクリックすると、MyController は次のようになります。

```

bl t el n      pc or lâ M C      i      ss a y
e e e e ble      en { grc      f c      i      Ra      ;      a ( )
e n n      ru lñ
}
t e blñte { r      pcg g      m      i      is      N      a ( )      ;
e n rterm      or lâ M C
}
etblñt      eupAn cg ou      cc i      ( )
e e n rt      rue l n d      ð rano      mouf scf i      ,      a
e e w      e rex d A eh      te n egrntpe e ð g e =gt      rm Ba.      gd      Pa(.)      ] Pa a s (. ) i ( ' ;
}

```

クイック修正によって生成されるメソッドは、action メソッドに使用する標準署名を使用します。これは public であり、PageReference を返し、引数を含みません。

最終的に、メソッドの定義は、新しい取引先値を使用してデータベースを更新する必要がありますが、まずその前にメンバー変数を定義して、データベースから取得される取引先情報を保存する必要があります。取引先のメンバー変数がない場合、データベースから取得されるレコードは、その値がページを表示するために使用された後で保持されないため、ユーザによるレコード更新は保存できません。このメンバー変数を導入するには、コントローラコードの 2 つの部分を変更する必要があります。

- メンバー変数をクラスに追加する
- が最初のクエリを実行するときにメンバー変数を設定する

```

bl t el n      pc or lâ M C      i      ss a y
e An t om      cou      cc      a ;
}

```

```

e e e e b l e   en { grc          f c      i      Ra      vs      a ( )
c n m ru     lu ; 

t e b l i t e { r p c g g m      i      i s      N      a ( ) ;
n r t e r m   or ld M C           ; 

etbiAit      &uAn cg ou cc i      ( a == ) 
t n nou fcdh i      a a ; 
t e n e tou e cd ne td c m trano = msouf i cc , a$ 
e ew e rex d A eh te negrntpe e & g e =gt rm Ba . gd Pa ( . ) ] Pa a s ( . ) i 
n r t rm ou cc a ; 
}

```

メンバー変数が適切に配置されたため、メソッドで行う必要があることはデータベースの更新だけです。

```

bl t el n p c or ld M C i ss a y
c An t on cou cc a ; 

e e e e b l e   en { grc          f c      i      Ra      vs      a ( )
t td n nou p cc a a ; 
c n m ru     lu ; 

t e b l i t e { r p c g g m      i      i s      N      a ( ) ;
n r t e r m   or ld M C           ; 

etbiAit      &uAn cg ou cc i      ( a == ) 
t n nou fcdh i      a a ; 
t e n e tou e cd ne td c m trano = msouf i cc , a$ 
e ew e rex d A eh te negrntpe e & g e =gt rm Ba . gd Pa ( . ) ] Pa a s ( . ) i 
n r t rm ou cc a ; 
}

```

に対するより堅牢なソリューションとしては、さまざまな例外の検出や重複の検索などを行うことが考えられます。ここでは単純な例を示すことを目的としているため、そのような詳細は省略しています。

このページをテストするために、[取引先名を変更] 項目の値を変更して、[新規取引先名を保存] をクリックします。標準取引先コントローラの例と同様に、ページは単に新しい取引先名で更新されます。次の例では、save アクションを拡張して、現在のページを更新する代わりにユーザを別の確認ページに移動します。



メモ:

ページが正しく表示されるように、URL に有効な取引先 ID を指定する必要があります。たとえば、
g が取引先 ID である場合、次の URL を使用します。

```
h e : p Salesforce_instance // pg dg / a g/y s Bal 00H 000R00 6
```

navigation メソッドの定義

カスタムコントローラのaction メソッドは、データベース更新および他の計算を実行するほか、PageReference オブジェクトを返して、ユーザを別のページに移動することができます。

PageReference は、ページのインスタンス化への参照です。多数の属性の 1 つである PageReferences は URL、一連のクエリパラメータ名および値で構成されます。

カスタムコントローラまたはコントローラ拡張では、次のいずれかの方法で、PageReference を参照またはインスタンス化できます。

```
g      existingPageName Pa.
```

組織すでに保存している Visualforce ページの PageReference を参照します。このプラットフォームはこのようにページを参照することで、コントローラまたはコントローラ拡張が指定されたページの有無に依存することを認識し、コントローラまたは拡張が存在する間はページが削除されないようにします。

Force.com プラットフォームでホストされる任意のページに PageReference を作成します。たとえ

えば 'partialURL' を recordId に設定すると、`newPageName` が SalesforceInstance の

にある Visualforce ページを参照します。同様に、`partialURL` を recordId に設定すると、指定したレコードの詳細ページを参照します。

この構文は、PageReference はコンパイル時ではなく 実行時に構成されるため、`existingPageName` の
ページ以外の Visualforce ページの参照にはお推めしません。実行時の参照は、参照整合性システムには使用できません。したがって、プラットフォームはこのコントローラまたはコントローラ拡張機能が指定されたページの有無に依存することを認識しないため、ユーザによるページの削除を防ぐためにエラーメッセージを表示しません。

```
g      existingPageName Pa.
```

外部 URL の PageReference を作成します。次に例を示します。

```
g      newPageName = www.salesforceinstance.com/p/g /a /y S Pa
```

この例では、ユーザが [保存] をクリックした後に、そのユーザを新しい URL にリダイレクトすることを想定します。これを行うには、まず次の URL に移動し、クイック修正を使用して `mySecondPage` という 2 つ目のページを作成します。

```
h      e : pSalesforceinstance.com/d/p/g /a /y S Pa
```

それから、`mySecondPage` に次のマークアップを追加します。簡略化のために、チュートリアルの最初の方で定義した次の標準コントローラベースのページを使用します。

```
t      n t e gndprd por hñ n C au scc a a " " >
e      ll de r r } mH$ U ! s E s N a!
e      w@torn p tegrn n < dñ tX ac m á oíi cc ! a p. a a< . >/
```

ここで、[action メソッドの定義](#) (ページ 109)で作成した元のページに戻り、取引先 クエリパラメータを URL に指定していることを確認します。新しく作成した「`mySecondPage`」ページに PageReference を返すように、コントローラの メソッドを編集します。

```
bli t el n p c or lñ M C i ss a y
c An t om cou cc a ;
e e e e ble en { grc f c i Ra ; a ( )
```

```

t      td n   ou p   cc   a   a ;
e e e e n   ger n   e   fie c   ge   n   gRam d   s g   Pa   Pa   y S   Pa
e   et ed t&gt;g dr r u s   Pa R i   ( ; )
e   n re mu d   g   s   Pa
}

t   e   blte {r   pcg g   m   i   is   N   a( )   '
n rterm   or l&gt; M C   y   ;   '

etblAt   &upAn cg ou cc i   ( )
t   n   nou fcdh i   ( a == )
t   ene   tou e cd ne td c   m   t rano = msouf i cc , a$ 
e   ew   e   re x d A eh te   negrntpe e &gt; g e =gt   rm Ba .   gd   Pa(.) ] Pa a s (.) i
n rt rm   ou cc   a ;
}

```

上記のコードでは、`PageReference` の `isExternal` 属性は `true` に設定されています。この属性が設定されていない場合、`PageReference` はブラウザに返されますが、移動は発生せず、元のページの URL のままになります。移動先の URL を変更する場合は、`isExternal` 属性を設定する必要があります。

ここでページをテストする場合、[新規取引先名を保存] をクリックすると `mySecondPage` に移動しますが、データのコンテキストは失われるため、`isExternal` で使用できる値はありません。これは、リダイレクトが発生したときにコントローラがコンテキストの状態をクリアするからです。そのため、`PageReference` のパラメータの対応付けて クエリ文字列パラメータをリセットする必要があります。

```

blt e tl   t p c d   Mr l&gt; pr d   &us a   y   a
c An t om   cou cc   a ;
e e e e blt   en { grc   f c   i   Ra   &   a( )
t   td n   ou p   cc   a   a ;
e e e e n   ger n   e   fie c   ge   n   gRam d   s g   Pa   y S   Pa
e   et ed t&gt;g dr r u s   Pa R i   ( ; )
e   et ed t   g gt   rm ts dn   Pa d cpa a s (.) i( ' , ' a i . ; )
e   n re mu d   g   s   Pa
}

t   e   blte {r   pcg g   m   i   is   N   a( )   '
n rterm   or l&gt; M C   y   ;   '

etblAt   &upAn cg ou cc i   ( )
t   n   nou fcdh i   ( a == )
t   ene   tou e cd ne td c   m   t rano = msouf i cc , a$ 
e   ew   e   re x d A eh te   negrntpe e &gt; g e =gt   rm Ba .   gd   Pa(.) ] Pa a s (.) i
n rt rm   ou cc   a ;
}

```

ウィザードの作成

ここまで Visualforce マークアップとコントローラの重要な機能について説明してきましたが、この最後の例では、こうした機能と一緒に使用して 3 ステップから成るカスタムウィザードを作成する方法を説明します。このウィザードでは、次のようにユーザが商談と同時に、関連する取引先責任者、取引先、および取引先責任者のロールを作成できます。

- ・ ステップ 1: 取引先と取引先責任者に関連する情報を収集する
- ・ ステップ 2: 商談に関連する情報を収集する

- ・ステップ3: 作成されるレコードを表示し、ユーザが保存またはキャンセルできるようにする

このウィザードを実装するには、ウィザードの3つのステップのそれぞれに対応する3ページと、各ページ間のナビゲーションの設定とユーザが入力したデータの追跡を行う1つのカスタムコントローラを定義します。



重要: 複数の Visualforce ページにまたがって使用されるデータは、最初のページでデータを使用しない場合でも、最初のページ内で定義する必要があります。たとえば、項目が 3 ステッププロセスの 2 ページ目と 3 ページ目で必要な場合、1 ページ目にもその項目が含まれている必要があります。項目の属性を **fv** に設定することで、この項目をユーザに非表示にすることもできます。

これらの各コンポーネントのコードは、下記のセクションに含まれていますが、3つのページはそれぞれコントローラを参照し、コントローラは3つのページをそれぞれ参照するため、まずその最適な作成手順を理解する必要があります。やっかいなことは、ページがないとコントローラを作成できませんが、コントローラでページを参照するにはページが存在している必要があるということです。

この問題を解決するには、最初に完全に空のページを定義し、次にコントローラを作成してから、マークアップをページに追加します。したがって、ウィザードページとコントローラを作成する最適な手順は次のようになります。

tt 1. 1 ページ目の URL は :p_salesforce_instance // p pp に移動し、[Create Page oppyStep1 (ページ oppyStep1 を作成)] をクリックします。

2. ウィザードの他のページである opp 2と pp についても、上記のステップを繰り返します。

4. ここで、作成した 4 つのページのエディタに戻り、それらのコードをコピーします。これでウィザードは期待どおりに機能します。



メモ: 空のページを作成することはできますが、その逆のことはできません。ページがコントローラを参照するためには、そのコントローラのすべてのメソッドとプロパティが設定されている必要があります。

商談ウィザードコントローラ

次の Apex クラスは、新規顧客商談ウィザードの 3 つのページすべてのコントローラです。

t ethlnt to tn p c Cgo c c á ()
t tn on t finalic wa t tnic (co a= C c) a()
t rtn ru o c c ; a a; ()

t blite n tro ptc t{ mpj ro u i pØ i y Ø i y()
t t no ro m f tpplh eo o ro inu t tppin y e u)pp i y o i y()
n t tu no ro u pp i ; y

t bht n t rdn pue tø pp lo C ágo Ø i y Ra R ()
rndo e lfh w lot t n t rdn(u == o pp lo)C = c o i y Ra ; ()
e n r ru r lo ;

w z e trd h erere eruh n /gr et arde foaceo fs ra Rñ h g p
n w ze t t rethe oe t ttrial tehi r dea d mo oa di o ia s
t etbne e eoe e negret b e/fac etcn d mao Rae o dn o s d Lg s
n e ure h ñ mer o m g // gss svp a a
e eeble e nt { grc f c i p Ra s ()
e n r terut g pp p Pa ly s;

e eeble e nt { grc f c i 2 p Ra s ()
e n r terut g pp 2 p Pa y s;

e eeble e nt { grc f c i p Ra s 3 ()
e n r terut g pp p Pa y ss;

t e at dh w Thz ec a rd sh/ dr rtet asur ho a, dh s s
t et n tro u pp Ø/ i i s a
e eeble ea grn fl c c ic Ra a ()
e n gr teno ed oe wn e pgA Ra n te nd pd gt Rñ tr ec o row u Ba . spa a
ten@ ret ve tpg dr r u ci y Pa Ri (;)
n t tu no ro u pp g i y Pa }
t e et dh r Tðen m pfl A/s efo lt ouf on jif d vac a a
ment et e ht gte tr eho d/e tle h a wago sp f h i a a
t t no ro u pp // i y
e eeble en { grc f c i Ra s a ()
e tet r te ne C eoBh tn co n et/ f at gñ a ca p ic s á , y
en n toe tn b h mpt oen enoume // oc b h m pld fa
ten nout tnœ dn p co c dh p. = a ;
tn t rn ou cc i s a ;
e tet rt tn e C oe B h tnc re aetr/ fe a gñ d alñ i s fi , s i i
t ee et e etn h d ot re e dauh/e se f d a or ca i si s a
e et end et what nb t tn e pnb/ h dasic ouh cc a a a
tn to n co t d n cc oñ d cc a a = a i.;
tn t tn r o c c i s ; a
e tet r t ten eo ro la B tn ppe n /tr af rg i cy i s i , a
et @ t or m lwe ho t hn /phauh cñ asi i a .
t t no rot un pp t d n cc oñ d cci y a = a i.;
tn t t no ro u pp i s i ; y
e tet r t n nC tñp le e c wætlo b c a t nio ro hu a pp i y
etn t dn o h c c // a
t r to no ro ut dt pp o ro I u d pp i y = i iy;
t tm lo d tnd c oc I dc c . a = ia ;
tre r r lo i s ;

e e t h̪l tð t e tr əho d̪ // l ha y̪s go s p f i a a
e whn t t n o hro u pp // i y

e e n t ðgr e œ w̪i œ x gA tpp R̪a t ð nd prd qt R̪a tr eC o r o w u Ba . ſpa a (i vyi)
e e et œ t ð g œpr r u c y P̪a R i (;)

R̪er ru o g pp y P̪a

商談ウィザードのステップ1

次のコードは、ウィザードの10ページ目(pp)を定義します。このページでは、関連付けられた取引先責任者と取引先に関するデータをユーザから収集します。

te n e g o m p l d t p c t n & e o n u a t c o m p l d r C b" lt On rō u y pp " a Sy" O i " y
r c p < s i >
n une o{ nofr ml f C c i i a ()
e nr nd & ea rm v fu fu se oua n à "h(c y s y i s a " ;)
e en t & ln fr Cruc r u i i \$ a) ;
e n r ru l f sa
r c p < s i / >
te e n te t & ph row & l < tu a & r h m c n t u a i t h p N l s o i p" y s i = 1 s 3" >
x ro m p f < a >
ex te tB l g e p l p c m t < or ekChao and filo m d' s i a" = i " >
e e ttB lbgo e uT p e t o < d t t > k t n t h e fi o h t e t s t rhai s p o h s pa a a a a
e mt ed be o o o m e B lef o t p a c e n ke & w arLc m r pp a hai, a a a a a a a a
e e tx tt ne b mu a T on e d c ! --tB elBgo u p o i c ai iks a s a a
e et et ll te a e hor l a & ep tlo sa e k s e e h e h tn gr pio f c s a Ra
ete ext text et w o rphh s i a >-
ex ttB lbgo p u p o c < k a a s >
x tt oBn dmm & n d < } ec & x l u a p i a = ls "v a" N " > /
x tt oBn dmm & n e no & } c da & ma l i a & c! a v a" a" a"
ex ttB lbgo p u p o c < k/a a s >
ex e tB lega p p o & l a n k o a n n a m & fi = i a" >
t en e ttB hl ga & n w o c ! a w i l d k l a p a i t l i s i, & hs h asya s ja i
e n r n t t e l c o g i w l b l p s i a. >-
x et n u l d p d n p { ou < c c m t e u n f n i o y " ce m N " v a a" ! a . " a > /
x et n u l d p d n p { ou < c c & t e u n f i o y " ce i s a" ! a \$ " > /
ex e tB lega p p o c < c k/a a s >
ex e tB lega p p o & l a n k o a n n a m & fi = a i a" >
x et n ut l d p d t e p o & c a c t m l f n u i t e = b r e a s N " v a a" ! a s N " a > /
x et n ut l d p d t e p o e < c a c t m l f n u t e = l k c a s N a n v a a" ! a s N a " a > /
x et n ut l d p d p a { < o t d t h u e f n = } co a & v p a" ! a " > /
ex e tB lega p p o c < c k/a a s >
ro m p f < / a >
g p p < / a >

ウィザードの1ページ目のマークアップについては、次の点に留意してください。

- `l go p p ec` タグは、`k`オプションで `l go p u w c < k` 子要素を取り込み、コンポーネントのヘッダーとフッターに表示されるボタンを制御できます。`l go p p c` の本文に表示される
 - `l bao p u w c < k` タグの順序は重要ではありません。ウィザードのこのページでは、

ttB lbgo pu p c < k タグに、ページブロック領域のフッターに表示される [次へ] ボタンが含まれます。

- ・ ウィザードは、[キャンセル] ボタンがクリックされると JavaScript コードを利用してダイアログボックスを表示し、終了するかどうかをユーザに確認します。この例では、簡略化のためにマークアップに直接 JavaScript を含めていますが、実際には JavaScript コードを静的リソースに配置してそのリソースを代わりに参照することをお勧めします。
- ・ ウィザードのこのページでは、[次へ] ボタンがコントローラのメソッドをコールし、そのメソッドが次のステップに返します。

```
ttB lbgo pu p c < k a a s >
ttB oBnfmM pmt oe<}ec ax l u2p ia="! ls "v a" N " > /
ttB lbgo pu p c < k/a a s >
```

コマンドボタンはフォームに表示する必要があります。これは、フォームコンポーネント自体が、新しい基づいてページ表示を更新するためです。

- ・ `lga p poc & k` タグは、データのセットを表示用に整理します。テーブルと同様に、`lga p poc & k` は 1 つ以上の列で構成され、各列は 2 つのセル（1 つは項目の表示ラベル、1 つは値）に展開されます。`lga p poc & k` タグの本文に含まれる各コンポーネントは、列数に達するまで、行内の次のセルに配置されます。列数に達したら、その次のコンポーネントは次の行の最初のセルに配置されます。

`u ldp p` などの一部のコンポーネントは、自動的にページブロックセクション列の両方のセルに一度に展開され、項目の表示ラベルと値の両方に入力されます。たとえば、このページの [取引先責任者情報] 領域では、[名] 項目が最初の列、[姓] 項目が 2 番目の列に入り、[電話] 項目が次の行の最初の列に折り返します。

```
te n tB lego p pto dnl & ndk a rao c m& fi =" a i a" >
x et n ut ldp d te po & fc ac tmlenu ite="b rasa s "a a" ! a s N "a > /
x et n ut ldp dte po e<{ c ac tmlenu tel e' 1 )cas Na riva a" ! a s Na "a > /
x et n ut ldp d po { < o tad tmlenu dn="} co e Pw p a" ! a " > /
e tB lga p poc <c k/a a s >
```

- ・ 前のコードの抜粋に含まれる最初の `u ldp p l` タグの `e a t` 属性は、コントローラの `firstName` メソッドから返された取引先責任者レコードの `firstName` 項目にユーザの入力を割り当てます。

ページは次のようにになります。

図 19: 新規顧客商談ウィザードのステップ1

商談ウィザードのステップ2

次のコードは、ウィザードの2ページ目(Step 2)を定義します。このページでは、商談に関するデータをユーザから収集します。

```

t e n e  g o m p l a t i p c t n t e o n u a t o e p l a r C b t" l t O n r a u y pp " a S y" O i " y
r c p < s i >
n u n e o{ nofr al f C c i i a ( )
e n r n r n l G erA r m v fuw t u s e o u a n à "h( c y s y i s a" ; )
e en t et ln fr Cruc r u i i $ a ) ;
e n r ru l f s a

t e r c p < s i / >
te e n t e t ø p r o w etl < tu a sr HmC met u a i t h m N l s o i 2 p" y s i = " s 3" >
x ro m p f < a >
ex te tB l go p l p nc r e n t u k a m p o a m d i f l o - m d i i y i a" = " i " >
ttB lBgo p u p o c < k a a s >
x tt oBndmm w e n d oe < } e a l u a r p o i a = " ls " v a" v i p s" > /
x tt oBndmm w e n d oe < } ec a x l u a p i a = " ls 3 " v a" N " > /
x tt oBndmm w e n d no < } c da à ca l l a = " cl a " v a" a"
ex ttB lBgo p u p o c < k/a a s >
ex tB l e n p o t c l t e n k r o a t u a m p o s m d i f = " o i y i a" >
x et n u l d p d e n p r o e < { app m t E u t à o = " r o u ) i p p y m v a a" ! i y " a > /
x et n u l d p d n p o t o < { app m t u t i n = b r t u n i ) p p y m v a" ! i y a " > /
x et n u l d p d e p e o t < u e l { app E C i i u t = " n e o r e t u p l s p p " v a c a" ! i y D s
x et n u l d p d e p e o < u e { app g E m t i u t = " n e o r e t u y S p a " g a m a" ! i y $ N a
x et n u t l e p d p o { < l e c a c u E r l o j i r # b R a " v a" ! . " > /
ex e tB l g a p p o c < k/a a s >
B l go p p c < k/a a >
ro m p f < /a >
g p p < /a >

```

フォームに [完了予定日]、[フェーズ]、および [取引先責任者の役割] 項目を配置するマークアップは、他の項目と同じですが、`u l d p p` タグが各項目のデータ型を調べて表示方法を決定します。たとえば、[完了予定日] テキストボックスをクリックするとカレンダーが表示され、そこからユーザが日付を選択できます。

ページは次のようにになります。

図 20: 新規顧客商談ウィザードのステップ2

商談ウィザードのステップ3

最後のコードブロックは、ウィザードの30ページ目(pp)を定義します。このページでは、すべての入力データが表示されます。ユーザは、操作を保存するか、前のステップに戻るかを決定できます。

タグでページに書き込むだけです。

最後のページは次のようにになります。

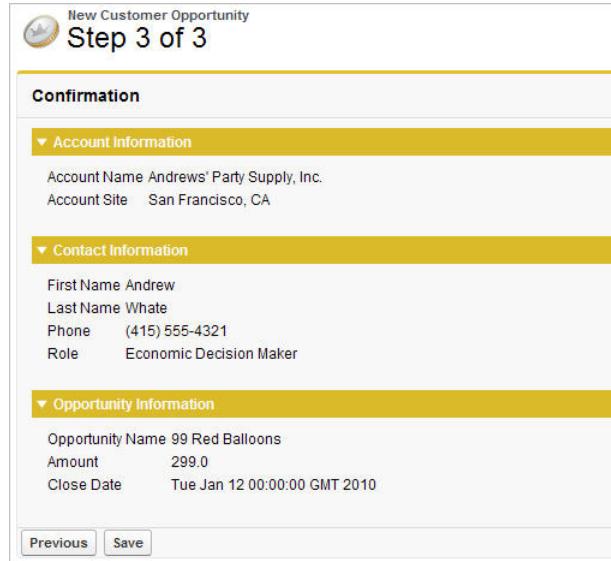


図 21: 新規顧客商談ウィザードのステップ 3

高度な Visualforce ダッシュボードコンポーネント

Visualforce ページは、ダッシュボードコンポーネントとして使用できます。ダッシュボードでは、ソースレポートから得たデータを、グラフ、ゲージ、テーブル、総計値、または Visualforce ページなど、視覚化されたコンポーネントとして表示します。コンポーネントは、組織の主要な総計値のスナップショットおよびパフォーマンスの指標を提供します。各ダッシュボードには、最大 20 個のコンポーネントを含めることができます。

標準コントローラを使用する Visualforce ページをダッシュボードで使用することはできません。Visualforce ページをダッシュボードで使用するには、そのページがコントローラを含んでいないか、1 つのカスタムコントローラを使用しているか、または StandardSetController クラスにバインドされたページを参照している必要があります。Visualforce ページは、これらの要件を満たさない場合、ダッシュボードコンポーネントの「ページ」ドロップダウンリストにオプションとして表示されません。

次の例は、ダッシュボード内で使用でき、カスタムリストコントローラを使用する Visualforce ページを示します。「Barbara Levy」という名前の取引先責任者に関連付けられたすべてのオープンケースを表示します。

```

t e n e <oeplde pcr tr <t ea b aD ='' ivC s " a a Sy=" s " a>
x t t e l go p p c < k a a >
t ex e } c m c C ! Na as 's sa
e x e B el{go p l c u < T k a rca va c a" ! s sa"v ='' a" >
e x e f lo ut h u } < u c a v a" ! s sa" > /
e ex e B l go p l p c < T k/a a a>
ex B l go p p c < k/a a >
g p p < / a >

```

このコードは、ページに関連付けられたカスタムリストコントローラを表示します。

```

e ke le { r p r c C i ss a iv s a
t e blit tr t e n re rB u p c g go C m ci i s L a s a N a a ( )
} te ble te { p c g C i s a s sa ( )

```

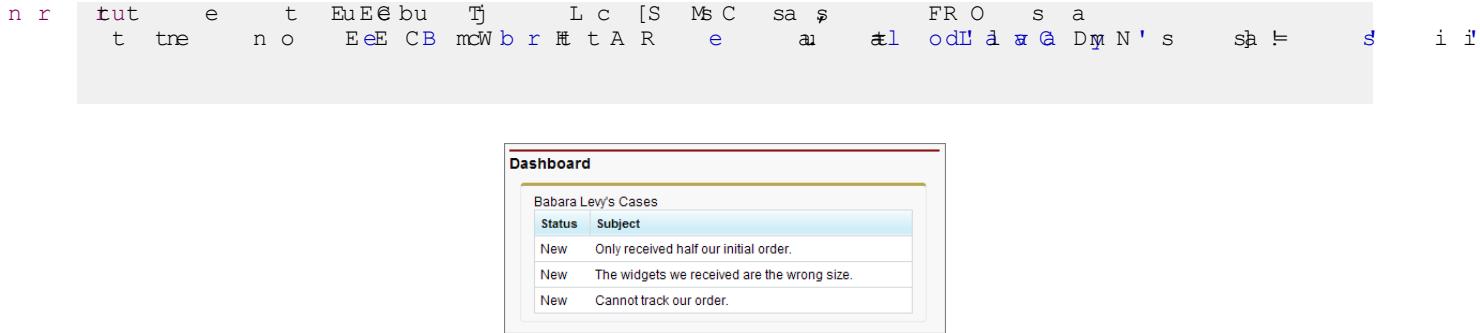


図 22: ダッシュボードで実行する Visualforce ページのサンプル

関連リンク

[Visualforce ダッシュボードコンポーネントの作成](#)

Visualforce と Google Chart の統合

Google Chart は、さまざまな視覚効果でデータを動的に表示する方法を提供します。Visualforce と組み合わせることによって、Google Chart はダッシュボードを使用するよりも高い柔軟性と配信可能性を提供できます。グラフは URL を介して生成されるため、画像を使用できるところであれば、視覚化を共有して組み込むことができます。

Google Charts API の使用には 2 つの前提条件があります。1 つ目は、データの符号化方法の決定です。Google Charts API には、テキスト、簡易、および拡張という 3 つのデータ符号化タイプがあります。この例では、簡易符号化のみを使用します。2 つ目は、使用的するグラフの種類の決定です。この例では、ユーザは棒グラフと線グラフのいずれかを選択します。

カスタムコントローラには、上記の要件に対応する `c` と `c` という重要な関数が 2 つあります。

- ・ 関数 `c` は数値を取り、Google Chart の簡易データエンコードタイプに変換します。詳細は、Google Charts API ドキュメントの [「Simple Encoding Data Format」](#) を参照してください。
- ・ 関数 `c` は、Google Chart API に要求を行う URL を構成します。

次のコードは、Visualforce ページのコントローラを表します。

```

t t he ten h o cTif cn tdo hi g cross tae trin sah f i h ha s i
t od gAr dI I G * a P / 
k u lt p d c gn do Eyc ss a G a ai
x t t n e ne E coen dlp n htrr o / budi s a ay s a a
bli nle nk p edoe tnk Eyc en ax d pss a d Ec ip s i
bli nlt p c mero de xdt pp i Eya U ss a os E p i y i
e tx en xt{ d n } o Ec p s i
n n wdo eh gE me etn en hp d/ e mi g e tkd ra ru sak ia h y a s
ete r n en eio pe lgcun de s d o i gv a s ai y G .
t en mh z Tm pd i s i s i i i a i i ( ) / 
t e et n t ep n Mg d p ne g i d t gca bM , pi s > i a ; s ;
e e n n ne m m md o t lgcue tu / e di i re n ppm i g h doas gc i
e et l ph ntDnru g . i s a s s i * i i ( ) / 
e et en n { p neg I d t gca bM a i a ; s ;
e n n ne m m md o t lgcue tu / e di ro n ppm i g h doas gc i

```

```

e et e e t l wh ntTdnru gy . i s ass i* i i ( ) /
e et en n { me g rido t gc v M a i i ; s ;
e et n do h gEr erd pg/ o d A q y alI IME E s ' PO y S P
e et n l m d p i si * . / nblu ne{ m pndo EgC i p E TIME E EyE EX L S P, DN
e et l u tem h mleTne o u tn / mrg hnon o a doi f gc i a a i
e et nbu e nt r pge Item t gr }i p i ; v s ;
e et l u tem h mleTne o u tn / mrg hnon o a doi f gc i a a i
e et nbu ex t r pge Item t gr }i p a ; v s ;
n et n do h gEr nt rd e epqA/ n d h d y I a g Pi y G
nblu en e{ do ep c EgC e te t g i p y y ; v s ;
t blit tne d C to eip le tr d dsb sp qn s a pvi y a
t blit tne d C o ep e d ar ejt lpf d s q c y qn p
t blu er te prg g g h p i i s a ; s ;
t blin t re pte t dn g i i s U L R ; s ;
e et e ewtd et I tcr eh hr le A b d s l d ias ya p a ; s ; i s ya
e et blB n do{ tde lte t plg i = 0 a ai ( )
e n blt m o{ g n do EgC i E a ai ( )
x m m n e n do p EgC i = 1 a ; 6 y L s ; p
t n td el r r plc f i s ya a s a ; ( )
e e e ble e ne t g c r f c c i Ra a ( )
n et rt t dg et t] d i s [ L a ss = a ss i ( , , ; 0)
en er g d l u pp d s V : a a = s ; '
t ttr d a e h : phU d g Rg sp m ' c A/ aidsa . . / a s= 360 00 '
t te m te d p tx x m cal =idp D s i s a & a = , y , y '
x mr d p d a | =1 1010 , 0 , , 0 , 6 5 '
xe ml e et d p 2: c r | : d M + | s s 3 s ; '
e r g of r tmh o dp r ip i T C( a a ( ' a a ==) 0)
t r t dn m b d p U L R = & a v s ; '
} 1 e r g of r nmh o dp ps i T C a a i ' a ==) 0)
t r t dn m l d p U L R = & a = s ; '
} 1 s
t e w wn r noe rk t do n Eg a note c u p t r o d i r pp idh ( ' sp ' s i s a y '
ew e et l d e + c eg n dt o hs ' sp + a ( a ' s i s . ; ')
t nro rt d dg to t d : ( i s a a P La ss )
e e t e ne et d l u pp E g l Itut n v + a a = ( m v . a O ( a a P . i ( ; ))
t td etl r en e m d p U uL R pp+ & a v ' a a
} n m ru lu r plo i s ya a ; ;

```

```

        t e t e tener rTm et n ade ht s ge r t trm spl pu i h y a av a a
        et e eme dn edn dcp f gcas s ipi * y . / 
        t t bt e nt p re ng gtdo i Erse in at c ds p Eg c d s M ( i y )
        t ne n def t Eg c t n r iruE (==T T i y . x ) ; ' '
        t l ne n dofe Eg c nsri pM(=E i y L S P ) s ; ' '
        t l ne n dof e Eg c snirE E T i y M DN ) ; ' '
        e t l n r ru s ; ' '
}

t et e dd e ll n w g e the m n se d h s a c i i a
ne te r ng n elo fl n g a td c a lo t e tc rd cgs a a h si ai
n et n dt e egc e l p l u f n d d l d t p y o u d f v o a v i f h a $ .
e d l n r d r n a e g m t d o m Eg c w o n a c r ap , a h i i i * s . /
t e blut e r ne ep cng do E c d g i I i s ( )
x d { mnd m ill (> a < i )
t e w wn r noe nk b do en Eg c eo E d u p o d d i dp V i ( ' a v i / a a ( '
w t e ete oai d d l n e d ncm sma $ g ai / a a ( '
n xm m+ + ti + '/ a ; ') )
}

l s
t e n rn ran t do gc dg M p i a ( ; )
}

t e t en dnter Th n n w l o s h e t g c m i m i d s p d h i a i i s a
e exrde ero t t defr t c p c s m p t z e t m m i im s s i i i s a
n o o c iv* i a . /
r t n d p i v v ai i i ( )
n e n f d o p Eg c i ( pM(=E i y L S P )
n n do gc M i 1=a ; 6
n n ndo gc M i i = ; 0
n n do w n e egc n M p m Mg i pi g = a < a , is > ;( )
n n t do gc u M p p i a ( , 0 ; ' )
n n t do gc uB M p p i 1a ( , ; ' )
n n t do gc u M p p2 C i a ( , ; ' )
n n t do gc u M p p E i a D3 ( , ; ' )
n n t do gc u M p p i a ( 4 ; ' )
n n t do gc u M p p i a ( F5 ; ' )
n n t do gc u M p p i a ( , G ; ' )
n n t do gc u M p p i a H 7 ( , ; ' )
n n t do gc u M p p I i a ( 8 ; ' )
n n t do gc u M J p p i 9 a ( , ; ' )
n n t do gc u M p p i 1a ( , , 0 K ; ' )
n n t do gc u M p p 2 M i 1a L ( , ; ' )
n n t do gc u M p p i 1a 3 N ( , ; ' )
n n t do gc u M p p i 1a ( 4 O ; ' )
n n t do gc u M p p i 1a ( , 5 ; P )
n n t do gc u M p p i 1a Q ( , 6 ; ' )
n n t do gc u M p p i 1a 7 R ( , ; ' )
n n t do gc u M p p i 1a ( 8 ; S )
n n t do gc u M p p i 2a ( , ; ' )
n n t do gc u M p p 2 i a U ( , 0 ; ' )
n n t do gc u M p p 2 i 1a ( , ; ' )
n n t do gc u M p p 22 i Wa ( , ; ' )
n n t do gc u M p p 2 i a 3 X ( , ; ' )
n n t do gc u M p p 2 i a ( 4 ; ' )
n n t do gc u M p p 2 i a Z ( , 5 ; ' )
n n t do gc u M p p 2 i a ( , 6 ; a )
n n t do gc u M b p p 2 i a 7 ( , ; ' )
n n t do gc u M p p 2 c i a ( 8 ; ' )
n n t do gc u M d p p 2 i 9 a ( , ; ' )
n n t do gc u M p p i a 3 ( , 0 ; ' )
n n t do gc u M p p f i 1a 3 ( , ; ' )
n n t do gc u M p p 2 i a 3 ( , ; ' )
n n t do gc u M p p h i a 33 ( , ; ' )
n n t do gc u M p p i a 3 ( i4 ; ' )
}

```

```

n   n t do    gc u M  p p     i      a 3( ,5 ;' ')
n   n t do    gc u M  p p     i      k a 3( ,6 ;' ')
n   n t do    gc u M  p p     i      a 3(7 , ;' ')
n   n t do    gc u M  p p     i      a 3(8 , ;' ')
n   n t do    ga u M  p p     i      9a 3( , ;' ')
n   n t do    gc u M  o p p     i      a (4 ,0 ;' ')
n   n t do    gc u M  p p     i      pi a (4 , ;' ')
n   n t do    gc u M  p p     i      a (4 , ;' ')
n   n t do    gc u M  p p     i      a (4 , ;' ')
n   n t do    gc u M  p p     i      a (4 , ;' ')
n   n t do    gc u M  p p     i      a (4 , ;' ')
n   n t do    t  gc u M  p p     i      a (4 ,5 ;' ')
n   n t do    gc u M  u p p     i      a (4 ,6 ;' ')
n   n t do    gc u M  p p     i      a (4 , ;' ')
n   n t do    w gc u M  p p     i      a (48 , ;' ')
n   n t do    x gc u M  p p     i      9a (4 , ;' ')
n   n t do    gc u M  p p     i      a (50 y , ;' ')
n   n t do    ge u M  p p     i      a (5 , ;' ')
n   n t do    gc u M  p p     i      a (5 ,10 ;' ')
n   n t do    gc u M  p p     i      a 13(5 , ;' ')
n   n t do    gc u M  p p     i      a (5 , ;' ')
n   n t do    gc u M  p p     i      a (5 , ;' ')
n   n t do    gc u M  p p     i      a (5 ,64 ;' ')
n   n t do    gc u M  p p     i      a (75 ,15 ;' ')
n   n t do    gc u M  p p     i      a (8 ,6 ;' ')
n   n t do    gc u M  p p     i      9a (57 , ;' ')
n   n t do    gc u M  p p     i      a (6 ,8 ;' ')
n   n t do    gc u M  p p     i      a 9 (6 , ;' ')
}
}

```

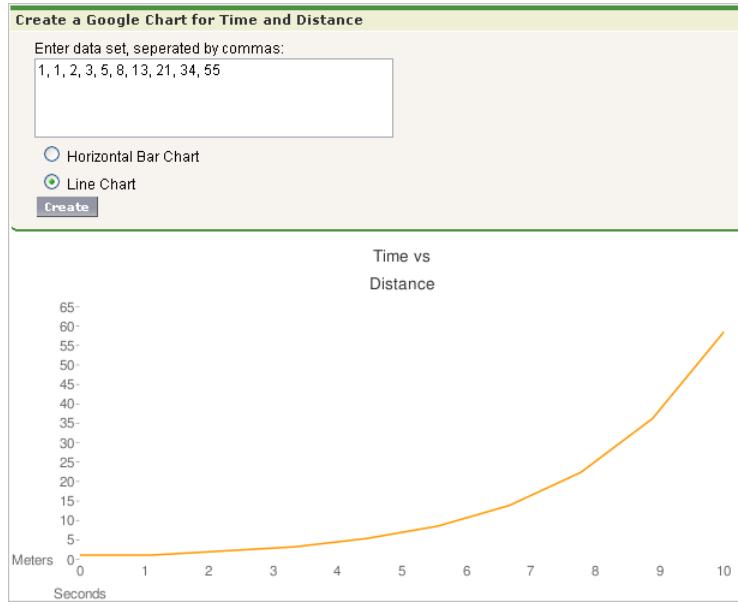
Visualforce ページには 2 つの入力要素 (グラフの種類とデータセットのそれぞれに 1 つずつ) が必要です。以下は、この情報を収集するフォームを構成するサンプルページです。

```

x t e n      egorplåtpc  oñ< gn  ado a Ejc =''  E      a ai " >
x ex ro m p f < a >
ex B lgo p p c < ka a
e t e e t l e r t Cde g r intec ='' nf a aT ca i D á s "a >
e x t te ou ubb p < a L a
e e t et nlt e r de tE rv d b a" op mm c a :> a a y s 'a
e x et net x uA p p k a
t tnd tdt ut ti rpt ni ='' o aa" i ='' Eds aia P"
e { et lt u d}w r o vlo a" clbr a a" s= 3" s= " 58 / > /
e e ex t el{ d po lc u } g as h R pi va a" ! "a
t e d du nrg o p c ya= D ia i " >
e e ex t t e lne po c lmpbr as qiv i a" a "a
e t e b m ro dBl r riL k a" i a a "a > /
e e ex t t e lne po c lmpil as oiv i a" i "a
e t e e b m r k il L a" i "a > /
e e ex t l d po c < /as R i a>
e x tm obndmm enoe t o < )rc a c a ia =! "a
e e et l ur r C v a" "a > /
ex B lgo p p c < k/a a >
x ro m p f < /a >
e { tmrd p t)re < dh l t a l ma r= U ph a " =a Sa "a
e e { n r d r d td l } r pC ='' lis ya 'a > /
g p p < /a a

```

サンプルでは、「 2 1 1 2, , 3 」という数字のシーケンスを入力します。このページは、次のように表示されます。



カスタムリストコントローラによるレコードの一括更新

一括更新を実行するページを作成するには、StandardSetController クラスに含まれるプロトタイプオブジェクトを使用します。

リストコントローラは、2つのレコードセットを追跡します。1つは、条件検索で選択されたすべてのレコードが含まれるプライマリリストで、もう1つはユーザが選択したレコードが含まれるセカンダリリストです。セカンダリリストは通常、ユーザがチェックボックスをオンにしてレコードを選択できる、標準リストビューページで設定されます。設定後、ユーザはカスタムリストボタンをクリックしてカスタム一括更新ページに移動できます。カスタム一括更新ページでは、プロトタイプオブジェクトが使用されて新しい項目値がユーザの選択したレコードに適用されます。プロトタイプオブジェクトは、ユーザが選択したすべてのレコードを操作します。カスタムコントローラでプロトタイプオブジェクトを取得するには、StandardSetController の `c` メソッドを使用します。たとえば、Opportunities の一括更新を有効にするには、その関連付けられたオブジェクトの単数形

R

を取得する (`opp`) を使用して、`c` 選択したすべてのレコードの項目値を設定します。

たとえば、1. `c` という Visualforce ページを作成します。

2. 次のコントローラを指定します。

```

  e e b l e t l e   p l e c d   n r o r o d i   s s s s   W   i   S   a
  e x A t   e n t g t e p d e   t o r l d n C   o B a .   S @ a S   s   ;
  e e e t k h   l z   p o c e   n r o x r A d t   e i n s   g t e p d i d S   t o e r m p r C o r l a r c S a a S
  t   n t e n o   o r d d r c   s   =   ; 
}

e e t   e b l u t e e e t p g { g l z   d M   i c i   y s   i   s ( ) 
t   t n e r e t m e t o e g C z d   c   s   .   s   ( $ )   ;( )
}

e e t   e b l u t e   r p g { g z r d   M   c i   i   R y   si   s ( ) 
t   t n e r e t m   o e   g r d   c   s   .   R   s   ( $ )   ;( )
}

```

3. 次のマークアップを指定します。

```

g p p < a a
n t e n d r d o r t l d t G h r o u s p a a ='' O i " y
e t r r d t e t c n o r o u p p V S ='' a i i s"
x e n e n e t e o l z d n r o r o d $ s ='' s W i s a "
ew e o d r h l f s a" s'a
d u o m pp i ='' "
>
x ro d m p u f m m < f a i ='' " >
e exe e g p opM < a ass a
ee e e t u r n m lt d { m d e e o e s t c e ='' y l m s: ) d ci i s ! y s i s"
e e t m o f s i ='' "
d u m m p i ='' s"
> /
e exe e g p opM < a ass a
ee e e t u r n m { r d z e c e s z m r d ='' R c si s ! Ry si s"
e e t m o f s i ='' "
> /
e ex tB l go t p l p nc r e e u t k a p p a t M d o ='' m @ d l u s s a - " a ='' i " i ='' "
e exe e g p opM < a ass sa > /
e ex e tB l g a p p o d s u m k a a s i ='' s" >
e e x et n e { u l d p t p t n e s r e u a p t g e m d n a ! g i m s a " a i ='' s a
e ex x t n e t e d u p r o e n e s t p p n e a r g i n h a d r S t v a l m d l c ' a ='' s a
e x et n u l d p p < / a >
e ex e tB l g a p p o c < c k/a a s >
e ex tB l B g o t p u p m o < o k s t o a m b u m s i a ='' " i ='' " >
e ex x t t e B n d m m e p o l t u < e a o } t a v d b a' w s a i a ='' k " a i ='' s " a
e ex x t t e B n d m m e p o h t < l n e a C n o a} c t v l d e' b c u a i a e ! a " i ='' "
e ex tB l B g o p u p c < k/a a s >
e ex tB e d g o t p l p c t & t d n k a r o c e u a e t p p t u l m s d l o c i i s " i ='' s i s "
e ex e B e l { g o e d l p t c u < T k a d a r t o v a d a' u b m s p p " v ='' a " i ='' " a
e ex x e n l o u e h p u n o < e m d a p p o v m a" p p . " a i ='' " a > /
e ex x e n l o u e h p u e o < } t a p p d v o a" g p p a " a i ='' s " a > /
e ex x e B l g o p p c < T k/a a a >
x ro m p f < / a >
g p p < / a >

```

4. [設定] から、[カスタマイズ] > [商談] > [ボタン、リンク、およびアクション] をクリックします。
5. [新規ボタンまたはリンク] をクリックします。
6. ボタン表示ラベル] を「フェーズの一括更新」に設定し、名前を「M g p []」に設定します。
7. 表示の種類] を「リストボタン」に設定し、チェックボックスの表示 複数レコード選択用] がオンになっていることを確認します。[動作] を「サイドバーを持つ既存のウィンドウで表示」に設定し、[内容のソース] を「[ルート] ページ」に設定します。作成したページの名前をクリックしてこのボタンに関連付けます。
8. [保存] をクリックします。
9. [設定] から、[カスタマイズ] > [商談] > [検索レイアウト] をクリックします。次に、[商談] リストビューの横にある [編集] をクリックします。
10. [カスタムボタン] の下で、[フェーズの一括更新] ボタンを [選択したボタン] リストに移動します。
11. [保存] をクリックします。
12. [商談] タブをクリックします。変更するいくつかの既存の商談を表示する検索条件を選択するか、作成します。
13. 各結果の横にチェックボックスが表示されます。任意の数のチェックボックスをクリックし、[フェーズの一括更新] ボタンをクリックして、選択したフェーズを目的の値に変更します。
14. [保存] をクリックします。

この例では、1つの項目を更新する方法を示していますが、プロトタイプオブジェクトの任意の数の項目を参照してユーザが選択したレコードに適用できます。プロトタイプオブジェクト内の、ユーザが設定しない項目は、選択したレコードに影響を与えません。プロトタイプオブジェクトでは、必須かどうかなど、項目のプロパティが保持されます。たとえば、`pp g m`などの必須項目についてページに入力項目を含めた場合、ユーザはその項目に値を入力する必要があります。



メモ: ユーザが選択または条件検索したセットのサイズをページが表示または参照するようにしたい場合、必要なのは `nro r od s` のみです。こうした表示は一括更新で変更されるセットに関する情報をユーザに提供するため、役立ちます。

第9章

Visualforce によるボタン、リンク、およびタブの上書き

Salesforce では、レコードの詳細ページの標準ボタンの機能を上書きできます。さらに、標準またはカスタムオブジェクトのタブをクリックすると表示されるタブのホームページも上書きできます。

標準ボタンまたはタブのホームページを上書きする手順は、次のとおりです。

1. 上書きするページに移動します。

- 標準オブジェクトの場合は、[設定] で [カスタマイズ] をクリックし、適切なオブジェクト(タブ)のリンクを選択して、[ボタン、リンク、およびアクション] をクリックします。
- カスタムオブジェクトの場合は、[設定] で [作成] > [オブジェクト] をクリックし、リスト内のいずれかのカスタムオブジェクトを選択します。

上書きするボタンまたはタブのホームページの横にある [編集] をクリックします。



メモ: 行動と ToDo には専用のタブはないため、上書きできるのは標準ボタンと標準リンクのみです。

2. 上書きの種別として [Visualforce ページ] を選択します。

3. ユーザがボタンまたはタブをクリックしたときに実行する Visualforce ページを選択します。

Visualforce ページでボタンを上書きするとき、ボタンが表示されるオブジェクトに対して標準コントローラを使用する必要があります。たとえば、取引先の [編集] ボタンを上書きするためにページを使用する場合、そのページのマークアップの `g p type` タグに属性 `or Id An Ca ou scc a` が含まれている必要があります。

```
t t n tegndprd por id An C e u acc a a =! " >
e t etne n go prc ! -- a >-
g p p < /a a
```

Visualforce ページでタブを上書きすると、コントローラのないタブ、カスタムコントローラのあるページ、またはコントローラのないページに対して標準リストコントローラを使用する Visualforce ページのみ選択可能になります。

Visualforce ページでリストを上書きすると、標準リストコントローラを使用する Visualforce ページのみ選択可能になります。

Visualforce ページで [新規] ボタンを使用して上書きする場合、[レコードタイプの選択] ページを省略することもできます。選択されている場合、Visualforce ページがすでにレコードタイプを処理していると想定されるため、新しいレコードを作成しても、[レコードタイプの選択] ページに転送されません。



ヒント: 上書きとして使用する Visualforce ページに機能を追加する必要がある場合は、コントローラ拡張を使用します。

4. 必要に応じて、この変更を行う理由をコメントとして入力します。
5. [保存] をクリックします。

上書きによって、ボタンの背後のアクションが制御されるため、ボタンの上書きは、Salesforce 全体に適用されます。たとえば、商談の [新規] ボタンを上書きした場合には、そのアクションが利用できるすべての場所でその代替アクションが有効になります。

- 商談のホームページ
- 取引先など、他のオブジェクトにあるすべての商談関連リスト
- サイドバーの [新規作成] ドロップダウンリスト
- この Salesforce ページのすべてのブラウザブックマーク

上書きを取り消す手順は、次のとおりです。

1. 上書きするページに移動します。
 - 標準オブジェクトの場合は、[設定] で [カスタマイズ] をクリックし、適切なオブジェクト(タブ)のリンクを選択して、[ボタン、リンク、およびアクション] をクリックします。
 - カスタムオブジェクトの場合は、[設定] で [作成] > [オブジェクト] をクリックし、リスト内のいずれかのカスタムオブジェクトを選択します。
2. 上書きの横にある [編集] をクリックします。
3. [上書きなし(デフォルトの動作)] を選択します。
4. [OK] をクリックします。

標準リストコントローラを使用したタブの上書き

標準リストコントローラを使用するページを使用してタブを上書きすることができます。たとえば、取引先標準リストコントローラに関連付けられた `Account` という名前のページを作成するとします。

t	n t e gnd prd por h An Ce a u r accd a a mc et o a' bc e l n v o a = acc a s' a Sy=" a '
x	B l go p p c < k a a >
x	e B el{go pbl p lct u s T k oai a ex va a' ! a s "v = a" >a
ex	e h lo ue hmu } < m a v a' ! . a 'a > /
	e B l go p p c < T k/a a a>
	B l go p p c < k/a a >
	g p p < /a a

[取引先] タブを上書きして標準の [取引先] ホームページの代わりにそのページを表示できます。

[取引先] タブを上書きする手順は、次のとおりです。

1. [設定] から、[カスタマイズ] > [取引先] > [ボタン、リンク、およびアクション] をクリックします。
2. [取引先] タブの [編集] をクリックします。
3. [Visualforce ページ] ドロップダウンリストから、overrideAccountTab ページを選択します。
4. [保存] をクリックします。



メモ: 適切なページレベルセキュリティを設定して、このページをすべてのユーザが使用できるようにしてください。

Visualforce のカスタムボタンおよびリンクの定義

カスタムボタンまたはカスタムリンクを作成する前に、ユーザがそのボタンまたはリンクをクリックした際に実行するアクションを決定します。

- [設定] で [カスタマイズ] をクリックし、該当するタブまたはユーザリンクを選択してから、[ボタン、リンク、およびアクション] を選択します。ユーザオブジェクトまたはカスタムホームページでは、カスタムボタンを使用できません。

カスタムボタンとカスタムリンクは、ToDo または行動の個々の設定リンクでの活動でのみ使用できます。ただし、[設定] で [カスタマイズ] > [活動] > [活動のボタン] をクリックして、ToDo と行動の両方に適用されるボタンを上書きできます。

カスタムオブジェクトの場合は、[設定] で [作成] > [オブジェクト] をクリックして、カスタムオブジェクトを選択します。

- [新規ボタンまたはリンク] をクリックします。
- 次の属性を入力します。

属性名	説明
表示ラベル	カスタムボタンまたはカスタムリンクのユーザページに表示されるテキストです。
名前	差し込み項目からの参照に使用されるボタンまたはリンクに付けられる一意の名前です。この名前は、アンダースコアと英数字のみを含み、組織内で一意の名前にする必要があります。最初は文字であること、空白は使用しない、最後にアンダースコアを使用しない、2つ続けてアンダースコアを使用しないという制約があります。
名前空間プレフィックス	パッケージコンテキストでは、名前空間プレフィックスとは AppExchange にある自社パッケージとそのコンテンツを他の開発者のパッケージと区別するための1~15 文字の英数字で構成される識別子です。名前空間プレフィックスでは、大文字小文字は区別されません。たとえば、ABC と abc は一意として認識されません。名前空間プレフィックスは、すべての Salesforce 組織にわたって必ずグローバルに一意なものを指定します。名前空間プレフィックスを使用することで、自社の管理パッケージのみを管理できるようになります。
保護コンポーネント	保護コンポーネントは、登録者の組織で作成されたコンポーネントからリンク付けたり参照したりすることはできません。開発者は、今後のリリースで、インストールの失敗を心配することなく保護コンポーネントを削除できます。ただし、コンポーネントが非保護に設定され、グローバルにリリースされると、開発者は削除できなくなります。
説明	ボタンまたはリンクを区別するテキスト。システム管理者がボタンとリンクを設定するときに表示されます。

属性名	説明
表示の種類	<p>ページレイアウトのどこでボタンまたはリンクを使用できるようにするかを決めます。</p> <p>詳細ページリンク ページレイアウトの [カスタムリンク] セクションにリンクを追加する場合に選択します。</p> <p>詳細ページボタン レコードの詳細ページにカスタムボタンを追加する場合に選択します。 詳細ページボタンは、ページレイアウトの [ボタン] セクションにのみ追加できます。</p> <p>リストボタン リストビュー、検索結果レイアウト、または関連リストにカスタムボタンを追加する場合に選択します。 リストボタンは、ページレイアウトの [関連リスト] セクションまたは [リストビュー] レイアウトと [検索結果] レイアウトにのみ追加できます。</p> <p>リストボタンの場合、Salesforce は自動的に [チェックボックスの表示(複数レコード選択用)] オプションを選択します。 このオプションは、リスト内の各レコードの横にチェックボックスを表示し、リストボタンのアクションへ適用するレコードをユーザが選択できるようにします。 ユーザがレコードを選択する必要がない場合には、このオプションをオフにします(たとえば、ユーザを別のページに移動させるボタンなど)。</p>
動作	<p>ボタンまたはリンクをクリックした際の処理を選択します。</p> <p>適用できる場合は、一部の設定にデフォルト値を使用します。 たとえば、新規ウィンドウに表示を選択した場合の新規ウィンドウのデフォルトの高さは 600 ピクセルとなります。</p>
内容のソース	Visualforce ページを使用するには、「Visualforce ページ」を選択し、ドロップダウンリストからページを選択します。 ホームページのカスタムリンクとして Visualforce ページを使用することはできません。

4. 終了後、[Save (保存)] をクリックします。

保存後、編集を継続するには、[Quick Save (適用)] をクリックします。

指定された URL を表示するには、[プレビュー] をクリックします。

コンテンツを保存せずに終了するには、[キャンセル] をクリックします。

5. 新しいボタンまたはリンクを表示するように、該当するタブや検索レイアウトのページレイアウトを編集します。

ユーザ用のカスタムリンクを追加した場合、そのリンクは、ユーザの詳細ページの [カスタムリンク] セクションに自動的に追加されます。 詳細ページボタンはページレイアウトの [ボタン] セクションにのみ追加できます。

6. 必要に応じて、ユーザのブラウザのデフォルト設定とは異なる設定を使用してリンクまたはボタンを開くようにするウィンドウのプロパティを設定します。

標準リストコントローラを使用したカスタムリストボタンの追加

標準ボタンやリンクを上書きすることに加えて、標準リストコントローラを使用するページにリンクするカスタムリストボタンを作成することもできます。これらのリストボタンは、オブジェクトのリストページ、検索結果、および関連リストで使用できます。また、これらのリストボタンを使用すると、選択したレコードのグループに対してアクションを実行できます。選択されたレコードのセットを示すには、式を使用します。

たとえば、カスタムボタンを商談の関連リストに追加して、選択したレコードの商談フェーズと完了日を編集して保存できるようにする手順は、次のとおりです。

- ## 1. 次の Apex クラスを作成します。

```
e tble sh { txc c g iE ss a Ra S  
e t e sh tex epwg A t e nEt gt ednd Heship r C or launcS a a S )  
t e n e teor dd r c z g s Ra S ( ; 0)  
}
```

2. 次のページを作成します。
このページを「**index.html**」と名前を付けてください。

3. すべてのユーザがそのページを使用できるようにします。

- a. [設定] で、[開発] > [ページ] をクリックします。
 - b. oppEditStageAndCloseDate ページの [セキュリティ] をクリックします。
 - c. 適切なプロファイルを [有効にされたプロファイル] リストに追加します。
 - d. [保存] をクリックします。

- #### 4. 商談にカスタムボタンを作成します。

- a. [設定] から、[カスタマイズ] > [商談] > [ボタン、リンク、およびアクション] をクリックします。
 - b. [新規ボタンまたはリンク] をクリックします。
 - c. 表示ラベルを「フェーズと日付の編集」に設定します。
 - d. 表示の種類を「リストボタン」に設定します。
 - e. [内容のソース] を「[ページ](#)」に設定します。
 - f. [コンテンツ] ドロップダウンリストから [A t dplp o C \[hi を選択します。D s a](#) を選択します。
 - g. [保存] をクリックします。
 - h. ボタンはページレイアウトを更新するまで表示されないことを通知する警告が表示されます。[OK] をクリックします。
5. 取引先ページレイアウトにカスタムボタンを追加します。
- a. [設定] から、[カスタマイズ] > [取引先] > [ページレイアウト] をクリックします。
 - b. 適切なページレイアウトの [編集] をクリックします。
 - c. [関連リストセクション] で、[商談] をクリックしてから、[e](#) をクリックしてプロパティを編集します。
 - d. [カスタムボタン] セクションで、[利用可能なボタン] リストの [編集のフェーズと日付] を選択して [選択したボタン] リストに追加します。
 - e. [OK] をクリックします。
 - f. [保存] をクリックします。

取引先ページにアクセスすると、[商談] 関連リストに新しいボタンが表示されます。

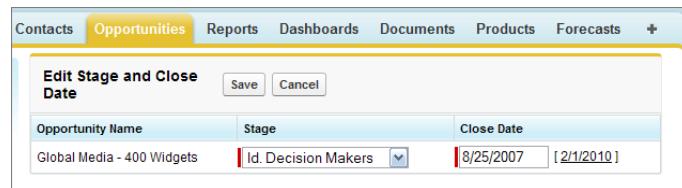


The screenshot shows a list of opportunities. At the top right of the list area, there is a new custom button labeled 'Edit Stage and Date'.

Action	Opportunity Name	Stage	Amount	Close Date
Edit Del	Global Media - 400 Widgets	Id. Decision Makers	\$40,000.00	8/25/2007
Edit Del	salesforce.com - 5000 Widgets	Closed Won	\$500,000.00	6/24/2007
Edit Del	salesforce.com - 500 Widgets	Closed Won	\$50,000.00	6/24/2007

図 23: 新規ボタンの例

商談を選択して、[編集のフェーズと日付] をクリックすると、カスタム編集ページが表示されます。



The screenshot shows a modal dialog titled 'Edit Stage and Close Date'. It contains fields for 'Opportunity Name' (Global Media - 400 Widgets), 'Stage' (Id. Decision Makers), and 'Close Date' (8/25/2007). There are 'Save' and 'Cancel' buttons at the top.

図 24: カスタム編集ページの例

レコードタイプの表示

Salesforce.com API バージョンが 20.0 以降の Visualforce ページではレコードタイプがサポートされます。レコードタイプを使用すると、異なるビジネスプロセス、選択リストの値、およびページレイアウトを、異なるユーザに提供できます。

[設定] でレコードタイプを作成すると、ユーザ側でそれ以上のアクションを実行しなくても Visualforce でレコードタイプのサポートが有効になります。レコードタイプを使用するオブジェクトの Visualforce ページには、ユーザによる設定が反映されます。レコードタイプ項目には、[d c I p](#) という名前が付けられます。

ユーザのレコードタイプ定義は、次のように `u ldp p` タグの表示に影響を与えます。

- `u ldp p` タグが、レコードタイプで条件検索される選択リスト項目を参照している場合:
 - 表示される `u ldp p` コンポーネントには、そのレコードタイプと互換性のあるオプションのみが表示されます。
 - `u ldp p` コンポーネントが、表示された編集可能な制御項目を持つ連動選択リストにバインドされている場合、レコードタイプと制御項目値の両方と互換性のあるオプションのみが表示されます。
- `u ldp p` タグがレコードタイプ項目を参照する場合:
 - ユーザが項目のレコードタイプを変更できるか、新規項目のレコードタイプを選択できる場合、`u ldp p` コンポーネントはドロップダウンリストとして表示されます。それ以外の場合は、参照のみのテキストとして表示されます。
 - リストが変更された際のページの更新または条件検索された選択リストの再表示は、開発者の責任で行ってください。

さらに、`u ldp p` タグのレコードタイプのサポートは、`u ldp p` の動作を参照のみとして実装した場合と同じです。

Visualforce ページで [新規] ボタンを使用して上書きする場合、[レコードタイプの選択] ページを省略することもできます。選択されている場合、Visualforce ページがすでにレコードタイプを処理していると想定されるため、新しいレコードを作成しても、[レコードタイプの選択] ページに転送されません。

第 10 章

静的リソースの使用

静的リソースにより、アーカイブ (.zip や .jar ファイルなど)、画像、スタイルシート、JavaScript、その他のファイルなど、Visualforce ページ内で参照できるコンテンツをアップロードできます。

[ドキュメント] タブにファイルをアップロードするよりも、静的リソースを利用する方がよい理由は、次のとおりです。

- ・ 関連ファイルを集めたものをディレクトリ階層にパッケージ化し、その階層を .zip や .jar アーカイブとしてアップロードできます。
- ・ ドキュメント ID をハードコードする代わりに、
 - グローバル変数を使用することで、ページマーケアップ内の静的リソースを名前で参照できます。



ヒント: さらに、静的リソースを使用して JavaScript またはカスケードスタイルシート (CSS) を参照する方が、マークアップをインラインに含めるよりも適しています。静的リソースを使用してこの種のコンテンツを管理すると、すべてのページや共有する一連の JavaScript 機能のデザインに一貫性を持たせることができます。

1 つの静的リソースのサイズは、最大 5 MB、1 組織が持てる静的リソースの合計は最大 250 MB です。

静的リソースの作成

静的リソースを作成する手順は、次のとおりです。

1. [設定] で、[開発] > [静的リソース] をクリックします。
2. [新規静的リソース] をクリックします。
3. [名前] テキストボックスで、Visualforce マークアップ内でリソースの識別に使われるテキストを入力します。
この名前は、アンダースコアと英数字のみを含み、組織内で一意の名前にする必要があります。最初は文字であること、スペースは使用しない、最後にアンダースコアを使用しない、2つ続けてアンダースコアを使用しないという制約があります。



メモ: Visualforce マークアップで静的リソースを参照し、そのリソースの名前を変更すると、Visualforce マークアップが更新されその変更が反映されます。

4. [説明] テキストエリアで、リソースの任意の説明を指定します。
5. [ファイル] テキストボックスの横にある [参照] をクリックして、アップロードするリソースのローカルコピーへ移動します。
6. 1 つの静的リソースのサイズは、最大 5 MB、1 組織が持てる静的リソースの合計は最大 250 MB です。
7. [キャッシュコントロール] を次のように設定します。

- 【非公開】は、Salesforce サーバにキャッシュされた静的リソースデータを他のユーザと共有しないことを指定します。静的リソースは、現在のユーザのセッションについてのみキャッシュに保存されます。



メモ: 静的リソースのキャッシュ設定は、ゲストユーザのプロファイルが IP 範囲またはログイン時間に基づいて制限されている Force.com サイトを介してアクセスする場合は、非公開に設定されます。ゲストユーザプロファイル制限のあるサイトでは、ブラウザ内でのみ静的リソースをキャッシュします。また、以前は無制限であったサイトに制限が設定されると、Salesforce キャッシュおよび中間キャッシュから静的リソースが解放されるまでに最大 45 日かかる場合があります。

- 【公開】は、Salesforce サーバにキャッシュされた静的リソースデータを、読み込み時間を短縮するために組織の他のユーザと共有することを指定します。

[ヘッダー項目定義に関する W3C 仕様](#)には、キャッシュ管理に関するより詳細な技術情報があります。



メモ: この機能は、サイト、つまり、静的なリソースを使用する有効な組織についてのみ有効です。

7. [保存] をクリックします。



警告: WinZip を使用する場合、必ず最新バージョンをインストールしてください。以前のバージョンの WinZip ではデータが失われる可能性があります。

Visualforce マークアップでの静的リソースの参照

Visualforce マークアップで静的リソースを参照する方法は、単独ファイルを参照するのか、またはアーカイブ(zip ファイルや .jar ファイルなど)に含まれるファイルを参照するのかによって異なります。

- 単独ファイルを参照するには、`<resource_name>` を差し込み項目として使用します。
`<resource_name>` は、リソースをアップロードしたときに指定した名前です。次に例を示します。

```
{ e   emrđe p t e ou < wđ ace g alIt# ! R s h g h h 'a i = " " 50 = " " 58 /
```

または

```
e n t el du rep c el a < p mai te Jic& M l a'b ! R sp . vy saia " > /
```

- アーカイブ内のファイルを参照するには、`ResourceReference` 関数を使用します。最初のパラメータには、そのアーカイブをアップロードしたときに指定した静的リソース名を、第 2 パラメータには、アーカイブ内での目的ファイルへのパスを指定します。たとえば、次のとおりです。次に例を示します。

```
{ e   mrdj p e e t< ou à ac U L ! R F Q p R s Z . s i , = " " 50 = " " 58 /
```

または

```
e n t el du rp c le a < e p a ou ieSov r c U Jld R F QbdRs l } i j fay s s/ a i i / . s
```

- 静的リソースアーカイブのファイルの相対パスを使用して、アーカイブ内の別のコンテンツを使用できます。たとえば、`c` という CSS ファイルには、次のようなスタイルがあるとします。

```
bl  <on r eit de mg k mg :a } mg ag i - fa s i a
```

Visualforce ページでその CSS を使用する場合、CSS ファイルが画像を検出できるようにする必要があります。そのためには、`img` や `mg sg ys .ss` を含む zip ファイルなどのアーカイブを作成します。パス構造がアーカイブ内に保持されるようにします。そして、アーカイブファイルを静的リソース「style_resources」としてアップロードします。ページに、次のコンポーネントを追加します。

静的リソースにはスタイルと画像が両方含まれているため、スタイルシートの相対パスが解決し、画像が表示されます。

- カスタムコントローラでは、`bl p` タグを使用して静的リソースのコンテンツを動的に参照できます。最初に、カスタムコントローラを作成します。

```
l bo g t el n or ld M C a ss a y
t e blt e re { p c g g m g Im i s Na a( )
} ntr ru t ru et g e ne ft he i dP i ; mg / si s a i a
```

次に、`ble pe` タグで `g vM i a` メソッドを参照します。

```
r g d pte nd <ortld m c aor ld M C " = " y " >
x e n re bl pre { m g el va i a vang vian a v a a il Na a > /
k e { nrd p e < e ou a m ac U P L ! R R s Z . iy V i, a a a ) > /
g p p < /a a
```

zip ファイル内で画像の名前が変わるのは、`mg Im` で返された値を変更できます。

第 11 章

カスタムコンポーネントの作成と使用

Salesforce には、Visualforce ページの作成に使用できる標準の組み込みコンポーネント（`input` や `button` など）のライブラリがあります。ユーザ独自のカスタムコンポーネントを作成して、このライブラリに追加することもできます。この章では、カスタムコンポーネントの概要と作成方法を説明します。

- ・ カスタムコンポーネントとは?
- ・ カスタムコンポーネントのマークアップ
- ・ Visualforce ページでのカスタムコンポーネントの使用
- ・ カスタムコンポーネントの属性
- ・ カスタムコンポーネントコントローラ
- ・ カスタムコンポーネントの定義

カスタムコンポーネントとは?

メソッドでコードをカプセル化すると、プログラムでそのメソッドを複数回利用できるのと同様に、カスタムコンポーネントで共通のデザインパターンをカプセル化することにより 1 つ以上の Visualforce ページでそのコンポーネントを複数回利用することができます。

たとえば、Visualforce ページを使用してフォトアルバムを作成します。アルバム内のそれぞれの写真の境界線は独自の色になっており、その下にはテキストキャプションが表示されます。アルバム内のすべての写真を表示する場合に Visualforce マークアップを繰り返すのではなく、画像、境界線の色、およびキャプションを配し

ます。一度定義すると、組織内のすべての Visualforce ページで、`blip` や `area` カスタムコンポーネントを活用できます。一度定義すると、組織内のすべての Visualforce ページで、`blip` や `area` カスタムコンポーネントを活用できます。

開発者がマークアップを再利用できるページテンプレートとは異なり、カスタムコンポーネントは、次の理由でページテンプレートよりも強力であり、柔軟性に優れています。

- ・ カスタムコンポーネントにより、開発者は、各コンポーネントに渡せる属性を定義できます。属性の値によって、最終ページでのマークアップの表示方法と、コンポーネントのそのインスタンスに対して実行されるコントローラベースのロジックを変更できます。この動作は、テンプレートのものとは異なります。テンプレートでは、テンプレートを使用するページからテンプレートの定義自体に情報を渡す方法がありません。

- カスタムコンポーネントの説明は、標準コンポーネントの説明と並んで、アプリケーションのコンポーネントの参照ダイアログに表示されます。一方、テンプレートの説明は、ページとして定義されているために、Salesforce の [設定] 領域からしか参照できません。

関連リンク

[カスタムコンポーネントの定義](#)

[Visualforce ページでのカスタムコンポーネントの使用](#)

カスタムコンポーネントの定義

Visualforce ページで使用するカスタムコンポーネントを定義する手順は、次のとおりです。

- Salesforce の [設定] で、[開発] > [コンポーネント] をクリックします。
- [新規] をクリックします。
- 表示ラベル テキストボックスに、設定ツールでカスタムコンポーネントの識別に使用するテキストを入力します。
- オブジェクト名 テキストボックスに、Visualforce マークアップ内でカスタムコンポーネントを識別するテキストを入力します。この名前は、アンダースコアと英数字のみを含み、組織内で一意の名前にする必要があります。最初は文字であること、スペースは使用しない、最後にアンダースコアを使用しない、2つ続けてアンダースコアを使用しないという制約があります。
- 説明 テキストボックスに、カスタムコンポーネント説明を入力します。この説明は、[保存] をクリックするとすぐに、他の標準コンポーネントの説明と共にコンポーネントの参照に表示されます。
- 内容 テキストボックスに、カスタムコンポーネント定義用の Visualforce マークアップを入力します。1つのコンポーネントに、最大 1 MB のテキスト、約 1,000,000 文字を入れることができます。
- [Version Settings (バージョン設定)] をクリックして、Visualforce のバージョンとこのコンポーネントで使用する API を指定します。また、組織にインストールされている管理パッケージのバージョンを指定できます。
- [保存] をクリックし、変更を保存してカスタムコンポーネントの詳細画面を参照するか、[適用] をクリックし、変更を保存してコンポーネントの編集を続行します。コンポーネントを保存するには、Visualforce マークアップが有効になっている必要があります。



メモ: カスタムコンポーネントは、[Visualforce の開発モード](#)で、Visualforce ページマークアップにまだ存在しないカスタムコンポーネントへの参照を追加することによって作成することもできます。マークアップを保存すると、コンポーネントに指定した名前に基づいて(指定の属性を含む)新規のコンポーネント定義を作成するためのクイック修正リンクが表示されます。

たとえば、`om_c_p` というカスタムコンポーネントがまだ定義されていない場合に、既存のページマークアップに `om_c_p` を挿入して [保存] をクリックすると、クイック修正によって `om_c_p` という名前の新規カスタムコンポーネントを定義できます。このコンポーネントのデフォルト定義は次のとおりです。

```

<apex:component>
    <apex:attribute name="c" type="String" />
    <apex:attribute name="p" type="String" />
    <apex:outputText value="Hello, {0} {1}!">
        <apex:param value="World" name="c"/>
        <apex:param value="!" name="p"/>
    </apex:outputText>
</apex:component>

```

この定義は、[設定] で [開発] > [コンポーネント] をクリックしてから myNewComponent カスタムコンポーネントの横にある [編集] をクリックして編集できます。

コンポーネントが作成されたら、`h e: mySalesforceInstance // p nameOfNewComponent` でこれを参照できます。`mySalesforceInstance` の値は Salesforce インスタンスのホスト名 (ro fcc c 3 sa s) など)、`nameOfNewComponent` は、カスタムコンポーネント定義の [名前] 項目の値です。[

コンポーネントは、一見 Visualforce ページのように表示されます。そのため、コンポーネントが属性またはコンポーネントタグ本体のコンテンツに依存している場合は、この URL から予測と違う結果が生じる場合があります。より正確にカスタムコンポーネントをテストするには、コンポーネントを Visualforce ページに追加してからページを表示してください。

カスタムコンポーネントのマークアップ

カスタムコンポーネントのすべてのマークアップは `om px p` タグ内で定義されます。このタグはカスタムコンポーネントの定義の最上位のタグである必要があります。次に例を示します。

```
t n n o om px p < a >
b < >
x t e t x e o u u p l up t <T e h a m t n b n o n o m e' i s p s y s ". > /
b < >/
t n n o om px p < /a >
```

他の Visualforce ページと同様に、マークアップは Visualforce と HTML タグの組み合わせを使用することができます。

さらに複雑な例では、カスタムコンポーネントを使用して、複数の Visualforce ページで使用するフォームを作成できます。`l c p` という新しいカスタムコンポーネントを作成して、次のコードをコピーします。

```
x t t e t e en b u e p r m trd< el n mta ie c e p e w d r e "rd w s p i f r n d =! gy
e t e te e b ej r p n d r u y" o " i =! " > /
x e x t te t el go{ p e p n c < ng ard a a d i =! i i ! " >
e x t d l l p < a i a > /
B l go p p c < k/a a >
t n n o om px p < /a >
```

次に、`rd p c` というページを作成して、次のコードを使用します。

```
er rd g p p < a a>
c l cr t rd p < ou Dcc i s ya =! " > /
g p p < /a a >
```

この例が正しく機能するためには、Visualforce ページを URL の有効な取引先レコードに関連付ける必要があります。たとえば、`I` が取引先 ID の場合、次の URL を使用します。

```
h e: p Salesforce_instance // pou dptcc /ai s/ ja i 1 00 000B003 5
```

ID として渡された取引先の詳細を含むページが表示されます。

ここで、`rd p c` のコードを次のサンプルで置き換えます。

```
rd g p p < a à
rd cl dr tnrd p & c DC ics ya = "a > /
g p p < /a à
```

同様に、ページを更新する前に取引先責任者の ID を渡します。取引先責任者の情報が表示されたページが表示されます。

[カスタムコンポーネントの属性](#)には、[コンポーネントの使用](#)についての詳細が含まれます。

Visualforce ページでのカスタムコンポーネントの使用

タグの本文は、コンポーネントを含むと必ず標準の Visualforce ページに追加されるマークアップです。たとえば、次の Visualforce ページは [カスタムコンポーネントのマークアップ](#)(ページ 140)(次の例では、コンポーネントは `c p` という名前で保存されています)で定義されているコンポーネントを使用します。

```
tnn o om p
tnn t egnd rd por h m C au acc a a =
tnn m om: C p < i si s < y > < à > /.
tnn g p p < /a à
```

出力は次のようにになります。

```
h m T page i si s y .
This is my custom component.
```

Visualforce ページでカスタムコンポーネントを使用するには、コンポーネントが定義された名前空間をコンポーネント名のプレフィックスとして付ける必要があります。たとえば、`e` という名前空間で `My Sp` という名前のコンポーネントが定義されている場合は、そのコンポーネントを `om: C p <` として `My Sp` Visualforce ページで参照できます。

便利な点は、関連付けられているページとして同じ名前空間で定義されているコンポーネントも、名前空間プレフィックスを使用できることです。これにより、上記のサンプルのページとコンポーネントが同じ名前空間で定義される場合、コンポーネントを `om: C p <` として参照できます。

カスタムコンポーネントにコンテンツを挿入する場合は、`Bm plo p <` タグを使用します。



関連リンク

- [カスタムコンポーネントとは](#)
- [カスタムコンポーネントの定義](#)

カスタムコンポーネントのバージョン設定の管理

Visualforce ページまたはカスタムコンポーネントに Salesforce API および Visualforce のバージョンを設定する手順は、次のとおりです。

1. Visualforce ページまたはコンポーネントを編集して、[バージョン設定] をクリックします。



メモ: [設定] の [開発] からページまたはカスタムコンポーネントを編集する場合にのみ、ページまたはカスタムコンポーネントのバージョン設定にアクセスできます。[開発モード] で編集する場合は、バージョン設定にアクセスできません。

2. Salesforce API の **バージョン** を選択します。このバージョンは、ページまたはコンポーネントで使用する Visualforce のバージョンも示します。
3. [保存] をクリックします。

関連リンク

[Visualforce のバージョン設定方法](#)

[Visualforce ページとコンポーネントのパッケージバージョン設定の管理](#)

カスタムコンポーネントの属性

標準の Visualforce マークアップ以外に、`om` `p` タグの本文でも、Visualforce ページで使用するときにカスタムコンポーネントに渡すことができる属性を指定できます。このような属性の値は、その後、コンポーネントやコンポーネントのコントローラ内(該当する場合)で直接使用できます。

`ttt ett 属性は b u p` タグで定義されます。たとえば、次のカスタムコンポーネントの定義では、
b rd および ro c という名前の 2 つの必須属性を指定します。これらの属性の値は、次の標準の `ttt ett` という Visualforce の式言語構文を使用してカスタムコンポーネントの定義で参照されます。

```

t n n o om p < a >
tt e teA rtb nu n o < f ! -- i D i i i s >-
xtt e t e m be u p m lmu d nra e tao e v the thy l 'a&#253; hso&#253; nin e' f&#253;nsi s v p a
et t e ne r et r q p n &#253; r u y" i S " i = " " > /
xtt e t e m b u p e bmr d r< boro da nida o c =ae tp le&#253;ro "r&#253; cs b rd f i si s
et t e ne r et r q p n &#253; r u y" i S " i = " " > /
e et n n ot om G p o < f ! -- D i i i >-
e { le h rd r b rd n &#253;ro s C y" ! ;" > /
x t et txe fu u p l up < Tlm u a v V a" ! y a" > /
t n n o om p < /a >

```

次のマークアップで Visualforce ページにこのコンポーネントを使用します。

```
t n n e m om: eC lm p <1 ue bMg rV loro r g C a" v y "a = " " > /
```

t r b eu p en t タグには `na et ia o c > p`、および `a s` 属性の値が必要です。 y

m 属性は Visualforce ページでカスタム属性を参照できる方法を定義します。属性の は、コンポーン
ネント内で一意にする必要があります。

d nr • `o c p` 属性は、カスタムコンポーネントが保存されたらコンポーネントの参照ライブラリに表示される属性のヘルプテキストを定義します。このカスタムコンポーネントは、使用可能な標準コンポーネントも含む参照ライブラリにリストされます。

• `属性は属性の Apex データ型を定義します。` 属性は次のデータ型のみを値として使用できます。

◊ string、integer、または boolean などのプリミティブデータ型

◊ Account などの sObject、My_Custom_Object__c、または汎用型の SObject

- ◊ String[]、Contact[]などの配列表記を使用して指定する一次元リスト
- ◊ `m_p` を使用して指定する対応付け。対応付けの特定のデータ型を指定する必要はありません。
- ◊ カスタム Apex クラス

`extent` の他の `attribute` 属性についての詳細は、[属性](#) (ページ 302) を参照してください。

デフォルトのカスタムコンポーネントの属性

カスタムコンポーネントには、必ず 2 つの属性が生成されます。これらの属性をコンポーネント定義に含める必要はありません。

`id`

ページの他のコンポーネントがカスタムコンポーネントを参照できるようにする識別子。

`rendered`

カスタムコンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの `true` に設定されます。

カスタムコンポーネントコントローラ

標準の Visualforce ページと同様に、カスタムコンポーネントを Apex で記述されたコントローラに関連付けることができます。この関連付けは、コンポーネントの `controller` 属性をカスタムコントローラに設定して行います。コントローラを使用すると、関連付けられているページにコンポーネントのマークアップを返す前に追加のロジックを実行できます。

コントローラでのカスタムコンポーネントの属性へのアクセス

関連付けられているカスタムコンポーネントコントローラでカスタムコンポーネントの属性の値にアクセスする手順は、次のとおりです。

1. 属性の値を格納するカスタムコンポーネントコントローラのプロパティを定義します。
2. プロパティの getter メソッドと setter メソッドを定義します。次に例を示します。

```
public class CustomComponentController {
    private String name;
    private Boolean isPublished;

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public Boolean getIsPublished() {
        return isPublished;
    }

    public void setIsPublished(Boolean isPublished) {
        this.isPublished = isPublished;
    }
}
```

setter が値を変更していることに注目してください。

3. コンポーネント定義の `attribute` タグで、`name` 属性を使用して定義したばかりのクラス変数に属性をバインドします。次に例を示します。

```
<apex:component controller="CustomComponentController">
    <apex:attribute name="name" type="String" default="My Component" />
    <apex:outputText value="{!name}" />
</apex:component>
```

```

e      e      tda n   oe om {d u p< >
e      br      < t ecn > v om} d & ap / i s" ! v       a"
e      te n doe     ored& r &l u dte & n & > oN l& n cl < u a / i s" ! v       a"
ex      B      l go p p c < k/a a >
t      te t o te n & hore d n r c le Mn ie b huar eV d en uxptA g s a do p sh a s a
e      t n n o om & p < /a >

```

属性を使用する場合は、getter メソッドおよび setter メソッドまたは 値および 値を含むプロパティを定義する必要があります。

- ページにコンポーネントを追加します。次に例を示します。

```

g      p p < a >
e      t m n o: dne n po & <{d use p r te h & r } H a'mi $ U , ! s E s N "a > /
g      p p < /a >

```

ページの出力は次のようにになります。



大文字で表示されるように Apex コントローラメソッドが controllerValue を変更している点に注目してください。

第12章

動的 Visualforce バインド

動的 Visualforce バインドとは、レコードに関する情報を表示する際に必ずしも表示する項目を指定しない汎用的な Visualforce ページを記述する方法です。つまり、ページ上の項目はコンパイル時ではなく実行時に決定されます。これにより、開発者は1ページだけ設計して、さまざまな利用者に対し、権限や設定に応じて異なる表示にすることができます。動的バインドは、わずかなコーディングで登録者ごとに固有のデータを表示できるため、管理パッケージに含まれる Visualforce ページで使用すると便利です。

動的 Visualforce バインドは、標準オブジェクトとカスタムオブジェクトでサポートされます。動的バインドは、一般的に次のフォームを取ります。

```
reference expression] [
```

構成要素について説明します。

- `reference` は、sObject、Apex クラス、グローバル変数のいずれかに評価されます。
- `expression` は、項目名または関連オブジェクト名である文字列に評価されます。関連オブジェクトが返されると、項目または追加の関連オブジェクトを再帰的に選択するのに使用できます。

動的バインドは、数式が有効な場所であればどこででも使用できます。次のようなページで使用します。

```
reference expression] [
```

必要に応じて、`fieldname` を動的式全体の最後に追加できます。動的式が sObject に解決されると、`fieldname` はそのオブジェクトの特定の項目を参照します。 `f c` が Apex クラスの場合、項目は `l pod` または `i` である必要があります。次に例を示します。

```
An C ouc cc } ]! fieldname [ ' ' [
```

動的 Visualforce ページは、ページ上のオブジェクトに標準コントローラを使用し、追加のカスタマイズは [コントローラ拡張](#) で実装するように設計する必要があります。

Apex の `c` メソッドを使用して動的参照、特に、オブジェクトの項目にアクセスする動的参照の情報を取得できます。たとえば `datej An e cau lfc S gis M .` は、Account 項目の名前 (`a`) のマップを、Apex コントローラと拡張が認識できる形式で返します。

! 重要: 静的参照は、ページを保存したときに有効かどうかチェックされ、参照が無効の場合はページを保存できなくなります。動的参照は、その性質上、実行時にのみチェックできます。ページを表示するときにそのページに無効な動的参照が含まれていると、ページはエラーになります。有効なカスタム項目またはグローバル変数への参照を作成できますが、その項目またはグローバル値が後で削除されると、ページは次回表示時にエラーになります。

リレーションの定義

`Object1__c` と `Object2__c` は両方とも、評価結果がオブジェクトリレーションになるような複合式にすることができます。たとえば、`Object1__c` というオブジェクトに別のオブジェクト `Object2__c` へのリレーションがあるとします。2つのオブジェクト間のリレーションの名前は、`Relationship__r` となります。

`Object1__c` に `Relationship__r` という項目がある場合、次の動的キャストルックアップはすべて同じ項目への参照を返します。

- `Object1__c.Object2__c['myField']`
- `Object1__c['Object2__c.myField']`
- `Object1__c['Object2__c'][myField]`
- `Object1__c.Relationship__r[myField]`
- `Object1__c[Relationship__r.myField]`
- `Object1__c[Relationship__r][myField]`

関連リンク

[グローバル変数への動的参照](#)

[グローバル変数](#)

標準オブジェクトでの動的参照の使用

アクセスする項目の既知のセットを使用して単純で再利用可能なページを構築するには、動的 Visualforce バインドを使用します。このアプローチには、どの項目がユーザの処理対象となるのかを容易にカスタマイズできるという利点があります。

次の2つの例は、説明のために意図的に簡略化されています。動的 Visualforce を十分に活用した高度な例は、[「ユーザがカスタマイズ可能なページでの動的参照の使用」](#) を参照してください。

単純な動的フォーム

次の例は、動的参照を使用する Visualforce ページを構築する最も簡単な方法を示します。

最初に、表示する項目の「動的な」リストを提供するコントローラ拡張を作成します。

```
blu   blu   ep cAn em tou ddcc i r Dss a yi a   L si s
      et An p en to ue d de x Art b nyite nd prd h Eo rs mpsr C dr l a rBa . S a a
      e or d d r c d el d d bl ld . Ha s i a s ; )
      t blu e n te p c { dg bl ld L i < s i s > i a s
      g
      te e d bl { f d li i a s == )
      te e d e bl wld t n r i g a s = L i < s i s > ; ( )
      te e d bl ld t n dd d ur Ii a s . a ( 's y , ')
      te e d bl le n d n n iu gi a s . a ( ' R a ; ')
      te e d bl ld d n ll
      te re r u d bl ld i a s ;
      }
      te t r p i a s ;
      }
```

次に、上記のコントローラ拡張を使用する `oudce` というページを作成します。

t e tx	n n	te gnd n	gnd p n	prd no	por et	id An	kn C en	au tou	acc ddcc	a sr Ds="1"	a yi	" a	E L si s	"> >
xe e			g p	gpMbr	<		a	ass ssk	> /	> /				
x e ex		ro te tB	m lg@	p lp	f tane	<	a	>						
e ex		ro te tB	m lg@	p lp	f tane	ok@ a docan d	i	" i	" i	" i	" i	" i	">	
e ex	x et n	e { u ldp lt@n	 te tB	lga p lp	clo @ tane	<	où mka	acc m@ a	v	a" ! i f	N "a s "v	>		
e ex	x et n	e { u ldp lt@n	 te tB	lga p lp	clo @ tane	<	où mka	acc m@ a	v	a" ! i f	N "a s "v	>		
e ex	t r	p p	<c	/a	>a									
ex		e tB	lga p lgo p	poc p c <	k/a k/a	a								
x	ro	m p f	<	/a	>									
		g p p	<	/a	>									

この例では次のようなことが行われます。

す。各文字列は、Account オブジェクト内の項目名に対応付けられます。

d bl • 1d リストはハードコードされていますが、項目はクエリや計算から特定したり、カスタム設定から読み取ったりすることができます。ハードコードされていない場合は、より動的な操作が提供されます。動的参照の利点は、このようなことが可能であることです。

マークアップは、`<pre>` タグを使用して文字列をループ処理します。

- **u ldp p** タグは、Account の項目名を表す **te** 反復要素を参照して、**1d** 内の各項目を表示します。動的参照 { を表示します。動的参照 CC] f が、実際に値をページに表示します。

標準コントローラによる動的参照の項目の読み込み

Visualforce は、ページの or は、reCn t て（または a oar 1d r C ）で実行される SOQL クエリを自動的に最適化し、ページで実際に使用される項目のみを読み込みます。オブジェクトと項目への静的参照を含む Visualforce ページを作成する場合、項目とオブジェクトは前もって知っておく必要があります。ページを保存するときに、Visualforce はどのオブジェクトと項目を SOQL クエリに追加する必要があるかを判別して保存できます。この SOQL クエリを、後でページが要求されたときに or 1d r C が実行します。

t 動的参照は実行時に、`or like` が SOQL クエリを実行した後に評価されます。動的参照でのみ使用される項目は、自動的には読み込まれません。動的参照が後で評価されたとき、その項目は存在しないデータとして解決され、結果的に SOQL エラーになります。読み込み対象の項目と関連オブジェクトがわかるように、コントローラに追加情報を指定する必要があります。

e ページコンントローラのメソッドを使用して読み込む追加項目のリストを渡すことで、
e nd rd or l d r C クエリにはくつでも項目を追加できます。前述の例では、これはコントローラ拡張のコン
ストラクタで次のように実行されます。

ten e or d r c d el d d bl ld . fa s a ;)

コンストラクタは、ページマークアップが使用するプロパティと同じプロパティを定義します。

これは、コントローラ拡張のインスタンス化時に読み込むすべての項目のリストがわかるページに適しています。項目のリストが、要求処理における後の時点まで判別できない場合は、コントローラで `toCall` をコールし、項目を追加できます。これにより、コントローラは修正されたクエリを送信することになります。この技法の例は、[ユーザがカスタマイズ可能なページでの動的参照の使用](#)を参照してください。

 メモ: コントローラへの項目追加が必要になるのは、`orList C` または `S a a` `orList C` にデフォルトクエリを使用する場合のみです。コントローラまたはコントローラ拡張が独自の SOQL クエリを実行する場合、`toCall` の使用は不要で、効果がありません。

これらのメソッドの詳細は、`orList C` のドキュメントを参照してください。

関連オブジェクトへの動的参照

この例では、ケースレコードの Visualforce ページを作成し、特定の項目を編集可能にします。表示される項目の一部は関連オブジェクトのもので、動的参照を使用してリレーションをトラバースする方法を示します。

最初に、`ApexController` という Apex シャドウコントローラ拡張を作成します。

```

public class ApexController {
    private Case c;
    private List<Account> accounts;
    private List<Contact> contacts;

    public ApexController(Case c) {
        this.c = c;
        accounts = [SELECT Id, Name, ParentId FROM Account WHERE ParentId = :c.Id];
        contacts = [SELECT Id, Name, AccountId FROM Contact WHERE AccountId IN :accounts];
    }

    public Case getCase() {
        return c;
    }

    public void updateCase() {
        c.Name = 'Updated Case';
        c.Status = 'Open';
        c.Description = 'This is a test case';
        c.AccountId = accounts.get(0).Id;
        c.ContactId = contacts.get(0).Id;
        c = Database.update(c, true);
    }

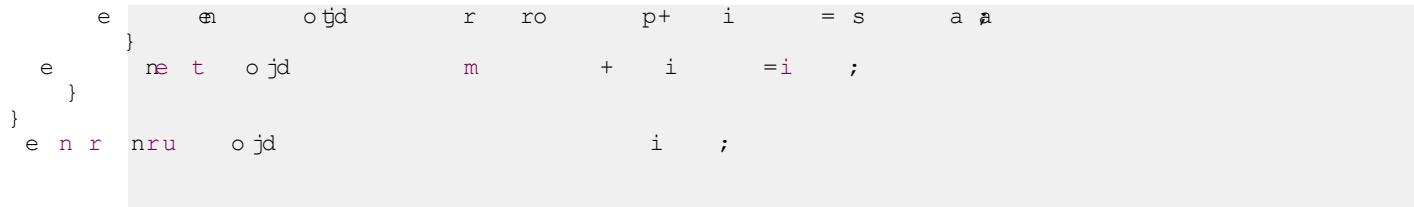
    public List<Account> getAccounts() {
        return accounts;
    }

    public void updateAccount() {
        accounts.get(0).Name = 'Updated Account';
        accounts = Database.update(accounts, true);
    }

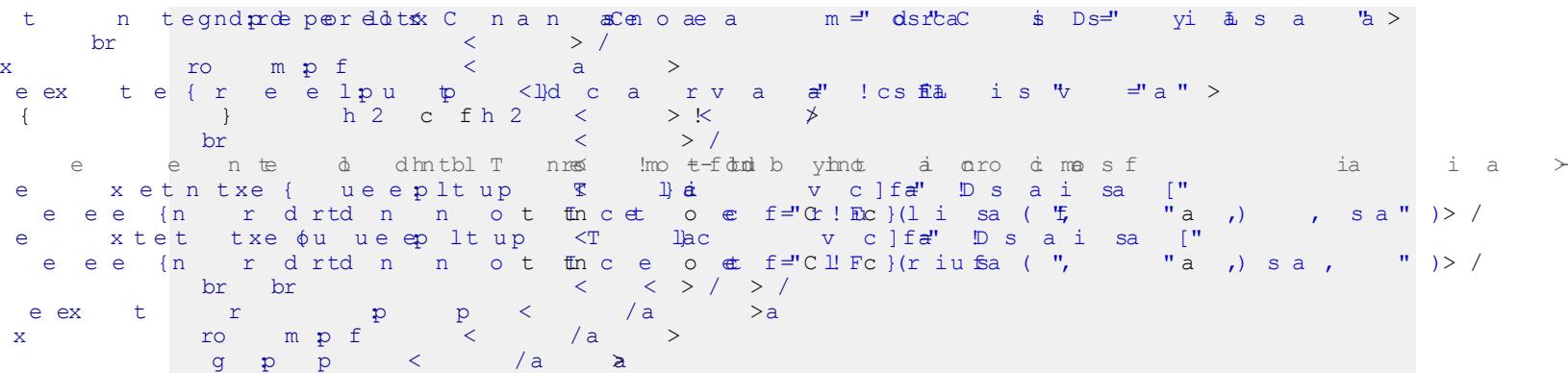
    public List<Contact> getContacts() {
        return contacts;
    }

    public void updateContact() {
        contacts.get(0).Name = 'Updated Contact';
        contacts = Database.update(contacts, true);
    }
}

```



対応するページはこの拡張を使用して特定のケースとそれに関連付けられた取引先責任者に関する情報を取得します。



このページに、クエリパラメータとして指定された、有効なケースレコードの ID を使用してアクセスします。たとえば、`CaseNumber` が `500 000000` のページに次のようなフォームが表示されます。

Contact.Name	Jon Amos
Contact.Email	info@salesforce.com
Contact.Phone	(905) 555-1212
CaseNumber	00001000
Origin	Phone
Status	Escalated

この例では、次の点に留意してください。

- コントローラ拡張では、オブジェクトを表示できるようにコンストラクタが独自の SOQL クエリを実行します。これは、ページの `CaseNumber` がデフォルトでは関連項目を読み込まないためですが、カスタマイズした SOQL クエリが必要になる使用事例は数多くあります。クエリの結果は、プロパティによってページで使用できるようになります。コンストラクタでクエリを実行するための要件はありません。プロパティのメソッドに簡単に追加できます。
- SOQL クエリは読み込む項目を指定するため、**単純な動的フォーム**では必要だた**を使用する必要はありません。**
- SOQL クエリは実行時に作成されます。ユーティリティメソッドが項目名のリストを SOQL EEC ステートメントでの使用に適した文字列に変換します。

- マークアップでは、フォーム項目は、pを使用する項目名を反復処理し、項目名変数cを動的参照で使用して項目値を取得することで表示されます。各項目はuppやauppの2つのコンポーネントで記述される可能性があります。これらのタグ上の表示属性が、2つのどちらを実際に表示するかを制御します。項目名が文字列「Contact」を含む場合、情報はuppタグに表示され、含まない場合はuppに表示されます。

ユーザがカスタマイズ可能なページでの動的参照の使用

Visualforce 動的バインドの最大の利点は、オブジェクトでどの項目が使用可能かを知らなくてもページを作成できることです。次の例はこの機能を示しています。すべてのオブジェクトで必須の _____ 項目を除き、Account オブジェクトの項目を一切知らずにカスタマイズできる取引先のリストが表示されます。これは、
_____ を使用してオブジェクトに存在する項目のリストを取得し、Visualforce 動的参照を使用することで可能になります。

この例で示す機能は単純です。メインリストビューには最初、取引先名のみが表示されますが、ユーザは[リストをカスタマイズ] ボタンを使用して、リストに追加する項目を選択できます。ユーザが設定を保存すると、リストビューに戻り、動的に生成された Visualforce ページの追加の列にそれらの項目が表示されます。

メモ: 項目を知らずにページを作成することは、[項目セットによる動的参照 \(ページ 159\)](#)を使用する方法でも可能です。

最初に `vi` `a` `s` というコントローラ拡張を作成します。

t p e c z m t u { b d m C i d l s r a y i a s i L a s a
A o m dt no lde @ R d s r o s a q ss ss
t e e p t g t d l p n d t d r a C o r l a n c S a a s ;
e en e g r f c v g a Ra v a Pa

t e e t h T h r o l f / h s i s s p a i s " " a
e n e e p t e g d e d w n t t m & S n i S r > g N sa= < S i S > ;()
e n e t p e g e d e d w n t v t m & n S i S > s g N sa= < S i S > ;()
n e n p e e g e b l v n t c t a S m S x g N sa= < S i S > ;()

t e z p m t e b d m C e x d l A r t y i n a t s g t d l p n d a s t o r m r C o r l a n c S a a s
o t d h n r c o r l d r c i s . = ;
d l d t b l h W F a V s i i s i i y()

e t l e l d o t l d f h f i a a a i s i s s
d t p d l d t { b d l h a i W F a V s i i s i i y()
n r e M t g p e m o d h j l d < c d d S , f s . a S F > i s =
t e m o d t e j A n e c t u b d m C S g a S M . p i s . a ()
n e r e e f g t { l d : f (i S k s i s . y S ())
e { e f n m w a n l s (N l ' a l ') / p i a s a s y a s y a
e n t e u l d d d S N s a . a (;)

e t e a d f g f e g r k e i A { h s b l s (D .) s i i s s s ()
n e e b l c c m d i a s s N s a . a (;)

e w e t h n o f l o h l / h i s s i i s
e t h e t T h r g e n e e d e / s i m s m r a c r f c m f y i a s
e n t p c e g { g l l d l i < s p i s D i s F y a s ()
n e r d e g l w n l d l p i s s e n e i t r e > i s g F y a s = L m i s s i s > s (N s a ;)
l t l d r o p i s F y a s s ;()
e r u d l l d p i s F y a s ;

t e o e g o u o n m c r W / C a s i s

```

e e e e b l e n t e r c { z u f o c m c i R a s i ( )
e e e e e x A e g t e n g r n p w c g a P a Ba . Pa; ( )
c e n r t r e z g u n o t m C m c Pa s d l y i a s ;
}

t t t n r t e r u w:d w// a i s v i
e e e e b l e n { g r c of c h i R a s ( )
e e e e e r o h t r e t f c n q w n A / $ h h d s l a f - y i s i s
t e n e e d r l d r x ;( )
t e n e e o r l a r c d d l e d g l l d p. f a s D ( i s F y a s ( ; ) )
c e n e r r u g w a Pa

t e t e e r t t C l r e h o t e w e r o t / p t a f l e h t h e i o s g h s p i s s a
t e b l i t t e l e p e e t o d q p i n d i < s o c S O i p > S O i s ( )
c e e n t t r u l e n e t e e t c o p l d s m O i s F S s ( N s a; )
}

t e b l i t t e l e p t e e n d c t g p l n d s o c O i p u S O i s ( )
c e e n t t r u l e n t e e o r a t o e p u l d s @ i s F S ( S N s a; )

t e r t t e d n t t o c l e n w p t a < s c o S p t O i r > s g O i s F p s ( S i s > s )
t t n t r n t e g o n t t L p i < s t r s > g L s i s s = L p i < s i s > ( s ; )
c e t t o t o r p i L s i s s ;( )
t t t t l t e a c o n s t o e p t < s p S l n O i > o i c s = p i < s S O i > ;( )
t t m o t r f t g o o o : p ( i s s i L s i s ) t @ o w n o d d p i s . a( )
e e t t e d n e t e t c d r p e n e m e S O i b l s ( n c n m o N a s ( c i , ) s s N s a . i s a s ( )
c t n r r m o o p i s;

t e r n e t r e p d t g r o { n c m i g a i s Na a ( i s s ) t e r n e r u e b l n c c n m o ci s s N s a . : * s a s ( ) ? ' s s;

t e t e e e r e o i t T p p t z c / u s o h m c o i r e t w b f d s p a i a k s a a a
t e t E m h t t r d h n d u / s t e h d t e t c t r > f h n k o i s h i c , s
c e e e t o e e r t f h m l c p e d l / t r o s m v r o f s h m i f i s s
t e b m n t t p e n e t r { e g t e d t s a c g < s i s > s ; s ;
t e b m n t t p c e n e e n t e g t e d t s a c g < s i s > s ; s ;

e e t d l t t m h e u c o l H t / e t a c g g d a k d a i p s . Pa s v a i a a
n e t r n d r o ro hm f // b l { d Ad p d i v i ( )
e e e o t m d e e l t e d le e d n t e f m u s s ( d s m N s a , s N s a ; )
}

e b l e { d d p o c o m i v i R v ( )
e e o e m t l d e e a l e n t d e u e l e e d t f e g l ( s d , m s N s a s N s a ; )

t e e d p t o m l d e n r v g i v m L s i s s i s s , s ,
t t e n e r t t g o m e o m e r { < r T g s d S m > v m < s i s > v F )
t e m o t { r f g m : ( i s s i s )
e e n e e f b l n c f n m o i d l i s s N s a . i s a s ( ) )
e e o m o d d T v . s ( ; )
e e e r e o m o r m o m v F . v s ( ; )
}
}

```

メモ: クラスを保存すると、Visualforce ページの欠落に関するプロンプトが表示される場合があります。
 この問題は、mc メソッドでのページ参照が原因です。[クイック修正] リンクをクリックしてページを作成します。コード内の後の個所にあるブロックから Visualforce マークアップが貼り付けられます。

- 標準コントローラメソッドの `get` および `set` が `hs` (メソッドで使用されます。) このメソッドにより、リストビューに戻ります。これらが必要なのは、表示する項目のリストが変更された可能性があり、表示用のデータを読み込むクエリを再実行する必要があるためです。
- `two` つの action メソッド `mc` と `hc` がリストビューからカスタマイズフォームに移動し、再び戻ります。
- navigation action メソッド後に行われることはすべて、カスタマイズフォーム関連の処理です。これらのメソッドは、コメントに注記されているように、大きく 2 つのグループに分けられます。最初のグループは、カスタマイズフォームで使用される `mc` がリストを提供し、`hc` は項目をリスト間で移動する 2 つのボタンを処理します。

ここで、次のマークアップで `mc` と `hc` を用いた Visualforce ページを作成します。

```

t n tegndprd por h AnCe atu r accd at arc o# cc " v s =" a La i s "
e tx n n not e z mt bmc $ dlsr yi a s i L s a " >
x br < > /
ro m p f < a >

e ew t ewnlte o d c gnu the h erdst orilh riC mo, s h S a a s >-
x B l go p p c < ka a >
x xt te eoueb pt l p tleAn waoulce cc wa tro a": S f V si " # il si s " > /
e ex t t d w ptdc z< e { as a t i sui # il rsds " s 1 # v a" hi " >
e ex t nte d uproe ne e app etargedad r S v bl # h aT = " a > /
e e ex t t len{ po etcl uwpl nas} ooi s p a hi si Oi s " > /
e ex t t l p c < /as L i s >
ex bl go p p c < k/a a >

t l t mo Tou e<fce ! u -shins lo u ancs s as i a s >-
x teB l go pt l a e sut k de m da i = s' = i " >
x exe e g p opM < a ass sa > /
x ex n etl re opd p bl< p h a Ta G i = " a >
e ex e B el{go pbl p lt u T k o a t ac r va a' bc La i s " v = " a " a >
e ex e n lo ut hpu } < mca v a' ! aN " a > /
e et e ne e hde T mrm < r ! c -i sf sc p yi a a >-
e e ex t e { r lep u p l < } ld a pr v a a' hi sf hiya s " v = " a " >
e e ex t e n lo ut hpu } < cca ]f v a' ! a " > /
e ex t r e B l go pbl p c < T k/a a a >
ex n l r op p < p /a a G >
ex B l go p p c < k/a a >

br < > /
x t@Bndmm ptoe l uz u {amC n tae v z ca os h i s " ia = ! s i " > /
x ro m p f < /a >
g p p < /a >
```

このページには、組織内の取引先のリストが表示されます。上部の `l go p p c` は、取引先用に定義されたビューの標準ドロップダウンリストを提供します。これらは、標準 Salesforce 取引先ページでユーザに表示されるのと同じビューです。このビューは、`for l d r C` で提供されるメソッドを使用します。

2 つ目の `l go ep pt c` には、`ka p a pe` で追加された列が含まれる `l go pbl p c < ta a a` が保持されます。繰り返しコンポーネントのすべての列は、取引先項目 `cc]f v` への動的参照を使用して、ユーザがカスタム選択した項目を表示します。

このミニアプリケーションの最後の部分はカスタマイズフォームです。`mc` と `hc` というページ `Li a s` を作成します。このページは、コントローラ拡張の作成時にすでに作成されている場合があります。次に貼り付けます。

```

t n tegndprd por h AnCe atu r accd an arc ro# " v s =" a La i s "
e tx n n not e z mt bmc $ dlsr yi a s i L s a " >
```

```

x     br      <      > /
x     ro m p f      <      a      >
x
x     teB e d go p l p c t < lkd a ceœa l ti d = " nl $B $ol $Dc i s c " ja = " sk i " >
x     exe e g p gM < a ass sa > /
x     ex n l r d p l o p u m c < a aiG s=" 3" >
e e ex t t le e en p d c t e l < ed l aœrL in di # s f i s " i = " s " a
e { e e e k u u d e d t elucm el v z a'ut s wt " l i & d = " " s : 2 " " ps0 y=" i
e e ex t t len{ pœ cœntu p t u l a n d } Odc s p a" ! s Oi s " > /
e e ex t t l p c < /as L i s >
e ex n l r o p p < p a a G >
e x t t e Bndmm p o l u < a a v a" > "
t { n e e Ad c o d e r r t d r n l iBa $l b c " c = " sk i " > /
e x t t e Bndmm p o l u < a a v < a" " > "
t { n e e o e e d e o e n e r r t d r n i a E" oll Rovc " c = " sk i " > /
e ex n l r o p p < p /a a G >
e e ex t t e l e e t p d e l e d l e a s L n d s i l = " s f i s " i = " s " a
e { e e d n t u u d e d t elucm el a'ut s wt " l i & d = " " s : 2 " " ps0 y=" i
e e ex t t len{ e p o e c t u p l n a s } ooi s p a" ls Oi s " > /
e e ex t t l p c < /as L i s >
ex n l r d p p < /a aIG >
e t e e m l d r n t r < e n M e l o m e r g s t a bl & su / & u ia cc m a s s y
ex B l go p p c < k/a a >

br < > /
x t t e Bndmm p e w e l a < o a h ldhna w o } a" c o S sh s" i a = " ls " > /
x ro m p f < /a a >
x g p p < /a a >

```

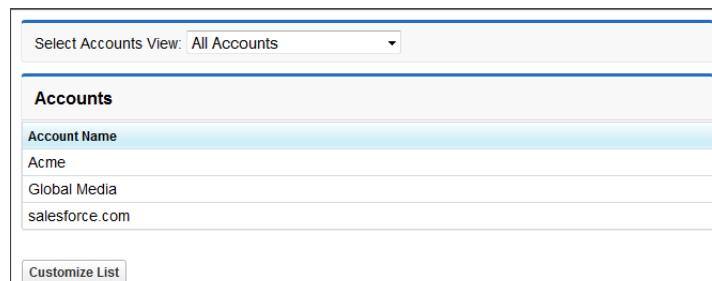
この単純な設定ページには、2つのリストが表示されます。ユーザは、左側の使用可能な項目のリストから、右側の表示する項目のリストに項目を移動します。[これらの項目を表示]をクリックすると、リスト自体に戻ります。

このマークアップでは、次の点に留意してください。

- このページは、取引先が表示されていなくても、リストビューと同じ標準コントローラを使用します。これは、表示する項目のリストが含まれるビューステートを維持するために必要です。このフォームでユーザの設定がカスタム設定のような永続的なものに保存された場合、この操作は不要になります。
- 最初のリストは、nd oc pU メソッドをコールすることで入力されます。フォームが(2つのt t oBndmm p o < コンボーネントのいずれかを経由して)送信されると、フォーム送信時に選択されたリストの値が c プロパティに保存されます。対応するコードが、もう一方のリストを処理します。
- これらの移動する項目の「デルタ」リストは、クリックされたボタンに応じて d o またはメソッドで処理されます。

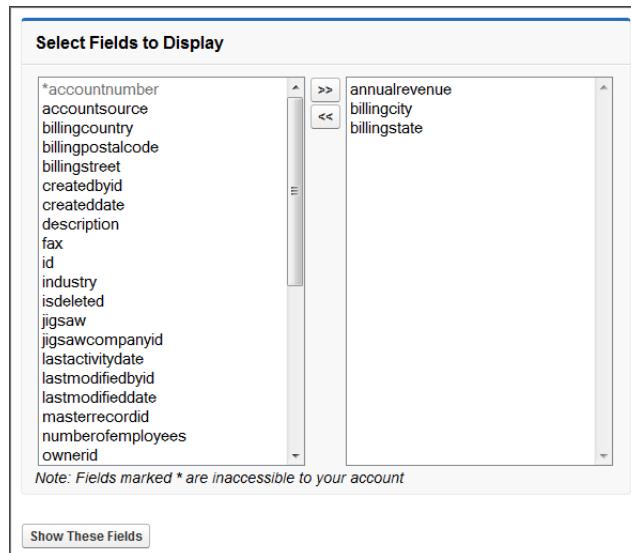
コントローラ拡張とこれらのページを作成し、組織内の z mtp o b m C D / a に移動すると、次に似たシーケンスが表示されます。

1. デフォルト状態のカスタマイズ可能なリストに取引先名項目のみが表示されます。



[リストをカスタマイズ] をクリックします。

2. 表示設定画面が表示されます。



一部の項目を右側のリストに移動し、[これらの項目を表示] をクリックします。

3. カスタマイズされたリストビューが表示されます。

Select Accounts View: All Accounts			
Accounts			
Account Name	Annual Revenue	Billing City	Billing State/Province
Acme	\$100,000,000	New York	NY
Global Media		Toronto	Ontario
salesforce.com		San Francisco	CA

[Customize List](#)

カスタムオブジェクトおよびパッケージでの動的参照の使用

パッケージ開発者は動的 Visualforce バインドを使用して、ユーザがアクセスできる項目のみをリストできます。これが必要になるのは、オブジェクトの項目を表示する Visualforce ページを含む管理パッケージを開発する場合などです。パッケージ開発者は登録者がアクセスできる項目がわからないため、動的ページを定義して登録者ごとに表示を変えることができます。次の例は、Visualforce ページを使用するページレイアウトと一緒にパッケージされたカスタムオブジェクトを使用して、異なる登録ユーザに同じページを表示する方法を示します。

1. Book という名前で次の項目とデータ型を持つカスタムオブジェクトを作成します。

```

A   u   ro   :Text(255)      i
B   r   :Text(13)           NS
r   blu  r   :Currency(4, 2)    i P
      h   :Text(255)        iPs

```

デフォルトでは、新規カスタムオブジェクトを作成すると、そのオブジェクトのレイアウトが作成されます。このレイアウトを Book Layout と呼びます。

2. レイアウトを変更し、上記のカスタム項目が表示されて、作成者、最終更新者、所有者、名前などの標準項目が削除されるようにします。
3. 新規カスタムオブジェクトタブを作成します。オブジェクトを Book に、タブスタイルを Books に設定します。
4. [Book] タブに切り替えて、Book オブジェクトをいくつか作成します。このチュートリアルでは、項目内のデータは実際には関係ありません。

例 5. 次のコードで `o_E` という名前のコントローラ拡張を作成します。

```
w   blu      n   prc e hlx g{ bn oon i   io E s i k   ss a      $  
te  ex A t     rp t e ndprd toer m d r Caor l d rBe . S a a ;  
te  t t     n e ip   e b geo wld t v   n&  sr iS >g   H   s =   < s iS > ;()  
t   e tlu  bneoon epx AotE   nk t eindprd toer m d r C d r l d rBe . S a a )  
t   ten   ot dñnr c or l d r c i s . = ;  
e   e te   re Metg p eho d h j ld< c d d S , fs . a s H >i s =  
e   e te   ho dnejB e oot l d Sc g f k M . p i s . a ()  
t   ne  re ef gt{ ld : f (i s ks i s . y s ())  
e   e nel el dñ c bl ccld Ø/ iyf ss i s  
e   e te dd ef gf eg r ke iA á( sbl sd(D.) si i s . ss ()&&  
e   e te dd e gf g t b{ á u s . m G(D.) si i s . ss ()  
e   b oo ld dd k H s . a ();)  
}  
}  
  
{ t   t blu      n   epce { g l bl l dL i <s i s > i a a H s  
t   te ee   dr l d r x . s ;()  
t   te n   e or l d r c ddn tl dt n e r b goa H L s L( i <s k i s > ( H s ; ))  
e   t e   nw m t rt n e r b goo ld L i <s k i s > ( H s ; )  
}
```

6. このコントローラ拡張を使用して Book オブジェクトの値を表示する、

という名前の Visualforce

ページを作成します。

```
t   n t e gndprd por edk tx oon n a e t e o abnkoon =" o E"__ k $ s=" " >  
ex br te t e l go p l p B c < > /  
e e ex t e { r d p e u p l b l p l da v a l a i afa N a >  
e   extet tx ho u p p <T> a  
e   e l t e u b j & < > /  
e   extet tx e fu u p le u p < > /  
e ex t B l go p p c < k/a /a >a  
g p p < /a >
```

7. コントローラ拡張はパッケージ化されるため、[Apex クラスのテストを作成](#)する必要があります。手始めに次の基本的なコードで `o_E_k` という名前の Apex クラスを作成します。

```

w     blu      n     prc e   tix gh( bn oon i   io E s i k   ss a       $ 
t     e ix A t     np t g ndprd toer m d r Caor l d rB . S a a           ; 
te    t t      ne ip e b gownd t v     ns & S r i S >g   I   s =     < s i S > ;( ) 
t     e tbd     bneoon epxc AotE     nk t g indprd toer m d r C d r l d rB . S a a 
t     te n     re Metg p eho d h j ld< c a d S , fs . a S   I   >i   s = ; 
e     e     te     ho dnej B e oot     l d S c g f & M . p   i   s .   ;( ) 
t     me     re ef gt{ ld : f   (iS ks i s . y S ( )) 
e     e     nel el dau c bl ccld &/ iyf   ss i s 
e     e     te ed ef gf eg r ke iA a( sbl s d(D .) s i   ( .s) s$ ( ) && 
e     e     te ed e gf g t b{ a u s . m G (D .) s i   ( .s) s   ( ) ) 
e     e     b oo ld dd k   I   s . ss ( ; ) 
} 
} 
t     en     e or l d r cddn tl dt     ne r     b goa I d l s L( i <s k i S > ( I   s ; )) 
{ t     t blu      n     eprce { g l bl     ldL i <s i S & i a a I   s 
t     tenee   dr l d r & 
t     tene   e or l d r cddn tl dt     ne r     b goa I d l s L( i <s k i S > ( I   s ; )) 
e     te     nw t rt     ne r     b goo ld     L i <s k i S > ( I   s ; ) 
} 
}

```



メモ: この Apex テストは、あくまでもサンプルです。パッケージに含めるテストを作成する場合は、肯定的な結果と否定的な結果を含む、すべての動作を検証してください。

- nb oo d8. という名前のパッケージを作成し、カスタムオブジェクト、Visualforce ページ、
e bn oon o E k Apex クラスを追加します。その他の参照される要素は自動的に含められます。
- nb oo d9. パッケージを登録者組織にインストールします。
10. パッケージをインストールしたら、[設定] で [作成] > [オブジェクト] をクリックし、[本] をクリックします。
g という新しい項目を追加します。
 11. 新規 Book オブジェクトを作成します。ここでも、レコードの値は実際には関係ありません。
 12. URL にパッケージ名前空間とブック ID を追加したものを使用して、
ページに移動します。たとえば、GBOOK が名前空間で、a00D0000008c7t4 がブック ID の場合、最終的な URL は
e he n ecorp x fc Be sc w b /Sg i dst a /la GOV K _Gsi になりますQ0 D0000003 5
- 登録者組織からページが表示されると、ページにはパッケージ化された Book 項目のすべてと、新しく作成された Rating 項目が含まれます。異なるユーザおよび組織は引き続き必要な項目を追加でき、動的 Visualforce ページは、適宜調整されて表示されます。

Apex 対応付けとリストの参照

動的バインドを使用する Visualforce ページは、マークアップ内の Apex M および データ型を参照できます。

たとえば、Apex が次のように定義されている場合、

```
t t b l e n e { p c g d l p p i < s i s >
g
c e n w m t r t { n t r n n g o l u j < s } r w i s > i s , ' i ' a ' i ' ; '
}
s ;

t e a b l e n e t p c r g l i < s i
g
c e n w m t r e t { n r g } l i < s i , 0 , ;
}
s ;
```

Visualforce ページでは次のようにアクセスできます。

```
t e { r e t l p u } p r < r a o v a p " il " v = " a s " >
x t e t t x e f u e p l up < T d } a p h r v p a ! s[ " < > / > /
t r p p < / a > a
```

同様に、次の Apex M があるとします。

```
t b l e t n r e p n c M t g p d g r a < c a i s , i s > i s
g
c e n w m t u l o { r n M g p g < a i s , i s >
e w l o o n l o d k i k si => ' P a , '
e n e d r n r c G y => F ' a ,
}
; s ;
```

Visualforce ページは、次のように値を表示できます。

```
t e { r e l p u f r } e r o a c d r v a a " li s " v = " a K y
x t e t t x e f u u e p l up d & T a v a " li K y > / --
x t e t t x e f u u e p l uped & T r o a d r c v br a " li s i [ K y < > / > /
t r p p < / a > a
```

組織のカスタムオブジェクトに含まれていないデータを使用してフォームを作成するには、`u p p` タグのリストと対応付けの動的参照を使用します。Apex コントローラに一連のインスタンス変数を作成したり、そのフォームデータのためだけにカスタムオブジェクトを作成したりするより、単一の対応付けを使用するほうが、かなり作業を簡略化できる場合があります。

カスタムコントローラで処理するフォームデータを格納する対応付けを使用する Visualforce ページを次に示します。

```
x t e n g a r p l a p c t e n < o a l d M r a C p L = " i s s s a " >
x t e x e e n o u u x p d t b p o l < o u b a o P a c = " " k y a = " " >
e e e x e r e g p g M < a ass sa > /
x ro m p f < a >

e e e x e t e { r e l p u p u l d e a p v l a l a " il f i s " v = " a K y
e e x t e t t x e f u u e p l up & T d a f v : a " li K y > / 
e e x t e n t x e { u p t l n u p e & u l d a l p v f b a " li i s i [ K y < > / > /
e e e x e t r p p < / a > a

e x t m o B n d m m p e n o e t o < t b u a m l d a i a = " ls i i D " a a
```

```

e      t l u bt e endre b u no r wx d r a"b oß " i = " " = " " > /
x t x ro m p f < /a >
t te nou u p p < /a a
g p p < /a a

```

フォームで使用する単純なコントローラを次に示します。

```

blt l tecñc or lMr Cp Lss a s s s a
t blt n rpn M g {p e g tauild < apig , is i E s ; s ;
t blt ten{ pc or lMr Cp L is s s a ( )
et n e unldt p n { r n M g p E gs = < aiS , is >
te r nnmf te oJ l e eei sAn e d=> d ' ypp' g Na a=> 2 s , ' a=> 41 ;
}
e e e eble n erc tbuf mld i Ra s i ED a(a)
t edtno at m tn g ht n I g u hSti p W si i ;( )
c n m ru lu ;
}
etn blt t de dtnocat m tn g ht n ii v gi u hSti p W si i ( )
t eu ld e stupn e g p ir gñue st e u ld f'a , 'g v g aoi no E m s f.+ ( 'a1 ') .0
}

```

M には、sObject または sObject 項目の参照を含めることができます。これらの項目を更新するには、入力項目の項目名を参照します。

```

blt n prc h4 An c io M cpcC i a ss a a
t An t An g I poou t e Aac mo w<ea sta pnc T , t An g I poou ec < a , > ;( )
t blt An p o M cpcC i a ( )
t An r g I ouf ec d E EtC M t { ou LaM cfs MII AR O L 1 0 )
t At mo ou pc T p a . i ( , ; a )
}
etblt n t An g I poou tcoAn & oam pcc,T > a ( )
n r rut An mo ou pc T a ; ;
}
}
x t en gorpld pcAn < o Ma cpcA = " a " >
x ex t e { r lpu tpAn m ou a pcc v a ue" mcc a "v = " a Na " >
x etn e { u ldp l p t An mo eau pcc v mcc a ! a Na N "a > /
x ex t r p p < /a >
x ro m p f < /a a
g p p < /a a

```

未解決の動的参照

動的参照が解決しない場合、実行時に次の問題が発生する可能性があります。

- 特定のキーに対応付けられた値がない場合、Visualforce ページはエラーメッセージを返します。たとえば、次のコントローラがあるとします。

```

blt t e n p doc or ldr CT i ss a
t blt n r ptn M g ep tegdot i < )mis p , is > a ; s ;

```

```
t      teln{ ex top cgtm lg i is y Ky ; s ;
t      ble n { do orcldr CT i
t      t de r tn M g p gda t < as ip,nim op sg =a < aiS , is > ;( )
t      t de eau M ltr pp ne p sz rga p gd sa , ' Ks i s ia ; ' }
}
```

このページでは実行時にエラーが発生します。

```
t      ten gorpltepa do< or ldr CT a =
e      e ne r d em blT < ! -is gh sp a a >-
x tet txe fu ut pl up n<do } dh M vp ] a! a [ 'Pai " ' > /
g      p      p      < /a   a
```

リキュー キーが **空** の場合、Visualforce ページは空の文字列を表示します。たとえば、上記と同じコントローラを使用すると、このページは空白を表示します。

```
t      ten gorpltepa do< or ldr CT a =
e      e ne r d em blT < ! -is scpk s a a >-
x tet txe fu ut pl up n<do } dh M vp ] a! a [ " " > /
g      p      p      < /a   a
```

項目セットの使用

 メモ: このリリースには、項目セットのベータバージョンが含まれています。本番品質ではありませんが、既知の制限があります。

動的バインドを使用して Visualforce ページに項目セットを表示することができます。項目セットとは、項目をグループ化したものです。たとえば、ユーザの名、ミドルネーム、姓、肩書を示す項目を1つの項目セットにして持つことができます。そのページを管理パッケージに追加すると、システム管理者は項目セット内の項目の追加、削除、並び替えを行って、コードを変更せずに Visualforce ページ上に表示する項目を変更できます。項目セットは、API バージョン 21.0 以降の Visualforce ページに使用できます。1 ページで最大 50 個の項目セットを参照できます。

Visualforce での項目セットの使用

項目セットを Visualforce で直接参照するには、`c $ グローバル変数をキーワード` と組み合
わせます。たとえば、Contact オブジェクトに 3 つの項目を表示する `mp p` という項目セットがある場合、Visualforce ページは次の反復によって項目データを参照できます。

```
t      n tegndprd pdr h C o a ac ca a =
t      t e { r e ltu pbt ej e adc ev ap ac lr Oor y mprp al N saiv =
x tet txe fu u pt ltnup < } ac dr yf a! a " < > / > /
ex t      r p p < /a >a
g      p      p      < /a   a
```

次の項目セット内の項目の特殊なプロパティを使用して、項目の表示ラベルやデータ型など、追加情報を表示するように選択することもできます。

プロパティ名	説明
required	オブジェクトでその項目が必須かどうか

プロパティ名	説明
ld	h 項目の範囲情報のリスト
bl	L 項目の UI 表示ラベル
n d	R 項目セットでその項目が必須かどうか
pt	Y 項目のデータ型

たとえば、`mp__p` の項目の表示ラベルとデータ型には次のようにアクセスできます。

```

t n tegndprd pdr h c o a ac ca a " " " " a>
x teB el go p lpe a lde kraor a in p=" si P N sa >
e ex e B el{go pble p lde u < Btk aej ea dc eva lsi ab lr Oon y mp rp a s f N sa"v
e x e n lo u mpx < fa v a" ! " >
e ex te ene e p end r ex ht m a a fa " a < a / a a>
e x n lo u mpx < /a >
e ex e n lo ue mpx } b k fa v a" L ! . a" >
e x n lo u mpx < /a >
e ex e n lo ue mpx } < fa pT v a" ! . y" >
e x n lo u mpx < /a >
ex e B l go pbl p c < T k/a a a>
B l go p p c < k/a a >
g p p < /a a

```

この Visualforce ページが管理パッケージに追加され、配布された場合、登録者は `mp__p` 項目セットを編集できます。Visualforce ページを生成するロジックは変わりませんが、表示は各登録者の実装に応じて異なります。管理パッケージの項目セットを参照するには、項目セットの先頭に組織の名前空間を追加する必要があります。上記のマークアップを使用すると、`mp__p` が Spectre という組織のものである場合、項目セットは次のように参照されます。

Apex での項目セットの使用

Visualforce ページで標準コントローラを使用するときには、項目セットの項目は自動的に読み込まれます。カスタムコントローラを使用する場合、ページの SOQL クエリに必須項目を追加する必要があります。Apex は、項目セットとそれが含む項目を検出できる `Schema` オブジェクトを提供します。これら 2 つのシステムクラスについての詳細は、『[Force.com Apex コード開発者ガイド](#)』の「FieldSet クラス」を参照してください。

サンプル: Visualforce ページへの項目セットの表示

このサンプルでは、Dimensions 項目セットに含まれるすべての項目を動的に取得します。取得した項目のリストを使用して、これらの項目を表示に使用できるようにする SOQL クエリを作成します。Visualforce ページは、クラスをコントローラとして使用します。

```

e blu ele n trc d M dhl i ssa d as i sa
e blu ner { ph e M tch rt m c igdn i as _ ; s ;
e blu n tr { ph c M dhl i d as i sa ( )
e et r nch n gch d M dhi s . = i as ;( )
} te blu e e e p e enlddet b{ r m M gld S E a S > d s ( )

```

```

    e n r t e reu b ej n er c ed t M e ch ld O es c et y . m o i ag l d _ 空 D $ si $ s . 空 s ;( )

    e n r e n e r et pd M e ch { n g ad Ma ch i 空 ( ) ;
    e e e e e o e te f m l d h t e b r e t m M { ( S g l d a s i s . 空 s ( ) )
    e e e e t n q t g l d f + h y = . 空 + Pa ( ) ;' , ' t n e r e r u e b n q n q D M M y h ' N c A R M I T i s L _ 1 ; ' t n e r e r u e b n q n q D a s a a y ( N ) ;
}

```

上記のコントローラを使用する Visualforce ページは単純です。

```

x t e n e g o r p l a e p o a t r < d Ma ch l a > " 空 i s a >
x ro m p f < a >

ex t e t B l g o p t l p t & do< u k a c a i = " B i s a >
e ex e t B leg a p p o t l & r do au a c s i = " P " >
e ex x e t n e { ue l d p le p & m a i m h E v a ! N " a > /
e ex e t B l g a p p o c < c k/a a s >

e ex e t B leg a p e p o c l n & n k a m o a s i D = " i s s >
e ex e t e { r e l p u p l d < r f a v a a ! li s " v = " a " >
e ex x e t n e { ue l d p le p & t m a l d h E f v f h a ! i l Pa " >
e ex e { r e n d p e a d f d b r i n d F ! a ( . i , . i " ) > /
e ex t r p p < /a > a >

ex B l g o p p c < k/a a >

x ro m p f < /a >
g p p < /a >

```

上記のマークアップは、フォーム上の項目を必須項目として示す必要があるかどうかを判定するために使用する数式です。項目セット内の項目は、項目セット定義または項目自体の定義によって必須にすることができます。この数式では両方を処理します。

項目セットの考慮事項

項目セットに追加された項目は、次の 2 つのカテゴリのいずれかに入れることができます。

- 項目が [項目セットで使用可能] とマークされている場合、項目は項目セット内に存在しますが、開発者はパッケージ化された Visualforce ページ上でそれを表示していません。システム管理者は、その項目を [利用可] 列から [項目セットで] 列に移動することによって、その項目セットをリリースした後、表示できます。
- 項目が [項目セットで] とマークされている場合、開発者はデフォルトで、パッケージ化された Visualforce ページ上でその項目を表示しています。システム管理者は、項目を [項目セットで] 列から削除して項目セットをリリースした後、ページから項目を削除できます。

開発者が、表示される項目をリストする順序によって、Visualforce ページ上での表示順序が決まります。

パッケージ開発者は、次のベストプラクティスに留意してください。

- インストール済みの項目セットを持つ登録者は、ページに含めなかった項目を追加できます。項目セットの反復から一部の項目を条件に応じて除外することはできないため、項目セットによって表示される項目がすべてのデータ型で使用できることを確認してください。
- 項目セットには必須ではない項目のみを追加することをお勧めします。これにより、登録者が項目セットのすべての項目を削除した場合でも、その項目セットを使用する Visualforce は正常に機能します。



メモ: 項目セットは、API バージョン 21.0 以降の Visualforce ページに使用できます。

関連リンク

[\\$ObjectType で使用できるオブジェクトスキーマ詳細](#)

グローバル変数への動的参照

Visualforce ページでは、動的バインドを使用してマークアップ内のグローバル変数を参照できます。グローバル変数を使用すると、データに関する現在のユーザ、組織、およびスキーマ詳細にアクセスできます。グローバル変数の一覧は、付録「[グローバル変数、関数、および式の演算子](#)」を参照してください。

グローバル変数の参照は、`sObjects` や Apex クラスの参照と同じで、次のような同じ基本パターンを使用します。`reference` はグローバル変数です。

`reference expression]` [

関連リンク

[グローバル変数](#)

\$Resource を使用した静的リソースへの動的参照

静的リソースへの動的参照は、テーマその他の表示設定をサポートする場合に非常に便利です。

例 c グローバル変数を使用して静的リソースを参照するには、式 e c \$ou c ! mRcs] [S R s で静的リソースの名前を指定します。たとえば、静的リソースとしてアップロードされる画像の名前を返す getCustomLogo メソッドの場合は、静的リソースを m@pu t @u u@o coag a"] R s の ように参照します。

次の例では、2 つの異なる表示テーマの切り替え方法を示します。最初に、次のコードで h T という名前のコントローラ拡張を作成します。

```
e oln el { n p c c dl h T i ss aH a
e obn en e xpmlAht T n t e ndprd Htoer m@ ra C dr l@ }rBa . S a a )
t te blt t e pt rec eAeg g libl s i a<m Sh i S T> v i a a s ( )
te @u u@e he h @t @Y@ de s v pr a @a sc a c as i a s
t e row do h@t c le e ne f/k i s h mri ilh ai s i sa
c enwnt tu{ en e r g e e mrdn ( ) S mS h > T ' - w ' ' - ; ')
} t t blt e t ptc e l@e d @{ gnp l mi<m T pi > Oi s ( )
t t e@n t o en m@de<s t p Oil n > oOic s =pl i<s s Oi > ;( )
t e tenroe r e ftg en e@e g{l bl (is m hN @ v i a a s ( ))
e te t en e@n dd t p tel n e ede coai sp .hm a( m Shm Oi ( N a, N a; ))
}
c e te n r t ru n m dn p Oi s;
}
t e obne t re e{ pdg d cm hi Ti S s
```

```

    g
    re e e tlaie { lf d cm h (T == s
    e en w w reu deTe h & h asy a a
    t t eten r t e g w tm th L &<s r i S >gL i s = L i<s i s > ;( )
    e t e tm A ddeAe g l bl L ims h. aT (v i a a s ( ))
    ee e t e e lete d t cm h nT h s ] = L i s ;[ 0
    }
    t e e e t e eru l d cm h T s ;
}

e t e eAe g l blft n em { ho i T dv ui a a s ( . ) i wa ( a ) )
ee e t e e l e d cmlhu T s v a
}
}

```

このクラスに関する注意:

- コントローラ拡張にはデフォルトのコンストラクタがないため、コンストラクタは空です。
- アップロードした静的リソース+ファイルテーマの名前を `ThemeName` メソッドに追加します。静的リソース、特に複数のファイルを含む zip アーカイブの作成とアップロードの詳細は、[静的リソースの使用](#)（ページ 135）を参照してください。
- 最後の 2 つのメソッドでは、[テーマの一覧](#)と Visualforce フォームコンポーネントで使用するように選択されたテーマが提供されます。

ここで、このコントローラ拡張を使用する Visualforce ページを作成します。

```

t n tegndprd por id AnC au acc a a " "
e tx ne e o n e evl h e To drsh $= H fa " "
x ro m p f < a >
x Be legoe pd pw crn h kAT a i =" v iP " >
st ee t l p h < as ys
e { le u e e e t eau lv edc L R lsT l )s [ c ,s 'ys s / ys ss " ")> /
e e e w hm h T & V> i < 1 >
e e n t oaut lt p c ew c no hwX t nbres h g a h s i $ i s $ < . > /
ex e tB l ga p poc < k a a & >
x t te eoueb p tele p l < e t e a Lm to vaT & h s " = L si s " > /
e ex t t dte p d e smeh ass eitse ie = LL sd s " & cm " M T a" ls " > /
e ex t nte d uproe ne en epp e arge daer S w v r m " h T a = v iP " > /
e e ex t t len{ epo cl tu p n jas dn Oi & a! Oi s " > /
e ex t t l p c < /as L i s >
ex e tB l ga p poc < c k/a a & >
ex e tB l ga p poc < k a a & >
d t l & uc o end n < dd v e e gss=ams " s y= h o d aTC h & > ! -k ss >-
e h 2 hou & <g >ihs2 s HaS -i< a >
e t n h p rde do & rdecsite pation a ptnty Lm yo i re c $ a , s
e dt e el e qn ueueq eutie iasi dl qa i ept lt ha cd us mi v i v ia a
d el loro re ntre e a iu q u q u e v ntb as drM a h boro v i a . a i
rumt d lan due d arn o ius ih sc i c i & s < . >/
e e { le u e e e t eau lv edc L R lsT o & [ g p i ' sap / . < ')> /
ex e tB l ga p poc < c k/a a & >
ex B l ga p poc < k/a a >

```

x ro m p f < /a /a >
g p p < /a /a >

このマークアップについては、次の点に留意してください。

- ページでは標準取引先コントローラを使用しますが、取引先は関係ありません。コントローラ拡張を使用するには、コントローラを指定する必要があります。
- 最初の `l g a p p o c & k` には、テーマ選択用イジェクトが含まれます。`o u pro` `pp` を使用して選択メニューを変更すると、`l go p p c` 全体が再表示されます。これは、`l p e d h e t` タグが、更新された `>cm h` を動的参照のために取得できるようにするものです。
- ここで選択したテーマ設定は、コントローラのビューステートにのみ保持されますが、これに代えてカスタム設定に保存し、永続的にすることが簡単に行えます。
- テーマごとのグラフィックとスタイルアセットが含まれる各 zip ファイルの構造と内容は一貫している必要があります。つまり、各テーマの zip ファイルには `o g g` が存在する必要があります。

このページの zip

その例を次に示します。コントローラ拡張は、システムをクエリして、ユーザがアクセスできるすべてのカスタムオブジェクトの名前を取得し、そのリストを、新規レコードを作成するためのリンクと一緒に表示します。最初に、`__c` という名前でコントローラ拡張を作成します。

```

blt      n     prc      h  g      &{ im io    dlcir a    Dss a    yi a iH s    a
t      b  e   et  tuc ob  mCj e  letcilitu{ishe mcj  QD te    t gsa> s}  OD p    i sa    ;  v    s ;
blt      &n p  oe xclart      &n t eyndpadiHtos m  ra C d    Ba . S a a
t      el d hu ob mCj      c i s .  s  o s ;( )
}

blt      t el p  u ob mCj  i vc i    s  o s ( )
t      t e et tu ob emCj t le cLbwnijsj  ts e coDt tu iobsxmCj  ls =cL i<s s  OD i  sa> ;( )
e      t e em lgbol t r b net orwutst . a gDw gsoiem t ) t slihmk dm  si i  a
t      en re M te p  in de j e< dtcaq s  p  en lgbol S r by> =c s . a E  si ;( )
t      nro er e b  jd g :  (iSk .  y s ( ))
e      &t bo {d f  h  a W( . si ( ' ) )
e      t e et te liht hdm f dn l t AG t ro u o fmca amai sa  y  s i s
e      e e et em rdb t b c j elu tbo ejt agiepo O $  g s b D  e  . ( D . ) s i ;( )
e      t e bott j f un o mC g  (D i s s  is ( ))
e      t e d t le rot u t bfmt ejv// Sta du se mcs g s ,  s s s i s
t      e et tu obemCj  le bo wjt e setOdu ob imCj  l ic sa=  s  OD i  sa ( )
e      t e tbo bjo j  gel e tbo j r Dpl IC.  a (D,) i.s  a a ( ; )
e      t ob t j dd boc j l  o s . Da( i sa; )
}

t      t b  j  ro  c  o  s s ;( )
t      e et tu ob ne j  tl  c  b  i sj . s  OD  i sa=  o  s;
}

blt el et ptu cob e mCj  tl  nd le m ssma so ip  C  i psa  s  a a
t      kln  t  e pce r  g i  is  a s ; s ;
t      kln  t  e plc del t  g i  is  a s ; s ;
e      obB nte{do e p  te  bi  c  g i  a  aa ; s ;

t      e blt et tu op dmCj  e  crt  i  gn s mr @D  ph nsatS  Nr a abliSLC  a a ,  i a
t      e  t  e  rm  m  i s .  a s N a;a
t      e  t  el b l hr  bl  i s .  a  IS  a a
}

e      nbl e  er pg d r m ob e { jpb  oTo r m  C T ap q  a )
t      e et tu ob mCj  lt edo et tuc sb mCj el i oba oj e r rs  CODT p  i sa  )  ; a
t      t ner  r  e  hm eo r m odo  p  rma T  a s.  a ( .  a s; )
}
}

```

この拡張では、次の点に留意してください。

`__c` メソッドは Apex スキーマメソッドを使用して、使用可能なカスタムオブジェクトに関するメタデータ情報を取得します。`__c` メソッドは、使用可能なオブジェクトとカスタム設定に関する小さいメタデータセットを取得するための軽量の操作です。メソッドは、コレクションをスキャンして名前の末尾が「`_c`」(カスタムオブジェクトまたは設定であることを示す)の項目を探します。

これらの項目は、`__c` を使用してより詳細に調査され、選択したメタデータがカスタムオブジェクト用に保存されます。

`__c` を使用して、項目がカスタムオブジェクトであるかどうかをテストするのは、「ハッキング」のように感じられるかもしれません、もう 1 つの選択肢である

コードにはコストがかかり、`__c` へのコール数にガバナ制限がある

あります。大量のオブジェクトが含まれている可能性があるリストに対しては、「`_c`」文字列のスキャンを最初に行うのが効率的です。

- このメタデータは、内部クラス `m Cj` に保存されます。このクラスは、保存すべき項目の単純な構造化されたコンテナとして機能します。
 - `m Cj` には `Comparable` インターフェースが実装されており、各オブジェクトの属性（この場合はカスタムオブジェクト名）でカスタムオブジェクト詳細のリストを並び替えることができます。

ここで、次のマークアップで Visualforce ページを作成します。

t n tegnd prd por id An C au acc a a " "
e tx n n n & an m o dlc r \$ Ds = yi a iH s a " >
br < > /

x t e ed{ bl ple uet Ta ob maj}e á áva r a" ol sl OD i sa'v # a i sá >
x n lo u mx < a >
e ex te ene p tñed r t chu o ex ant j a = a " a p f s o < /a a >
e xt et txe ñu ue p tl up < t l ác b l r v a" D ! i sa . a s > /
x n lo u mx < /a >
x n lo u mx < a >
e ex te ene p fn de Án ex t ac a = a p f 'a > i < s /a a >
e xt t enó u pl up t < Antka éo ím el d L hr F Q (i] D [i sa . a S N ")
e ee {n er dtr de tó l e e } tble c rx e D t Ci nsu u pba > p [< a ka L i < >
e xt t enó u pl up t < Antka éo ím el t d L hr F Q (i] D [i sa . a Si s ,
e te e b t j e mce l q p rm O jt e] y * ti ffa . a o] S . u y pi P L' p i s[ka L i
x t e d bl p T /a a a a >
q p p < /a a

特定のレコードが割り当てられていないページで使用できる便利なアクションは、`と` の2つのみです。レコードをクエリするページで、`e c we` グローバル変数は、`ot d`、`C l`、`VE`、`i` などのメソッドを提供します。特定の標準オブジェクトには、データ型に適した追加のアクションがあります。

関連リンク

\$Action

\$Action グローバル変数の有効な値

`$ObjectType` を使用したスキーマ詳細への動的参照

c \$ グローバル変数を使用すると、組織のオブジェクトに関するさまざまなスキーマ情報にアクセスできます。たとえば、オブジェクトの項目の名前、表示ラベル、データ型の参照に使用します。

c) \$ は「深い」グローバル変数であり、次のような「二重に動的」な参照に使用することができます。

e b eɪ e b c eɪs dle m fde e Ol ld f v smd[N] ai s s i [N a . v

次の例では、動的グローバル変数を使用して一般的なオブジェクトビューアを提供します。最初に、
j dd r c D という名前で新しいコントローラ(拡張ではない)を作成します。

blu em t p e c b m j d k r ic Dss a yi aOH a
 t l hte oT ekr ld r Å/ m set a fawhism a ja tc m n co om c yi VpO i

```

t r ne r p bog j bl ˇ pt a i s > y ss ˘ s;
t rt en er e g bl ˇ cdn á<s is > y ss ˘ s;
e tbli b ˇ bo jc i s o ;
g ; s;
te te eb j l cu pt s o w ( a; )
e e de aor ebl cdd l ui sv ss ˘ ( a; )
e ebo trj l deb At ll cld h = w Oa i ˘ D a;(a )
}

e tet b ˇ T nc r p g/ s o y s as a
t e blte rte [p cg bg j c i pt s o y ( )
t n r eru bo ˇ pt ˇ s . y; )
}

t e blte rte p cg b ej fnb c i j pt s cs o y s(0
e e boen ˇ te ke pt ge t ij es . e cyte r hto . c g s m y ( ) s i (.) N a; ( )
t n r eru bo ˇ pt ˇ s . y; )

c e e eo klt f add l o if/s b ˇ ss ci s s o
t tbli e nte p ce a g g{ bl l add i s > ss ˘ s ( )
t n r r ue e hbl cdd ˇ s . ss ˘ s ; )

r e e d p aor ebl cdd v wab i jsv b c js ss ˘ s s(0
e e e e hbl wadd t n i sr. ss g ˘ s = L i<s i s > ;( )
t en re Metg p e n d h j ld< cads , fs . a s ˘ >i s =
t en re ef gt{ ld : f (is ls i s . y s ( ))
e e fe mt e dd ie gfs( legNr he A )&( sbl sd( ) s i ˇ s) ss ( )
t e e e hbl cdd dd i s . ss ˘ s . a; )

e ret ˇ p j r t de bc At{v ll cl d h w Oa i ˘ D a(a )
n e r ex d eq te negrntpe e id gie st rm Ba . gd Pa(.) Pa a s (.) i ( ; )
et en r g n h etn te E ESQ A T g y l b l s dd Li is ( ss ˘ s t ( , ) , ' )
e te te bgM j ct FR QP ' o + y ( )
d Ed E q w H R = i ; ' tne r eru e the n q n hd ( asa a Q y( y ) )

en n xoA e to n pld e fof ˇ tildE EG i fr si ˇ as i s s i
t tr t n p tn t c g j etn ts t ag i sn L th il s i s rd s x pi s , i s s a a )

e t t { n e f h l u n n ri } lu( i s == ) a ; ;
t { r ne rof tip ri ros( pa a== ) s a a = ; ' ' ;

en rn g jd i s i = ; ' '
B net tao et r fm ˘ ia s = ; ;
t e tmo et { ft g m : h (i s i L i s )
n e t{ lu f m i ( ˘ i )
e t r et t f m I fili I i ( s f i ) s = s a a
e e { e l en o jd m ro sp+ i = i ; ;
e n t o jd m + i = i ; ;
e n r nru o jd i ; ;

```

このコントローラでは、次の点に留意してください。

- Visualforce コンポーネントはコントローラ拡張を使用できません。代わりに、このクラスはコントローラとして記述されます。コンストラクタは定義されないため、このクラスではデフォルトのコンストラクタが使用されます。
- オブジェクトのメタデータを収集するには、コントローラでオブジェクトを把握している必要があります。Visualforce コンストラクタは引数を取れないため、インスタンス化時に目的のオブジェクトを知る方法がありません。代わりに、公開プロパティ `Object` を設定することで、メタデータ検出がトリガれます。
- このクラスでは複数のメソッドが、前の例より若干異なる方法でシステムスキーマ検出メソッドを使用しています。

次の例は、オブジェクトに関するスキーマ情報と、クエリされるレコードの特定の値を表示する Visualforce コンポーネントです。次のコードを使用して、`Account` という名前で新規 Visualforce コンポーネントを作成します。

```

t n n t nom p or lpr ct e &om j d&r c D = yi aOH a" >
x t t e en b utp em te e ba d&arp n &ar "u y" s O " i = " "
t ed nre totec bo pe jh & csli{ &n = p g) bo ji s T ya". ss a = ! " > /
x ro m p f < a >
x ex te lgo p tel p c bo<} j k a ca &T = ! y" >
e ex tB legn p poc l & lkd a loau m & i = & s" 1s=" " >
e ex te ed{ &le ple ue T< bl &acd a av a & ! &sf & s"v = a" >
e ex x n lo u m < a >
e ex te ene e p &ed r ex &t a a = &p f L'a > < a /a a>
e ex tet txe &u u e p lte up e &T te j ae bo c j &e p &d ! &pf y ]f [ y i sL [
e ex x n lo u m < /a >
e ex x n lo u m < a >
e ex te ene p &nd a &ex t am I a &a p &a > NR a /a a>
e ex tet txe &u u e p lte up e &T te j ae bo c j &e p &d ! &pf y ]f [ y i s N
e ex x n lo u m < /a >
e ex x n lo u m < a >
e ex te ene e p &nd r ex &t a a &T &f &"a > < y /a a>
e ex tet txe &u u e p lte up e &T te j ae bo c j &e p &d ! &pf y ]f [ &y i s
e ex x n lo u m < /a >
e ex te ene e p &nd r ex &t u a a = &p v &"a > < a /a a>
e ex tet txe &u u p l up &T j a ]f v &a" ! & & > /
e ex x n lo u m < /a >
e ex te d bl p & /a a a a>
ex e tB lga p poc <c k/a a & >
ex e tB lga p poc <c k/a a & >
e ex x tit &Bndm p no clo & mka a a & s=" " &
e ex tit &Bndm p o w l u < a a v v &i " "
e ex tit &Bndm p o t ce An te & bo &ji a U Lc! R &B p &ji & v y i , . " )> /
e ex tit &Bndm p o t ce An te & bo &ci a U &Lc! R &B p &ji & v y i , . " )> /
e ex tit &Bndm p o ln u & o a C a v &i "
e ex tit &Bndm p o t ce An te & bo &ci a U &Lc! R &B p &ji [I y . , . " )> /
e ex tit &Bndm p o t ce An te & bo &ci a U &Lc! R &B p &ji [I Dy . , . " )> /
ex ex e tB lga p poc <c k/a a & >
ex ex ro m p f < /a >
t n n o om p < /a >

```

次の点を確認してください。

- このコンポーネントを使用するページでは、レコードを検索する必要があります。これを行うには、そのオブジェクトに標準コントローラを使用し、URL でレコードの `ID` を指定します。たとえば、

`http://na1.salesforce.com/Account?id=001300000G`

- 選択したレコードはすぐにコンポーネントの属性に渡されます。このパラメータは、すべてのオブジェクトメタデータ検出に使用されます。
- この3つの二重に動的な参照が、`e bo c j $ object_id` で開始し、各項目のメタデータを表示するのに対し、通常の動的参照は項目の実際の値を表示します。
- データ値の場合、値は、より自然な `c]f` (コンポーネントへのパラメータ)ではなく `]f` で、コントローラ内でgetterメソッドが使用されます。その理由は単純で、属性はすべての項目のデータを読み込むように更新されていますが、`c` は標準コントローラで読み込まれたときと同じ状態のままであり、読み込まれるのが `I` 項目のみであるためです。

最後に、新しいコンポーネントを使用して、任意の数の単純な Visualforce ページを作成できます。このページは、コンポーネントを使用して、次の2つのページのようなレコード詳細およびスキーマ情報ページを表示します。

```
t n tegndprd por id knC au acc a a " " >
t e ew bñ: j cr rta < D } o cyiVaO ci =! a " > /
g p p < /a >
```

```
t n tegndprd por id k C o a acc ca a " " a>
t e ew bñ: j cr rta < D } o cyiVaO ci =! ! "a > /
g p p < /a >
```

関連リンク

[\\$ObjectType](#)

[\\$ObjectType で使用できる項目スキーマ詳細](#)

[\\$ObjectType で使用できるオブジェクトスキーマ詳細](#)

第 13 章

動的 Visualforce コンポーネント

Visualforce の基本的な目的は、静的なマークアップベースの言語として、開発者が Salesforce のデザインとマッチしたユーザインターフェースを作成できるようにすることです。ただし、場合によってはプログラムでページを作成することが必要になります。通常、これは標準マークアップでは難しいか不可能な、複雑なユーザインターフェース動作を実現する場合です。

動的 Visualforce コンポーネントは、ユーザの権限やアクション、ユーザまたは組織の設定、表示されているデータなどのさまざまな状態に応じて、コンポーネントツリーのコンテンツや配置が異なる Visualforce ページを作成する方法を提供します。動的 Visualforce コンポーネントは、標準マークアップを使用する代わりに、Apex で設計されています。

動的 Visualforce コンポーネントは、次のように Apex で定義されています。

```
Component_namespace Component_name
```

たとえば、`<apex:component>` になります。

 メモ: 標準コンポーネントの参照には、すべての有効な Visualforce コンポーネントの動的な表現が含まれます。

Apex で動的に表現される Visualforce コンポーネントは通常のクラスとして動作します。標準 Visualforce コンポーネントに存在する各属性は、get メソッドや set メソッドによる対応する Apex 表現でプロパティとして利用できます。たとえば、`<apex:component>` 属性を次のように操作できるとします。

次のような状況では、動的 Visualforce コンポーネントの使用を検討してください。

- 複雑な制御ロジック内で動的 Visualforce コンポーネントを使用し、同等の標準 Visualforce では作成が難しいか不可能なコンポーネントの組み合わせを作ることができます。たとえば、標準 Visualforce コンポーネントでは、通常、コンポーネントの表示は `rendered` 属性とグローバル `isTrue` 式関数を使用して制御します。Apex で制御ロジックを記述することで、より自然なメカニズムを使用して動的にコンポーネントを表示することができます。
- 特定の項目を持つオブジェクトを反復処理することがわかっているが、具体的にどのオブジェクトかわからない場合、動的 Visualforce コンポーネントでは、汎用的な `sObject` 参照を使用してオブジェクトの表示を「埋め込む」できます。詳細は、「関連リストの使用例」(ページ 174)を参照してください。



警告: 動的 Visualforce コンポーネントは、組織の新規 Visualforce ページを作成するための主要な手段にすることを目的としたものではありません。既存の Visualforce ページを動的な手法で記述し直さないでください。

ださい。標準 Visualforce コンポーネントは、ほとんどの使用事例に適しており、推奨されています。動的 Visualforce コンポーネントを使用するのは、ユーザの状態またはアクションに応じてページを自動的に変化させる必要があり、静的なマークアップにコード化するのが難しい場合のみに限定してください。

動的コンポーネントの制限

Visualforce の機能には動的な状況では意味のないものがあるため、コンポーネントの中には動的に使用できないものがあります。このセクションでは、動的 Visualforce コンポーネントの制限について説明します。

- 次の標準 Visualforce コンポーネントに対する動的な表現は、Apex には含まれません。

x	t	e	t	Ø	r	b	u	p	<	a	ia	>			
x	e	t	n	n	Ø	o	m	Ø	p	<	a		>		
x	e	t	n	n	Ø	B	m	Ø	p	<	a		y		
x	t	Ø	a	o	m	Ø	p	p	<	a	Ø	i	>		
x	e	n	d	Ø		p	f		<	a	i	>			
x	n	e	d	Ø	t	n	n	Ø	p	<	a	y	i	a	>
x	e	n		Ø	l	du	p	c		á			>		
x	tn		Ø	r		p			<	á	s		>		
x			Ø		Ø	mp	p		<	a	a>a				
x	e		Ø		Ø	bl	p		<	va	i	a	a>		

- 動的 Visualforce コンポーネントが特定の sObject 項目を参照し、その項目が後で削除された場合、その項目参照用の Apex コードはコンパイルされますがページは表示時にエラーになります。また、`b1` や `p$` などのグローバル変数への参照を作成することはできますが、参照される項目を削除すると同様に失敗します。このようなページが引き続き予期したとおりに動作することを確認してください。
 - 動的な Visualforce ページと式では、静的ページよりも厳格に属性型がチェックされます。
 - 動的コンポーネントには、「**パススルー**」HTML 属性を設定できません。

動的コンポーネントの作成と表示



メモ: このセクションの例は、説明のために意図的に簡略化されています。動的 Visualforce コンポーネントを活用した例の詳細は、「[関連リストの使用例](#)」(ページ 174)を参照してください。

ページ上の 2 つの部分に動的 Visualforce コンポーネントを埋め込むことができます。

- e 1. ページの任意の場所に `n n map om c C < p` タグを追加する。このタグは、動的コンポーネントのプレースホルダとして機能します。
 - 2. コントローラまたはコントローラ拡張内の動的 Visualforce コンポーネントを開発する。

`n n map om eC <ap n タグには da u p >` という必須属性が1つあります。この属性は、動的コンポーネントを返す Apex メソッドの名前を受け入れます。たとえば、送信フォームの期限が過ぎたら、異なるセクションヘッダーのタイトルを動的に生成したい場合、次のマークアップとコントローラコードを使用します。

n t e g n d p r d p o r d h e t C x o a n n a c n a o e a t n e k' m o o m " b a C l i p E s - " y i a a " > /
n e d t n n e m p d r e n p C o e < o e p d a t e y i e a t e r h u V } h a " W b a D k a a " > /
r o m p f < a >
x e t n e { u l d p t l t n w t e < o a) C E m v a ! L a s N a " a > /

```

x x tit eBndmme w o lt u < en a o } a v a" v Sa ia =! ts "a > /
x ro m p f < /a >
g p p < /a a

blu h e p t n exmo om c C i np Dss a yi a a
ble t n exmo ome<C A tip 底 n ty na lpr d( or md r aC) d Ba . S a a
e bten ne ox dpe c re pteo phide egder (. u. h h h c dk a ( )
e et d e de u wntel n n Da c 2 a s 1a ( 0,7 , ;4 )
e ab do e t o d u d e esdo wedB en t v a d u= a y(. )sy a D ( < a ); 0
e ten ne ox dpe eC re p te ep n a e o wnde rc te n 底 ox dpe aC 底 p o a p rc .
e { ord u f i v ( )
t e e n te t o d rc l e rs h m e Tt ai dt a iW srD m st a o D ' + a a ( )
t e iter erun o d rc s 底
} 1 s
t e e n te t o d rc l ro h m o ai = F ' i ss ; '
t e iter erun o d rc s 底
}

```

1 ページに複数のコンポーネントを含めることができます。

各動的コンポーネントには、一連の共通メソッドおよびプロパティへのアクセス権があります。この一覧を確認するには、『Apex 開発者ガイド』の「コンポーネントクラス」の章を参照してください。

動的カスタムコンポーネント

カスタムコンポーネントを動的に使用すると、標準 Visualforce コンポーネントとまったく同じ動作をします。名前空間がカスタムコンポーネントの名前空間に変わるだけです。カスタムコンポーネントは、名前空間にあるため、次のように動的に作成することができます。

独自のコンポーネントの利便性のために、次のように名前空間を除外できます。

サードパーティがパッケージで提供するコンポーネントを使用する場合は、パッケージプロバイダの名前空間を使用します。

コンストラクタを使用して属性を渡す

プロパティを使用してコンポーネントの属性を設定する代わりに、コンストラクタを使用して1つ以上の属性のリストを渡すことができます。

属性がコンストラクタに定義されていない場合、コンポーネントのデフォルト値がその属性に使用されます。

プロパティを介してではなく、コンストラクタに属性を定義する必要があるコンポーネントが2つあります。

- レコードの Chatter 情報とコントロールを表示する場合、Component で、`rhd` がそのコンストラクタに渡されている必要があります。それ以外の場合、この属性は常に false になります。

- ユーザが一度に複数のオプションを選択できるようにする場合は、`multiple="true"` で、`list="true"` になります。

これらの値は、`string` ではなく `boolean` です。これらを单一引用符で囲む必要はありません。



警告: 属性名が Apex キーワードと一致する場合は、クラスコンストラクタを使用して属性を渡すことはできません。たとえば、`List` は予約キーワードであるため、`list="true"` はコンストラクタを使用して `list` を渡すことはできません。同様に、`for` もキーワードであるため、`for="true"` は `for` 属性をコンストラクタ内に定義できません。

式および任意の HTML の定義

- 式の言語ステートメントを `expr` プロパティを使用して追加できます。式のステートメントを渡すには、プロパティ名の前に `expr` を付加します。静的マーカアップと同様に、式は `apex:repeat` 構文でラップする必要があります。次に例を示します。

```
<apex:repeat id="example" value="0" to="10">
    <apex:outputText value="Hello, world!"/>
    <apex:outputText value="The value is: {0}"/>

```

有効な式には、標準オブジェクトやカスタムオブジェクトの項目を参照する式などがあります。次の例のようにグローバル変数と関数も使用できます。

```
<apex:repeat id="example" value="0" to="10">
    <apex:outputText value="Hello, world!"/>
    <apex:outputText value="The value is: {0}"/>

```

式で値を渡すのは、それをサポートする属性でのみ有効です。プロパティ以外で `value` を使用すると、式ではなく文字として解釈されます。

プレーン HTML を含めるには、`expr` タグの `value` プロパティを `fn` に設定します。

```
<apex:repeat id="example" value="0" to="10">
    <apex:outputText value="Hello, world!"/>
    <apex:outputText value="The value is: {0}"/>

```

facet の定義

式を定義する方法と同様に、`facet` は動的コンポーネントが使用できる特殊なプロパティとして機能します。次に例を示します。

```
<apex:repeat id="example" value="0" to="10">
    <apex:outputText value="Hello, world!"/>
    <apex:outputText value="The value is: {0}"/>

```

`facet` についての詳細は、「[コンポーネント facet の使用のためのベストプラクティス](#)」(ページ 282)を参照してください。

子ノードの定義

- プロパティを使用して、子ノードを動的 Visualforce コンポーネントに追加できます。プロパティは `children` で、オブジェクトのリストへの参照として機能します。

e 次の例では、`dm` `C` `p` を使用して子入力ノードで `m p f` を構築する方法を示します。

```

blen n ex oAp c eC tB p l go p i g c mo . km c . Pa D yi a ( )
ten n ex oAn C B p l go p B w ego te.nk n ex Bm k yB p Pa go p c . k . Pa ;( )

t mten C u ld eAmo p f o/ f acima i N a
ten n ox oAn Cet ep au ld p le p w n d n ten n ox oAn C N p fai ld pI a p
e ex ind enr { o l p t eAN fa o/ . ssim vs . a= ! ' N a; '
et end d mh N fa i . = N a
ten n ox toante C et ju ue b l p e p w h m h . ten n oL oL toante aC N pu au ba p p
e e e b mle l nu t An m Nu a cva . a= R ' a ; ? '
e et b mle ro f mh N a a . = N a

t mten C u ld a m o p f o/ f bacima i N
ten n ox oAn Cet p u tdeAn e p oue h b comld ten n ox oAn CN p f u ld pI p
teAn e oueh b rcm ld enr { o l p t NAn t eAn . ssim } b rcm a= ! ' . N ;
teAn e ouh b rcm ld d t eAn ou h b rcm f i . = ' N ; '
ten n ox toante C et p u u b t eAn e p oue h b comld ten n a c toante CN L pu u ba p p
teAn e oueh b rcm b le l m t An gN kou cva . a= ' a # ; ? '
teAn e ouh b rcm b ro t eAn f ou h b rcm a . = ' N ; '

t ttr t n b oto o bu m h m af s i
ten n ox oAn Cet epBndmmtypCo Bn wne o . ten n ox oAn Cts pBndmmtypCo
ct eBn ue o l u w a v . a= v S a '
ct eBn u o ntr { o e p o s } ca . ss s . i a = tv ' S a; '

e eAt bl em tra n hm omf c /s p s
n d B lego t hdn e c em e Gd y p Pa bmhi s . a( N a a; )
n d B lego t hdn e c em e Gd y p Pa indi s . a( N fa ; )
n d B lego t hdn e c dm GdeA ypePa ou h b rcm b l s . a( N L a; )
n d B lego t hdn e c dm GdeA ypePa ou h b rcm ld s . a( N 直 ; )
n d B lego t hdn e c dm tt Gd Byp Pau o . i s . w( a ; )

en r n ru d B l go c k y Pa ;

```

マークアップが次のように定義されている場合:

```

x ro m p f < a >
x n e d t n ne m o d m e p Co < o p d au yi a } mo Vm c a" ! yi a " > /
ro m p f < /a >

```

このマークアップは次の静的マークアップに相当します。

```

x ro m p f < a >
x B l go p p c < k a a >
x t te o e tu d p ro p < f mha L a = N " a > /
x et n e { u ldp lt eAn < o a j t a i cc m v a = N " ai = N " a > /
x t te o e tu b p toeAn < fu h b rcm a = N " > /
x et n e { u ldp lt eAn t eAm a i coe h b rcm d eAn ! ou h b rcm " i = N " "
x t t Bndmme p o lt u < en a o } a v a" v S a i a = N " a > /
ro m p f < /a >

```

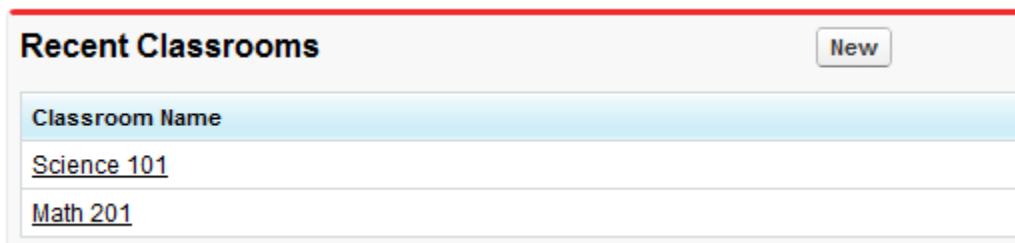
同等の静的マークアップでの要素の順序は、動的コンポーネントが `mo` `m c` メソッドの Apex コードで宣言された順序ではなく、`dm` `C` `p` に追加された順序です。

関連リストの使用例

動的 Visualforce コンポーネントの使用は、参照するオブジェクトの種別がわからない場合に最適です。一方、動的 Visualforce バインドの使用は、アクセスする項目がわからない場合に最適です。

動的 Visualforce を使用するための次のシナリオでは、アクセスする必要がある既知の項目セットを使用して単純で再利用可能なページを構築します。ページとそのカスタムオブジェクトは、未管理パッケージに配置され、同じ組織全体に配布されます。

最初に、Classroom という名前のカスタムオブジェクトを作成します。次の図のように、2つのオブジェクトを作成し、それぞれに Classroom および Math という名前を付けます。



次に、さらに2つのカスタムオブジェクトを作成して、それぞれに Student および Teacher という名前を付けます。各オブジェクトの作成が完了したら、次の手順を実行します。

- [カスタム項目 & リレーション] の下にある [新規] をクリックします。
- 主従関係] を選択し、[次へ] をクリックします。
- ドロップダウンリストで [教室] を選択し、[次へ] をクリックします。
- 続いて [次へ] をクリックし、すべてのデフォルト値を変更せずに残します。

次のオブジェクトと照合リレーションを作成します。

「新しい名前の新規 Student」と「新しい名前の新規 Teacher」の両方を割り当てます。

別の「新しい名前の新規 Student」と「新しい名前の新規 Teacher」の両方を割り当てます。

次に、Apex ページを作成し、次のコードを貼り付けます。

```

    public class RecentClassrooms {
        @AuraEnabled
        public List<Classroom> getRecentClassrooms() {
            return [
                SELECT Name, Description, StartDate, EndDate, Status, IsArchived
                FROM Classroom
                ORDER BY StartDate DESC
                LIMIT 5
            ];
        }
    }
}

```

e ent e u n g l i h a P a ;
 e t e d ed w n t t m n r s g N sa = < S i S > ;()
 e e { t d f I i i (s S)
 e ex A e t e negrntpe e t C g g t rm Ba . u d b a p . j Pa la s (.) i (' , ' ;)
 e e t e d l I f i s S = s a
 }
 e e e e b l e n { g r c of c h i Ra s ()
 e e e g d g l r o o s C ma Pa V Pa y F ss a ;
 e e t e e t g g t r m u d b o Pa . j Pa la s (.) i (' , ' ;)
 e e n r ru g
 }
 e t e b l u t t l p a o fct l p i s p s Oi > i s L o a s i s
 e t t t l m e a con stoep t < s p Sl n Oi > o ic s = p i < s s Oi > ;()
 t l r o o C m l r d o c i a n n s s d a E E C m s s T a s = D M C [S i , c] R O s s a ;
 t r o l r d o g m l r c o o c m (s s a s s a s)
 t n e w n o d d t p l n e o n c d i s p . c a n c s Oi i . , . a ;))
 t t n r r m o o p i s ;
 }
 e e e e b l e n t g r c z u f o c m i Ra s i ()
 e e ex A e g t e n g r n p s C g a Pa Ba . Pa ;()
 e e t e e t g g t r m u d b o Pa . j Pa la s (.) i (' , ' ;)
 e e n r ru n d g t l m r o d c m Pa y i a s s a i s ;
 }
 t e e w n o h b l B n t r n t r o e t @ / u f t c s g c l h a c s i s i s
 e t e b l u t t e d p a e t & { cl n d i < s c s p i > s Oi s
 t t e n t r e w o d t e e n e i t s e i s g > s d = L m i < s i s > s (N s a ;)
 e t t r o d r o s s s ;()
 e t t l m e a con stoep t < s p Sl n Oi > o ic s = p i < s s Oi > ;()
 t n o e t f f g r o d : (i s s s)
 t n e w n o d d t p l n o c i s p . a (s Oi s (s ;))
 t t n r r m o o p i s ;
 }
 e t e b l u t t d p a e n t o t c u l p d i < s o c s O i p > s Oi s
 t e e e e e m r a b b t b c j l u c D r a a e C n e o s s R e g r b s p c _ o s y D . s i
 t e e t m l d m l e c o l e i < s h t S g l d m l R k i o a s i > h R . p i R i a s i s ;()
 e t t l m e a con stoep t < s p Sl n Oi > o ic s = p i < s s Oi > ;()
 t e n e t f m l d m l { k o r h (S c p a i C R i a s i)
 t e n e r e r d g t m r e g c o i s N h a p . R i a s i N a ; ()
 e e e n e r e n e n t d r n e d W t u o / l m c ' o i y s h i p s i a s i s
 e e r l n e n l l u t r l n i m { m o (N c a t & N a i s a (')
 t n e w n o d d t e p l e o e d i s p m l a (m S Oi (N a , N a ;))
 t t n r r m o o p i s ;
 }
 e t d d p d c i v i s ()
 t e n e r e r t f q l d : c i (i s s s)
 }

```

e e t e l d id s N sa . ss(;) 
e e n t e aule d rm o m s N sa . v ss(;) 
} 

t ble nt d p c l c i v iU (iS s s s(;) 
e e n t e u ul d id c S N sa . ss(;) 
e e t e ele d rm o m s N sa . v ss(;) 
} 

e bten n ox toate Ce ntu u ep i e top r tfo lcmd o Pa ss a R L ass . o 
e ten n ox toate C ntu tan le dn yue l ten n ox toate Pa ny u Pap p . o 
e en e er t edf g l d : m (iS i s N sa ) 
e ten ne oxeom C t elp dn tp d ln e ten ne oxeom a st Lp ids =p . R L ass 
e nt dt l l d LRY is is =i ; 
e ten dn ue l t mdn oe dm n did d y o l Pa .i s . a (LRY is ; ) 
} 

n ten ru dn u l yO Pa 
}

```

保存しようとすると、Visualforce ページが存在しないというメッセージが表示される場合があります。リンクをクリックしてページを作成すると、その後にあるコードブロックが入力されます。

次に、`r oo Cm` という名前で Visualforce ページを作成し、次のコードを貼り付けます。

```

t n n tegndprd por ld s el ae ootCmr arda c trc # lss ac " _ v s =" a ssias " 
n n no tlm r ooc Cm $ Ds=" yi a ssLa is " > 
x n e d tn ne map dm qCo <op d au epita r too lcmd & ! ss a R L ass " > / 
x ro m p f < a > 
e ex teB l go p lp d e kCen l blz do m d' ss a sv i a 'a = " i " > 
e e ex e t el{ d po lcu } ko d j as IR i va & ! " > 
e e ex t t len{ po cl u pd tl ab} joip w & lis Lgasi s " > / 
ex B l go p p c < /as R i a> 
x tit ro m p f < /a > 
g p p < /a > 

```

最後に、`r ooc Cm D` という名前のページを作成します。このチュートリアルを最初から実行している場合は、コントローラ拡張を構築したときにこのページをすでに作成しているはずです。次のコードを貼り付けます。

```

t n n tegndprd por ld s el ae ootCmr arda c er cl=" ss arg r " _ M sv = " is Pa a" 
n n no tlm r ooc Cm $ Ds=" yi a ssLa is " > 
x m qbr < a ss sa > / > / 
x ro m p f < a > 
e ex teB e d go ep lptcl < n ikao a edeh l=t ps Rl piB asi eDc i s c " ja = " sk i 
e ex n l r d plo pu mc< a aiG s=" 3" > 
e e ex t t lee enpd c tel < ed l aesiL in si # s f i s " i = " s = " 0 
e { e e e i u d e dt elucm el v z a'u ls " i & = " " $ = " 0 
e ex wt l xd h:2 ps y= i '50> 
e e ex t t len{ poe cantu pt ul asd } oac w p& ! s ois " > / 
e e ex t t l r op p < p a a G > 
e ex n l r op p < p a a G > 
e x tm eBndmm pc o tl{ u < e ato cja v & c> i a & ! s " 

```

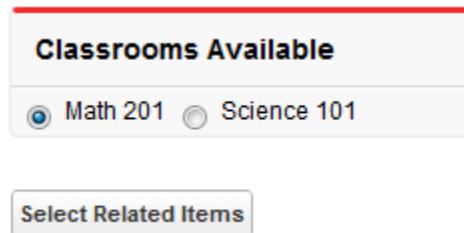
e e e e n e r t d r n l B ol o c R =' s k i " > /
br < > /

e x tit @Bndmm w o tl{ u <n e e a n t coa} l v <@" " b ia @'U ! s "
e e e e n e r t d r n l B ol o c R =' s k i " > /

e ex n l r o p p < p /a a G >
e ex t t ele e t pd del ed l e a s L n d s i l =' s f i s " i =" s " a
e { e e dntu u @e d t @l u c m e t @" u e " i g " =" " \$ =" " 0
et wt l x d h : 2 ps y " i " 50>
e e ex t t len{ e p o e c t u t k l n a s } o o i s p a " l s o i s " > /
e e ex t t l p c < /a s L i s >
ex n l r d p p < /a a i G >
br B l go p p c < k/a a >
x tit @Bndmm w e o w d tu < @ l a t @ a v w o @' c @ R L h a i s s' i a = l s " > /
ro m p f < /a >
q p p < /a a

これは、表示するオブジェクトリレーションを選択するオプションをユーザに表示するページです。「selected」リストと「unselected」リストへの入力は、動的な方法で行われます。

コントローラ拡張とこれらのページを作成したら、組織の 1 IP OO CM に移動します。次のようなシーケンスが表示されます。



Select Relationships to Display

Students_r
Teacher_r

Students_r
Teacher_r

Show Related Lists

Students

Action	Student Name
Edit Del	<u>Boont Amber</u>

Classrooms Available

Math 201 Science 101

Select Related Items

第14章

Visualforce とメールの統合

Visualforce を使用して、取引先責任者、リード、またはその他の受信者にメールを送信できます。Visualforce の機能を利用して Salesforce レコードを反復処理する、再利用可能なメールテンプレートを作成することもできます。方法は、次のトピックの説明を参照してください。

- Visualforce を使用したメールの送信
- Visualforce メールテンプレート

Visualforce を使用したメールの送信

メッセージを配信するカスタムコントローラを作成すると、Visualforce を使用してメールを送信できます。Apex

Visualforce でメールを送信するときに使用可能で多くの機能については、次のトピックで説明しています。

- メッセージクラスを使用したカスタムコントローラの作成
- メール添付ファイルの作成

メッセージクラスを使用したカスタムコントローラの作成

名前空間を使用するカスタムコントローラには、少なくともメールの件名、本文、および受信者が必要です。また、件名と本文を記入し、メールを配信するためのフォームとして機能するページが必要です。

という新規ページを作成し、次のコードを使用します。

```
ten e gorpl&pc d < l m a E a =s i "a>
m & < a ss sa > /
teB e l go p nlp n ctd < k lam o a u i =s S ai a Y
te nene & t ecmr p a . aR s v as >
te tell eou p wld h l &fw o & et o &unh m gtd e h l m o ury i s apia s .
br < > /
ce ed{ bl plt unt Ts ou bcc t atave c a ! b ard r c sa"v =a "a 1 = " >
n lo u mp < a >
x te ene e p end rex ht m a a a f "a & a /a a>
tne o } c mc ! a a
x n lo u mp < /a >
n lo u mp < a >
x te ene p fed r ex ht m aE a a f "a > a /a a>
tn o } cl mc E ! a i a
x n lo u mp < /a >
t e d bl p & /a a a a>
ro m p f < a >
```

```

t te br br < < > / > / 
et n t x e { eou u e b p tl w kai jt a o L bu ja br a c S: " " " " < > / a > / 
br br < < > / > / 
t te eou u b p l B p d s Ro a d f brva a" " y " " > / > / 
et n t x e u p l p u k b do a d w do r v a a" l o " y i c " " y 1 s " " 0 s " " 8 & / 
br br < < > / < > / > / 
t t t B l go p p c < k/a a > 
g p p < / a a 

```

ページマークアップでは、取引先 ID はページの URL から取得されます。この例が正しく機能するためには、Visualforce ページを URL の有効な取引先レコードに関連付ける必要がありますよたとえば、
が取引先 ID の場合、次の URL を使用します。

```
h e : pSalesforce_instance / # t /as / i I a EalB 00 000R003 5
```

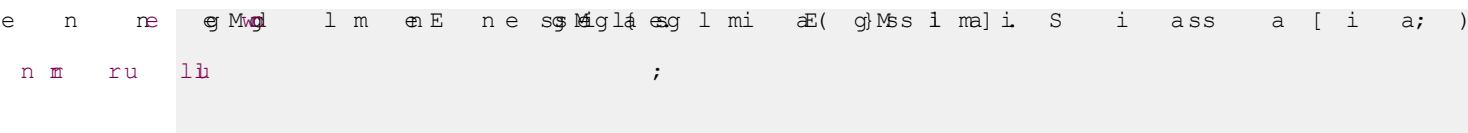
レコードの ID の取得についての詳細は、[Visualforce での項目値の表示](#) (ページ 20)を参照してください。

次のコードは、`Email` クラスを実装する `EmailController` というコントローラを作成して、取引先に関連する取引先責任者を受信者として使用します。

```

e b h { h p d c l m E i ss s i a 
k h n { t r e t j g c i i s s ; s ; 
b h n e r t e p d g g i i s y ; s ; 
r n t A h p t o m f cou v cc a a a ; 
t nt t @ r u e r d e t c o l u n / t e A a p t a s o u h b o c c j a c s a 
h d plcm E i s i a ( ) 
e n e t o a u c d t m n e a E E g s C [ t c n , C l a n t E t n o M a g c Q i c R O 
t e n a o m o u f r e c d A e h t e n e g r n t p e e t g e = g t r m B a . g d P a ( . ) ] P a a s ( . ) i ( ' ; 
t b l A n t t u p An c g ou cc i ( ) 
n r t r m ou cc a ; 
e e b l e e n { g r n d f c i Ra s ( ) 
e t e n f l m h D / i i a 
e n e g M g l g l m E E w n g M s m i a S n e i g M g s galim a E g M s i a i s i ass a;( ) 
e n e r d d r i s ass s; 
t nt tn ou f occ C n d m ] E a . sa [. i a= ) 
e t d d r t nt tn ou occ C a & s m s E a . sa [. i ; a 
e e t e t o o e w ou t g h p l d t l h t n l h o e / t a f l cr i h a sa a i i sa 
e e t n r o r f g I t nt tn ou oed id = c; $ a . sa ;(i ) 
t nt tn ou f occ C n d m ] E a . sa [. i a= ) 
e e d d r t nt tn ou otc ass s = d m ] E a . sa [. i ; a 
} 
t e re A g e d r e ] d d r i Sl [ s p s := ass s a i ( ' ; 0) 
e t e t e t h r m o p l / h s a a s i a 
e t l n e b t j b u c j i c a s ; ) 
e e l m A t d r e A d r i a s s s ( s s s ; ) 
e e l t m l B d o b d o T i a i P a y ( y ) 
t e n d l m h // S s i a . S i R a s [ = 
e n g M g l l m l u E r s s i j a . S i R a s [ = 

```



コントローラでは、次の点に留意してください。

- メールの件名と本文は、別の Visualforce ページで設定され、コントローラに渡されます。
- メールを送信するメソッドは `sendEmail()` と呼ばれます。この名前は、メールを送信する Visualforce ボタンのアクション名と一致する必要があります。
- メールの受信者、`to` フィールドに保存されているメールアドレスは、関連付けられた取引先で使用可能な取引先責任者のアドレスから作成されます。取引先責任者、リード、またはその他のレコードから受信者のリストをコンパイルするとき、すべてのレコードをループ処理してメールアドレスがレコードごとに定義されているか確認することをお勧めします。取引先 ID がページの URL から取得されます。

**Send an email to your
Burlington Textiles Corp of
America representatives**

Fill out the fields below to send an email.

Name	Email
Frankie	guy1@acme.com
Jack Rogers	guy2@acme.com

Subject:

Body:

Send Email

図 25 : sendEmailPage のフォーム例

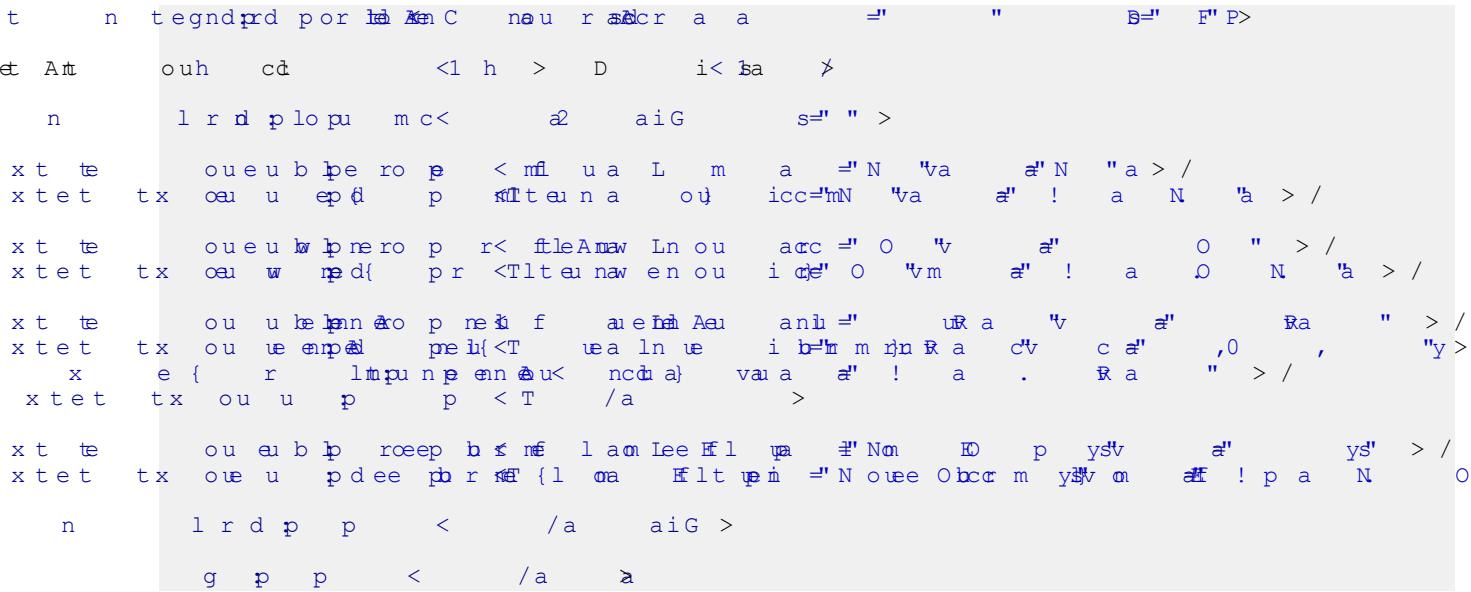
メール添付ファイルの作成

メールに添付ファイルを追加する場合、必要なのはカスタムコントローラに数行のコードを追加することだけです。メール添付ファイルのファイルの種類は `Apex` です。添付ファイルを作成するには、Apex

クラスを使用する必要があります。ファイル名とオブジェクトの内容の両方を定義する必要があります。

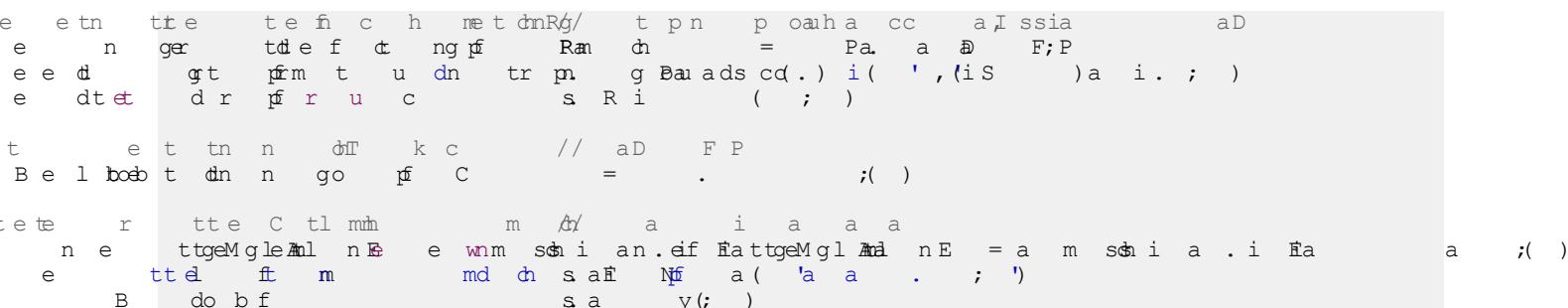
PDF添付ファイルの追加

次の例は、PDFとして表示されるVisualforceページへの添付をメール添付ファイルに変換する方法を示します。最初に、`PDF`というページを作成します。



メモ: PDF添付ファイルでの使用が推奨されるコンポーネントの詳細は、「PDFを表示するためのベストプラクティス」(ページ285)を参照してください。

次に、カスタムコントローラのメソッドでオブジェクトを作成します。次の例では、`PDF`のコードよりも前に配置する必要があります。



オブジェクトの名前が`a`の場合、添付ファイルを次のように関連付けます。

```
attachment att1 = new attachment();
att1.name = 'PDF.pdf';
att1.parent_id = 'a';
att1.type = 'application/pdf';
att1.body = PDFController.getPDF();
```

カスタムコンポーネントの添付ファイルとしての定義

カスタムコンポーネントを作成してVisualforceメールフォームで使用し、メールのPDFを表示することで、ユーザは送信しようとしている内容のプレビューを表示できます。

次のマークアップは、メールの添付ファイルを表す、`m dh` という名前のカスタムコンポーネントを定義します。

```

t en o om px pl bcd a ss=" " a
et At ouh cd <1 h > D i< ka >
n l r d p lopu m c< a2 aiG s=" " >
x t te oueub p e ro p < m u a L m a = N "va a" N "a > /
xt et tx oeu u ep d p < T t eu n a o u icc=MN "va a" ! a N "a > /
x t te oueub w p e ro p r < fileAnaw Ln ou aec = O "v a" O " > /
xt et tx oeu w npd{ pr < T t eu naw en ou i ej O "vm a" ! a O N "a > /
x t te ou ue h p e n f aueh Aeu anh = uRa "v a" Ra " > /
xt et tx ou ue emp& p e n < T ue a ln ue i b m m jn Ra c v c a" ,0 , "y >
x t et tx ou u p p < T /a >
x t te ou ue b p roep b k ne l am Lee Hl ya # N oee Obcc m y sv a! p a N > /
xt et tx ou ue u p d ee p b r & { l ma Hlt p n = N oee Obcc m y sv a! p a N > /
n l r d p p < /a aiG >
t n n o om px p < /a >

```

`m dh` ページを次のように置き換えます。

```

t n t egndprd por h m c nau r a cr a a = a " B=" F" P>
et t n c:m dh < a a > /
g p p < /a >

```

`n e` 次に、前の `m dh` の下部に表示するカスタムコンポーネントを追加します。

```

e tB el go p e h p at< A k a ah micro m v dP cc m a ! a . "a >
t n c:m dh < a a > /
B l go p p c < k/a a >

```

添付ファイルとプレビューの両方を変更する場合、`m dh` カスタムコンポーネントを変更するのは、どちらか一方の場所のみで済みます。

例: 添付ファイル付きメールの送信

`e` 次の例では、前の `m dh` の例に、Visualforce ページを添付ファイルとして追加するカスタムコンポーネントが加えられています。最初に、コントローラが次の処理を行います。

```

e blu { h p d c l m E i ss s i a
t hein { tre te bcd t j lgc i is s ; s ;
t blin e r te bcd lg i is y ; s ;
r n t Am p t onf cou v cc a a a ;
t nt t @ r ueret c olv/t eAa ptaps ouh bo cc j a c sa
t n d plcm E i s i a ( )
t n oau cc E E C T a =L [S N a,
t tre o E Et C lco L (l m A n t E tn o M a g c a. i c E R O . sa )
t An ou M cc FR O
e ex d A e E E t E n e d r n t p e d r g e =gt rm Ba . gd Pa( . ) Pa a s ( . ) i ( )
etblAat n rt rm ou cc i a ; ( )

```

```

e e e e b l e   e n { g r n      d f c      i      Ra      s      ( )
e   e t e n     fl mh      D      / i      i a
e   n e   g l e g   g l m e   eE wng Msmi a m S neig Mgs galim a E g Mss i a i s i ass a;

// Reference the attachment page and pass in the account ID
PageReference pdf = Page.attachmentPDF;
pdf.getParameters().put('id',(String)account.id);
pdf.setRedirect(true);

// Take the PDF content
Blob b = pdf.getContent();

// Create the email attachment
Messaging.EmailFileAttachment efa = new Messaging.EmailFileAttachment();
efa.setFileName('attachment.pdf');
efa.setBody(b);

e n e r      d d g      i s      ass s;
t   n t tn ou  focc  {C n d m ] E a . sa [.i a= )}
e   d d r t   n t tn ou  occ  C a&sm s E a . sa [.i a= )
e   t   e t o o e m ou  tg hp  l d t l h t n L h o e t / m e
e   e n ro      r fg I t   n t tn ou  oed{ id = c; & a . stat s ;(i ) )
e   e   d d r      t   n t tn ou  oec assGs =d m ]' E a . sa [ . i ; a
e   }

}

n t e re      A g e d r e ]      d d r      i Sl      [      s p s :=      ass s s i      ( , , 0)

t e t      e t      e t e hrm      o p      l / h s      a a s      i a
e   t e t l n e b t j bu      cj      ic &      S      s ; ) )
e   t e el m A t d r e A d r i      ss s (      ss s ; ) )
e   t e l t m l B do b do T i      i P a      y(      y )

email.setFileAttachments(new Messaging.EmailFileAttachment[] {efa});

e t e n      d      l m h      // ss      i a
e   n     g M t g l l m l u E r ss i a . S i Ras      [ =
e   n     n e   g M t g l l m e E n e ss M g l a s g l mi aE( g M s i m a i s i ass a [ i a; )

n n     ru      l h      ;
}

```

次に、Visualforce ページがメールを送信します。

```

x e t e n e      g o r p l h p c      d < l m a E a      " s      i " a>
x      m &      < a ss sa > /
x      t e B e l g o p n l p n c t d < k l a m o t a n e e & u      S e c n m a i      a Y      p a . a R      s v i a s
x      t e l l e o u p t e l d h l c f w o & t o e u n h m g t d e h l m o u r y i s a p i a s a

e x t e e d{ b l p l t u n t T s o u p c t a t a c & ! b a r d r c s a " v " a " a 1 " " >
e   x n l o u m p < a >
e   x t e e n e e p & d r e x & t m a a a & p f " a N < a / a a >
{ t t n e o } c m c ! a & a
e   x n l o u m p < / a >
e   x t e e n e p & d r e x & t m a E a & p f " a > & a / a a >
{ t t n e o } c l m c E ! a i a
e   x n l o u m p < / a >
x t e d b l p & / a a a a >

x      ro brm p b r f < a < > < > / > /
e x t t e e o u u & b t l p & j t a o L b u & a br a b s: " = " s " & / > / " 8 & /
e x e t n t x e { u e p l t u p & j t a & x b u t v n j l a ' m c l s g " i h = " s " a = " " 8 & /
e   x br br

```

```
te eou ub lp l B y dø Bro a dø f bva æ" " y = " y◊ / > /  
t n et x e û p lp u ßb do ãw do r va a æ" lo! "yic = " y 1s=" " 0 s=" " 8 ø  
br br < < > / > /  
  
tm øBndam ø o ð u <td{ eãnm ñø }v cd æ" s i "a ia =! ls " > /  
ro m p f < /a >  
B l go p p c < k/a a >  
  
<apex:pageBlock title="Preview the Attachment for {!account.name}">  
    <c:attachment/>  
</apex:pageBlock>  
g p p < /a æ
```

Visualforce メールテンプレート

開発者と管理者は Visualforce を使用してメールテンプレートを作成できます。標準 HTML メールテンプレートと比べて、Visualforce を使用する利点は、Visualforce では、受信者に送信するデータに対して高度な操作を実行できることです。

Visualforce メールテンプレートは標準 Visualforce コンポーネントを使用しますが、作成方法が異なります。

Visualforce メールテンプレートは常に、名前空間で始まるコンポーネントを使用します。その他に、次の操作を実行できます。

- すべての Visualforce メールテンプレートは 1 つの `<tg gl m l: m < Tss` タグ内に含まれている必要があります。これは、1 つの `g p p` タグ内に定義される通常の Visualforce ページと似ています。
 - `< tg gl m l: m < Tss` タグには、単一の `g dg Bml indo E < ss` タグか、単一の `a` タグか、`g1 g :B l pmds E Tss i a` タグのいずれかを含める必要があります。
 - 複数の標準 Visualforce コンポーネントを `<tg gl m l: m < Tss` 内で使用することはできません。これらには `ex1 p B`、`l go p pa c ie`、およびすべての関連する `p c` コンポーネント、さらに `m p f` などのすべての入力コンポーネントが含まれます。これらのコンポーネントを使用する Visualforce メールテンプレートを保存しようとすると、エラーメッセージが表示されます。

詳細は、次のトピックを参照してください。

- Visualforce メールテンプレートの作成
 - Visualforce メールテンプレートでのカスタムスタイルシートの使用
 - ファイルの添付
 - Visualforce メールテンプレート内のカスタムコントローラの使用

Visualforce メールテンプレートの作成

Visualforce メールテンプレートを作成する手順は、次のとおりです。

- #### 1. 次のいずれかの操作を実行します。

- ・ 公開テンプレートを編集する権限がある場合は、[設定]で[コミュニケーションテンプレート] > [メールテンプレート]をクリックします。
 - ・ 公開テンプレートを編集する権限がない場合は、任意の Salesforce ページ上部で、名前の横にある下向き矢印をクリックします。名前の下にあるメニューで、[設定]または[私の設定]のどちらか表示されるほうを選択します。次に、ページの左側で[メール] > [私のテンプレート]をクリックします。

2. [新規テンプレート] をクリックします。

3. [fc] を選択し、[次へ] をタリックします。

Visualforce メールテンプレートを使用して一括メール送信はできません。

4. テンプレートを保存するフォルダを選択します。

5. ユーザがメールを送信するときにこのテンプレートを使用できるようにするには、[有効] チェックボックスをオンにします。

6. [メールテンプレート名] を入力します。

7. 必要に応じて、[テンプレートの一意の名前] を変更します。この名前は、Force.com API の使用時にコンポーネントを参照するために使用される一意の名前です。管理パッケージでは、この一意の名前により、パッケージのインストール時に名前が競合することを防ぎます。この名前は、アンダースコアと英数字のみを含み、組織内で一意の名前にする必要があります。最初は文字であること、スペースは使用しない、最後にアンダースコアを使用しない、2つ続けてアンダースコアを使用しないという制約があります。[テンプレートの一意の名前] 項目を使用すると、開発者は管理パッケージで特定のコンポーネント名を変更できます。この変更は、登録者の組織に反映されます。

8. [文字コード] 設定を選択し、テンプレートの文字セットを指定します。

9. [テンプレートの説明] を入力します。テンプレートの名前と説明は、内部的にのみ使用されます。

10. [メール件名] にテンプレートの件名を入力します。

11. [受信者種別] ドロップダウンリストで、メールテンプレートを受け取る受信者の種別を選択します。

12. [関連先種別] ドロップダウンリストで、テンプレートが差し込み項目データを取得するオブジェクトを必要に応じて選択します。

13. [保存] をクリックします。

14. [メールテンプレートの表示] ページで、[テンプレートを編集] をクリックします。

15. [テンプレートの編集] をクリックし、Visualforce メールテンプレートのマークアップテキストを入力します。



メモ: 画像を含める場合は、サーバ上にある画像のコピーを参照できるように [ドキュメント] タブにアップロードすることをお勧めします。次に例を示します。

```
es      e   ngl  pto o gl ueñ e à he ihe pret ve o trñe ec er l /7 sa sang r I. sv / sv /
d      w      d      c pt1 D 015D000@600 i DB M 002 @00@000 Ga la = 7 705 68600
t      wt     g h d     h     i     =" " 46     =" " 46 /
```

16. [Version Settings (バージョン設定)] をクリックして、このメールテンプレートで使用する Visualforce および API のバージョンを指定します。組織で AppExchange の管理パッケージをインストールしている場合は、このメールテンプレートで使用する各管理パッケージのバージョンも指定できます。通常は、すべてのバージョンにデフォルト値を使用してください。デフォルトでは、メールテンプレートは Visualforce、API、および各管理パッケージの最新のバージョンに関連付けられます。特定の動作を維持するには、Visualforce および API の古いバージョンを指定できます。最新バージョンのパッケージのものとは異なるコンポーネントや機能にアクセスする場合は、管理パッケージの古いバージョンを指定することもできます。

17. [Save (保存)] をクリックし、変更を保存してテンプレートの詳細を参照するか、[Quick Save (適用)] をクリックし、変更を保存してテンプレートの編集を続行します。テンプレートを保存するには、Visualforce マークアップが有効になっている必要があります。



メモ: Visualforce メールテンプレートの最大サイズは、1 MB です。

Visualforce メールテンプレートを使用して一括メール送信はできません。差し込み項目の

`en {g e} enfield_name と gs_ .} field_name` は、一括メールでのみ機能し、
Visualforce メールテンプレートでは使用できません。

次の例は、取引先責任者に関連付けられたすべてのケースを表示する Visualforce メールテンプレートの定義方法

を示します。この例では、`p p` タグを使用して取引先責任者に関連するすべてのケースを反復処理し、テンプレートの本文に取り込みます。

```

ne me tg glem item < t Tss ic a qpi a pC a i i y=" " a
r l d to An ou Tpcc a y=" "
e bu t r c fCamo e t qps e roch d: bao m T ! a . " a
e l e u g gme n l c jup g a =a b i i . a a "
r tee o urop T mapp m c y=" s a. " >
tn m g g Bml indo E < ss i a i a y
l m h < >
b do < >
e e t e n n p } c m D > a! pi i . < a , >/
w B t oe d p e to { t ex le fcd a s lni a } o s sm Ta ! p a . < a . >/
e bl b rd r < a = " & >
r < >
t t t h r Gm tn r h gn > s N h < > & > & >
t r t rh t ltm t E < h u h a h a < > & sa >
x t r xe { r p e p t l u < r l ad > ov a = " v C T a ! a . s sa" >
r < >
e d r h f < < > a =
t en e h l : p r a o } & does 1 c / sa }sa bcr nC. ! / i . " > ! . s Na
x d tn yr g d < > ! & i < >
x t dtn ot c l m c d < > ! . a i < a >
x t d t } u c d < > ! . S < sa >
ex t r p p < / a > a
bl p < / a>
n r c < >
x t t enou utt p l wup < > Kal E pr a v o & c // s sa . " >
e te ro ro m ntl d nrot emolF g o l i rai o ifica c i Ssa .
x t t nou u p p < > Ka L i >
n r c < >
b do l m h < >
tn m g g Bml indo E < ss i a i a y
ne me tg gl m l: m < Tss i a i a a

```

このマークアップでは、次の点に留意してください。

- **属性** `rlp d` と `i i` はメールテンプレートのコントローラとして機能します。これらを使用して、他の標準コントローラで使用できるものと同じ差し込み項目にアクセスできます。
- **属性** `e` メールの受信者を表します。
- **属性** `c p` メールに関連付けるレコードを表します。
- **属性** `en` コンポーネントには、Visualforce マークアップと HTML を両方とも含めることができます。
- **属性** `gl g` `B l pnd` `E Tss i a` コンポーネントには、Visualforce マークアップとブレンテキストのみを含めることができます。
- 受信者または関連オブジェクトの言語に基づいて Visualforce メールテンプレートを翻訳するには、`ne me tg gl m l: m < l Tss i a g i a` 属性を使用します(有効な値は、「en-US」などの Salesforce

サポート言語キー)。言語属性は、メールテンプレートの `text/html` および `application/json` 属性の差込項目を受け取ります。差し込み項目で使用するカスタム言語項目を作成します。メールテンプレートの翻訳には、トランクルーションワークベンチが必要です。この例では、差し込み項目を使用して、メールを受信する取引先責任者の名前を取得します。

関連リンク

[Visualforce メールテンプレートでのカスタムスタイルシートの使用](#)

Visualforce メールテンプレートでのカスタムスタイルシートの使用

デフォルトでは、Visualforce メールテンプレートは、他の Salesforce コンポーネントの標準のデザインを必ず使用します。ただし、自分のスタイルシートを定義してこれらのスタイルを拡張したり、上書きしたりすることができます。

他の Visualforce ページとは異なり、Visualforce メールテンプレートでは、参照されている [ページスタイル](#) または [静的リソース](#) を使用することはできません。CSS はメールテンプレートのプレビューペインでは表示されるようになりますが、メール受信者にはプレビューペインと同じように表示されません。タグ内の CSS を `style` タグで定義する必要があります。

次の例では、メールのフォントを Courier に変更し、テーブルに境界線を追加し、テーブルの行の色を変更します。

```

ne me te glem item < t Tis ic a qpi a pC a i i y=" " a
r l d to An ou Tpcc a y=" "
e bu t r c fCano e t qps e ro&h d: >ao m T ! a . " a
r le o urop T mapp m c y=" s a. " >
tn m g g Bml indo E < ss i a i a y
l m h < >
e t tx p <c s y y=" ss" >
t bndo @ le mf exft : c } y :± 2a y p i ; $ 1 ;
bl
wt b rd rx d h : p i - ; 5
b rd n x g c p : p s - i a ; 5
et b rd r l d d : h s - y s a;
b rd r loro c : - #FF0000
bn r o d doro ck : a - #FFFFFF;

d
wt b rd rx d h : p i -1 ;
n xdd g p : p i a # 4;
et b rd r l lod : s - y s i ;
b rd r loro c : - # 000000
bn r o d doro ck : EEC@ - #FF ;
```

```

h
wt b rd rx d c : h : p i -1 # 000000
n xdd g p : p i a 4;
et b rd r l lod : s - y s i ;
b rd r loro c : - # 000000
bn r o d doro ck : a - #FFFF ; 0

1
b do < s / y >
< y
e e t e n n p } c m p D > a! pi i . < a , >/

```

```

e     bl b rd r      <      a      = " "  >
r
t     e t h r gm t n r < > s & < > & > & >
r     r t r dh t l t m t E < h u > a & a < > & sa >
x     t r x e { r p e p t l u < r l ad > ov a = " a " v C T a ! a . s sa" >
r
e     d r h f < < > a =
en    e h l : p r o < o > does 1 c // sa }sa bcr n. ! / i . " > ! . s Na
t     d d tn jr g d < > ! & i < >
x     t d ot c cl mcd < > ! . a i < a >
x     t d t } u c d < > ! . S<sa >
r
ex    t r p p < > / a > a
bl
b do
l m h
tn   m g g Bml indo < > s/s i a i a > y
ne   m tg gl m l: m < > s/s i a i a > y

```

Email Template Edit Template Preview

Subject Case report for Account: Joyce Bookings

HTML Preview

Dear Andy Young,

Below is a list of cases related to Joyce Bookings.

Case Number	Origin	Creator Email	Status
00001026	Phone	a_young@acme.com	New
00001027	Web	a_young@acme.com	New

[For more detailed information login to Salesforce.com](#)

図 26: 表示されている Visualforce メールテンプレートの例

カスタムコンポーネントの Visualforce スタイルシートの定義

Visualforce メールテンプレートの外部スタイルシートを参照することはできませんが、他の場所で参照できるカスタムコンポーネント内にスタイル定義を配置できます。たとえば、前例を変更して、`caseNumber` という名前のコンポーネントにスタイル情報を配置できます。

```

t en n o om p p l b g a ss=" " a
e t tx p < c s y y=" ss=" >
t bndo @ le mf oxfr t : c } y : i 2a y p i ; $ 1 ;
wt bl
b rd r x d h : p i - ; 5
b rd n x g c p : p s - i a ; 5
et b rd r l d d : h s - y s a;
b rd r l o r o c : - #FF0000
bn r o d o r o ck : a - #FFFFFF;

```

```

d
wt b rd rx d      h :      p      i a   i -1      ;
n xdd g p :      p      i a   i -1      ;
@ b rd r l lod :      s - y   s i ;      #
b rd r loro c :      -      # 000000
bn r o doro ck :      EEC@      -      #FF      ;

h
wt b rd rx d      h :      p      i a   i -1      ;
n xdd g p :      p      i a   i -1      ;
@ b rd r l lod :      s - y   s i ;      #
b rd r loro c :      -      # 000000
bn r o doro ck :      a      -      #FFFFF ; 0

l < s / y >
tn n o om px p < /a >

```

その結果、Visualforce メールテンプレートで、そのコンポーネントのみを参照できます。

```

tn m g g Bml indo E < ss i a i a y
l m h < >
l m lc:E < i a Sy > /
b do < y >
e t e n n p } c mp D > a! pi i . < a , > /
b do < ... >
l m h < /y >
tn m g g Bml indo E < ss i a i a y

```



メモ: Visualforce メールテンプレート内で使用される `om px` `pl` タグには `a` のアクセスペルが必要です。

ファイルの添付

Visualforce メールテンプレートに添付ファイルを追加することができます。各添付ファイルは、1 つの

コードでは、HTML と Visualforce タグを組み合わせることができます。

前の例は、データを反復処理してメール受信者に表示することで、Visualforce メールテンプレートを作成する方法を示しています。次の例は、データを添付ファイルとして表示するようにそのマークアップを変更する方法を示しています。

```

ne me te glem item < t Tis ic a qpi a pC a i i y=" " a
r l d to An ou Tpcc a y=" "
e bu tj rc fC Cane t qsa e roh d: jao m T ! a . " a
r t eo o urop T mpp m c y=" s a. " >
tn m g g Bml indo E < ss i a i a y
l m h < >
b do < y >
e t e n n p } c mp D > a! pi i . < a , > /
A t d e pdh e to { t< er le ftd a s lni a } o s sm Ta ! p a . < a . > /
n r c < >
x t t enou utt p l wuep < khal E pr a v o c // s sa . " >
e te ro ro m ntl d nrot emolF g o l i rai o ifica c i Ssa .
x t t nou u p p < Ka L i >
n r c < / >

```

```

b do < / y
l m h < / >
tn m g & Bml indo EK ss i a i a y

t t e m t g n g :m d< ss i a a a >
t r xe { r p e p t l u< e r l ad }ov a =a "v C T a! ! a . s sa" >
{ ex b re&n : bcr n s & ! . s Na
n ix g n }r : g & i ! & i
{ r ro x tle mn E o: )c D m c a i a ! . a i a
{ xthtr Gm } : uc s & ! . S sa
t r p p < /a >a
t t e m t g n g :m d< ss i a a a >
ne me tg gl m l: m < Tss i a i a a

```

このマークアップは、添付データファイルとして、書式設定なしでメールに表示されます。次のいずれかのオプションを使用すると、データをより読みやすい形式で表示できます。

- ファイル名の変更

r & r 属性の変更 s

- スタイルと画像の追加

ファイル名の変更

t t e m t g n g :m d< タグには、添付されたファイルの名前を定義する `m f` という属性があります。わかりやすい名前を定義することをお勧めしますが、必ずしもそのようにする必要はありません。未定義のままにしておくと、Salesforce が名前を生成します。

拡張子のないファイル名は、デフォルトでテキストファイルになります。添付ファイルは CSV として表示できます。

```

t t e m t e g n g :m l d< ms i a c a a i =a s sav.s >
t r xe { r p e p t l u< e r l ad }ov a =a "v C T a! ! a . s sa" >
} bcr n ! . s Na
n }r g ! & i
{ en o )c D m c ! . a i a
{ t r uc ! . S sa
t r p p < /a >a
t t e m t g n g :m d< ss i a a a >

```

データを HTML ファイルとして表示することもできます。

```

t t e m t e g n g :m l d< ms i a l c m a h a i =a s sa . " >
l m h < / >
b do < / y
bl b rd r < / a = " " &
r < / >
e t t h r Gm t n r & d< s & & < > & > & & > & & >
t r t r d h t l t m t E < h u & a & < > & > & & > & & >
r < / >
t r xe { r p e p t l u< e r l ad }ov a =a "v C T a! ! a . s sa" >
r < / >
d r h f < < > a =
en e h{ l : p r a( o ) bcr n c // sa }sa bcr n . ! / i . " > ! . s Na
d < < /a >
x d tn }r g d < / > ! & i < / >
x t e l t n o t )c D m c d < / > ! . a i < / >
x t d t } u c d < / > ! . S sa < / >
r < / >
t r p p < /a >a
bl < / a>
b do < / y
l m h < / >
t t e m t g n g :m d< ss i a a a >

```

gn g :m d < コンポーネントごとに定義できるファイル名は1つのみですが、メールには複数のファイルを添付できます。

renderAs 属性の変更

他の Visualforce ページと同様に、コンポーネントの属性を PDF に設定すると、添付ファイルが PDF として表示されます。次に例を示します。

the met eng n :m &dr<e ss i la a maf d c s=" F" Pfl =a s sa . " >
l m h < >
b do < >y
n ou{ de le t que r spne d & o ians Ta yc ! a : a p s sa sD a <@
bl b rd r < a = " " &
r < >
e t t h r Gn t n r & gh > s N & < > @ & > & sa >
t r t rdh t ltm t E < h u & a & a < > & sa >
r < >
t r xe { r p e ptl u & r l ad > ov a = " a " v C T & ! a . s sa" >
r < >
d r h f < < > a =
en e h(l : p ro{ o) & does 1 c // sa }sa bcr nc. ! / i . " > ! . s Na
d < < >/ a >
x d tn }r g d < > ! & i < >
x t edtn ot c d mcd < > ! . a . i < a >
xt t d t } u c d < > ! . & sa >
r < >
t r p p < / a >a
bl < / a>
b do < / y
l m h < / >
the mt qn q :m & < ss i a a a >

リリースする前に、表示されるページのフォーマットを確認してください。
に留意してください。

を使用する場合、次の点

- ・ サポートされている表示サービスは PDF のみです。
 - ・ Visualforce ページを PDF として表示する機能は、印刷用にデザインされ、最適化されたページのためのものです。
 - ・ 印刷用の書式設定が容易ではないか、入力やボタンなどのフォーム要素が含まれる標準コンポーネント、または書式設定に JavaScript が必要なコンポーネントは使用しないでください。これには、フォーム要素が必要なコンポーネントなどが含まれますが、これに限定されません。
 - ・ PDF 表示では、JavaScript で表示されるコンテンツはサポートされていません。
 - ・ PDF でページのすべてのテキスト（特に日本語などのマルチバイト文字やアクセント記号付きの国際文字）が表示されない場合は、CSS のフォントを調整してそれに対応するフォントを使用します。次に例を示します。

これはサンプルページです。

選択したフォントは、Visualforce PDF 表示サービスで使用できる必要があります。現在、マルチバイト文字を含む拡張文字でサポートされているフォントは「Arial Unicode MS」のみです。

- PDF 作成時の最大応答サイズは、PDF として表示される前で 15 MB 未満です。これは Visualforce 要求の標準制限です。
- 生成される PDF の最大ファイルサイズは、60 MB です。
- 生成された PDF に含まれるすべての画像の最大合計サイズは 30 MB です。
- PDF 表示では、URI スキーム形式で符号化された画像はサポートされていません。
- 次のコンポーネントは、PDF として表示するときに 2 バイトのフォントをサポートしません。

```
B ◊ l go p p c < k a a >
x t e e n ◊ o φ rc < as 直 a>
```

PDF として表示するページでこのようなコンポーネントを使用することはお勧めしません。

スタイルと画像の追加

添付ファイルでは、スタイルシートを使用してデータの表示方法を変更することもできます。スタイルは、Visualforce メールテンプレートと同じ方法（INLINE-CODE として、またはカスタムコンポーネントを使用して）で、添付ファイルに関連付けられています。

PDF として表示される添付ファイルは、[グローバル変数](#)を使用して静的リソースを参照できます。これにより、PDF の本文内の画像またはスタイルシートを参照できます。

たとえば、次の添付ファイルでは PDF にロゴが含まれます。

```
title met eng n :m Adhr< ss i la a maf d c B=" F" P> =a s sa . " >
l m h < >
b do < > y
{ e reg c ou} l o< o gc $ s = " ! R s . " > /
b do < ... >
l m h < / >
title met gng :m d< ss i a a a >
```

この添付ファイルは、静的リソースとして保存したスタイルシートを参照します。

```
title met eng n :m Adhr< ss i a a a B=" F" P>
l m h < >
etlee r b et le t tx k e h{ ie ps dys h oufA'y= } ' cs$/EM I' G ! R s L ss > /
b do < > y
b do < ... >
l m h < / >
title met gng :m d< ss i a a a >
```



警告: リモートサーバの静的リソースを参照すると、PDF 添付ファイルの表示に時間がかかる場合があります。Apex トリガで PDF 添付ファイルを作成する場合、リモートリソースは参照できません。参照すると例外になります。

Visualforce メールテンプレート内でのカスタムコントローラの使用

Visualforce メールテンプレートでは、高度にカスタマイズされたコンテンツを表示するためにカスタムコントローラを活用できます。このためには、そのカスタムコントローラを使用する Visualforce メールテンプレートにカスタムコンポーネントを含めます。

たとえば、メールテンプレートで「Smith」という語で始まるすべての取引先のリストを表示するとしています。このためには、まず、SOSL コールを使用するカスタムコントローラを作成して、「Smith」で始まる取引先のリストを返します。

```
blu  blu p d cAn m fou h ac ss a i s s
r n t ltpAn f toun c ou ac is > a s;
mhu d{ Arpcm fou h cc i i i s s ( )
n e due ccl te mano em oaf s eccs t h M a m E h ] NL a K i s ;
t blu te Ant ptcou tcen m du hsc > i s s ( )
n rt rm ou cc a s;
```

次に、このコントローラを使用する `u h cc` というカスタムコンポーネントを作成します。

```
t n nt nom p or lpr c< d Ama m fou h cc l="bad" i s s" ss=" " a
e ed{ bl pl u t m} oau h strava abu! i sc s"v =" a _a " >
n lo u mp < a >
te ene p fintdean e h ouex atcc ma =a "ap>f N< a / a a>
ten oj cc m ls -a N a
n lo u mp < /a >
e d bl p /a a a a>
t n n o om p < /a >
```



ヒント: Visualforce メールテンプレートで使用されているすべてのカスタムコンポーネントには `u h cc` のレベルが必要です。

最後に、`u h cc` コンポーネントを含む Visualforce メールテンプレートを作成します。

```
re me tg gl m tl em ba e j pashidam i gga de ns pC="tn c i p pC " c i i y" " a
r l d t o t n ro ut p pp a y" O i " y
tn m g d Bml ndo E < ss i a i a y
e a e t e on p qd t r < lh > s t yl us fn m su hi acc : a pi s a <s > /
t An m cou h cc < si s > /
e o wet le p th e < h > hd } as si T p a y . > /
tn m g d Bml ndo E s si a i a y
re me tg gl m l: m < ps i a i a a
```

`u h cc` コンポーネントには属性が必要ですが、この例ではこの属性は有効ではありません。この属性には、カスタムコンポーネントで使用されているオブジェクトとは異なるオブジェクト値をこの属性が取ることができるとということを示す目的でのみ `u h cc` という値が設定されています。



メモ: メールテンプレートで標準コントローラを使用する場合は、共有設定が適用されます。ユーザオブジェクトの組織の共有設定が [非公開] に設定されており、Visualforce メールテンプレートで名前やメールアドレスなどのユーザ情報にアクセスする必要がある場合は、`u h cc` キーワードを使用してカスタムコンポーネントまたはカスタムコントローラを使用できます。

ユーザオブジェクトの共有についての詳細は、Salesforce オンラインヘルプの「[ユーザ共有の概要](#)」を参照してください。

第 15 章

Visualforce Charting

Visualforce Charting は、単純で直観的な方法で Visualforce ページおよびカスタムコンポーネントにグラフを作成することができるコンポーネントのコレクションです。

Visualforce Charting とは?

Visualforce Charting では、SOQL クエリから直接作成するデータセットに基づいて、または独自の Apex コードでデータセットを作成することで、カスタマイズされたビジネスグラフを簡単に作成する方法を提供します。個別のデータ系列を組み合わせて設定することにより、組織にとって有用な方法でデータを表示するグラフを作成できます。

Visualforce グラフは、JavaScript を使用してクライアント側に表示されます。この機能により、アニメーション効果が設定されたグラフや視覚的に魅力的なグラフを作成できます。また、グラフのデータを非同期に読み込み/再読み込みすることで、反応が早いと感じさせるページを作成できます。

Visualforce Charting を使用する理由は?

標準の Salesforce グラフおよびダッシュボードでは不十分な場合や、組織にとってより有用になるようにグラフとデータテーブルを組み合わせるカスタムページを作成する場合は、Visualforce Charting を使用します。

Visualforce Charting に代わる機能

Salesforce では、さまざまなビジネスグラフをサポートする多数のダッシュボードおよびレポートを提供しています。Visualforce または Apex でのプログラミングが不要なため、これらのグラフをさらに簡単に作成およびカスタマイズできます。組み込みのグラフ作成およびレポート作成についての詳細は、Salesforce オンラインヘルプの Salesforce ヘルプの「ダッシュボードを使用した洞察の共有」を参照してください。

Visualforce Charting は柔軟性に富むだけでなく、簡単に使用できるように設計されています。ビジネスグラフィックスで一般的に使用されるさまざまな棒グラフ、折れ線グラフ、面グラフ、円グラフの他に、より専門的なグラフ作成用にレーダーグラフ、ゲージグラフ、散布図を提供します。他のタイプのグラフが必要であったり、ユーザまたはページとの高度な対話機能を追加する場合は、代わりに JavaScript Charting ライブラリの使用を検討することをお勧めします。このライブラリを使用すると作業量が増加しますが、より多くのカスタマイズを行うことができます。例として、「[Visualforce と Google Chart の統合](#)」(ページ 121)を参照してください。Visualforce での JavaScript ライブラリの使用方法についての詳細は、[Visualforce ページでの JavaScript の使用](#)(ページ 264)を参照してください。

Visualforce Charting の制限および考慮事項

このセクションでは、Visualforce Charting の考慮事項および既知の制限をリストします。

- Visualforce のグラフは SVG (Scalable Vector Graphics) をサポートするブラウザでのみ表示されます。詳細は、[「WC3 SVG Working Group」](#)を参照してください。
- Visualforce Charting では JavaScript を使用してグラフを描画します。Visualforce のグラフは PDF として表示されるページでは表示されません。
- メールクライアントは、通常、メッセージでの JavaScript の実行をサポートしません。メールメッセージまたはメールテンプレートでは Visualforce Charting を使用しないでください。
- Visualforce Charting は、JavaScript コンソールにエラーおよびメッセージを送信します。開発時は、Firebugなどの JavaScript のデバッグツールを有効な状態に保ってください。
- 動的 (Apex で生成される) グラフコンポーネントは、現在、サポートされていません。

Visualforce Charting のしくみ

Visualforce グラフは、一連のグラフコンポーネントを使用して定義され、その後コンポーネントはそのグラフでグラフ化するデータソースにリンクされます。

Visualforce でグラフを作成するには、次の操作を実行します。

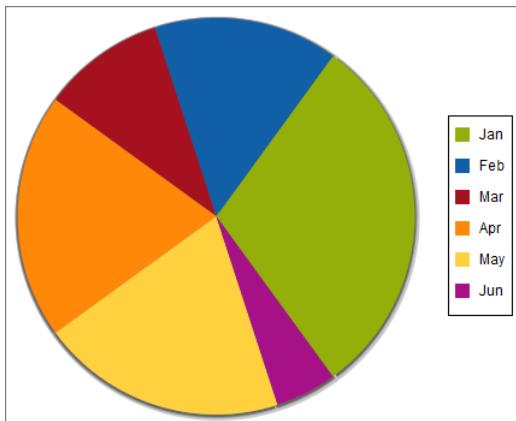
- グラフデータをクエリ、計算、およびラップして、ブラウザに送信する Apex メソッドを記述する。
- Visualforce Charting コンポーネントを使用してグラフを定義する。

グラフを含むページの読み込み時に、グラフデータがグラフコンポーネントにバインドされ、グラフを描画する JavaScript が生成されます。JavaScript を実行すると、グラフがブラウザ内で描画されます。

単純なグラフ作成の例

Visualforce グラフでは、少なくとも 1 つのデータ系列コンポーネントを囲むグラフコンテナコンポーネントを作成する必要があります。必要に応じて、追加の系列コンポーネント、グラフの軸、および凡例、グラフのラベルおよびデータポイントのツールチップなどのラベルコンポーネントを追加できます。

次は単純な円グラフとこの円グラフを作成するマークアップです。



```
x t e n e g o r p l d i p d e n <r te to a d h r a C l = " i r P a K " i = " i P " a>
x et r w t h g h h d < e adth ai } = " 3 " 50 p = " 4 50 = a &i " a >
e ex te r q t p l d e d < elain ld S s E m a = " " a a a = " " a > /
x t n t l d g t o r p g d h $ i = " i " > /
g p p < / a >
```

コンポーネントは、グラフコンテナを定義し、そのコンポーネントをデータソース、コントローラメソッドにバインドします。 は、返されるデータにアクセスしたり、各データポイントのラベル付けおよびサイズ設定をしたりするラベル項目およびデータ項目を記述します。

関連付けられているコントローラを次に示します。

```

e bl t t e p & cr or l̩ r Ci ss a P a
te e ble t e p̩ d t̩ g g l̩ W<s i D a aDi P a(a )
te e e t t d e g w̩ te e d̩ W<s i D g a a æ W<s i D a a )
e ed̩ e dd t d n g J a aWa( i D a(a 'aʒ' ;0))
e ed̩ e dd t e d g b a aWa( i D aFa1 ' , ' ;5)
e ed̩ e dd t d g r a aWa( i D a(a1 ' a, ' ;0))
e ed̩ e dd t d g r a aWa( p ɻ D a(a ' , ' ;0))
e ed̩ e dd t d n g uJ a aWa( i D a(a 'ya, ' ;0))
e ed̩ e dd t d n g uJ a aWa( i D a(a ' , ' ;5))

tn r ru d ;a a

r r l p̩ x W // a ss a
e bl e l { t p c d p̩ x g i W ss a D a a

t bl n er te p dg m jg i s a ; s ;
e et nbh{ t e ter pdg t I jg i a a ; s ;

e e e bl t t d p c ng ne ret n g t W m ir d D I a (ä S a,
t e ne n hm m i s . æ ; a
t t t d hd i s . æ a ; a a
}

```

このコントローラは、意図的に単純にしています。通常、データを収集するには 1 つ以上の SOQL クエリを発行します。

この例が示す重要なポイントを次に示します。

- メソッドは、`Dラッパー`として使用される内部クラス `PieWedgeData` という単純なオブジェクトのリストを返す。リストの各要素は、データポイントの作成に使用されます。
 - `PieWedgeData` クラスは単なるプロパティのセットであるが、基本的には、`=value` ストアとして使用される。
 - `eグラフ` 系列コンポーネント `r p p` では、系列内の各ポイントの決定に使用する `PieWedgeData` クラスのプロパティを定義する。この単純な例には不明確な部分はありませんが、複数の系列および軸を含むグラフでは、この規則により、1 つの `List` オブジェクトでデータセット全体を効率的に返すことができます。

グラフデータの提供

e Visualforce グラフは、コンポーネントのデータ属性を使用してそのデータのソースにバインドされます。

データは次の複数の方法で提供できます。

- ・ コントローラメソッド参照を表す式
 - ・ JavaScript 関数を表す文字列

- JavaScript 配列を表す文字列

関連リンク

[コントローラメソッドを使用したグラフデータの提供](#)

[JavaScript 関数を使用したグラフデータの提供](#)

[JavaScript 配列を使用したグラフデータの提供](#)

[グラフデータの形式](#)

コントローラメソッドを使用したグラフデータの提供

グラフにデータを提供する最も簡単な方法は、コントローラメソッドを参照する Visualforce 式を使用することです。`ctrl` 属性でコントローラを参照するだけですみます。

サーバ側では、[単純なグラフ作成の例](#)(ページ 197)にあるような独自の Apex ラッパーオブジェクト、`sObject`、またはオブジェクトに対することができるオブジェクトのリストを返すコントローラメソッドを記述します。このメソッドは、サーバ側で評価され、結果が JSON に逐次化されます。クライアントでは、`ctrl` がこれらの結果を直接使用するため、結果をさらに処理することはできません。

以下は、`sObject` を使用してこの技法を説明するための Opportunities(商談) のリストおよびその金額の棒グラフを返す単純なコントローラです。

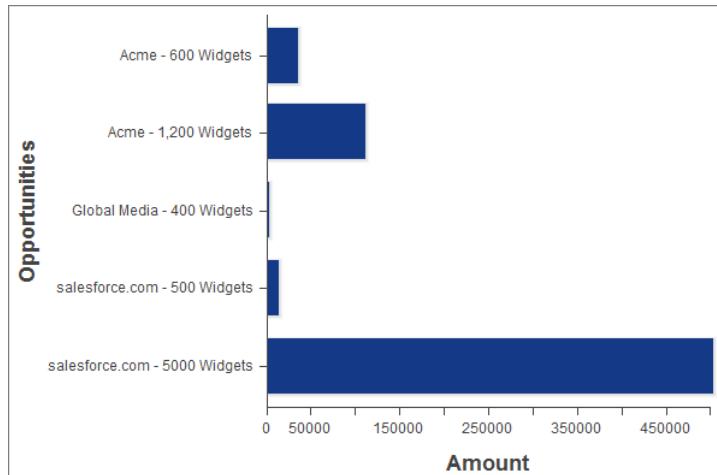
```

blt te n pcc or l& r pp i ss aO s
e t t et on ro u f pp /G sa O i i s
ex blt e nptc gte pde t{or l& m C oBa . S @ a S
e t n{ n o fllic i s( == )
e t e newn ex A t C ent gte pds t@rld retC b Bat S g@ a os ro D c ( a@ a L y
e t e net E m e To um lIpod [St ct na, r@ M a pp s ] @R O O i y;
e t e ten e o z C g s s Ra S (; 5)
t e tn r rm o C s ;
}
}

t blt ten ptro ut et pp g ro l u i <s pp i >y o i i s ( )
t ntr rut t m to e u en pp o L g@ s o ci >y s ) . R s ;( )
}

ten gor pl@epa < or l& a pp a " O s " >
t t r dt epn ro w@ e pp d ta "a@ O g h i hs" i " " 6@ " " 4 80
x ext e p t ero gipC@ oaliesap "d fa "ymeti n=" ro "u i sp@N "ai " " o
x ext e ptur m@p@e o@i o gao "m@nti "ofimisti la=" ou@i s=" " i "
e e e brt tn pron tzon rx atih a@ s bi o oi ar=" i " ai sa" "
x e e eld m@l oum @=" N "a@y " " > /
x t r ph < /a a
x te ed{ bl p@ eut nT<ro ju a parava o " !O ppi i s "v " " a " >
x e en do u m@rt l hunes fo ua elpp un " O m p@ "v " " a " ! . " a > /
x e en do u m@rt l hnu{ < oum d ut V no } aoum a@ pp "v " " a " ! . a " > /
x te d bl p " /a a a a>
g p p < /a a

```



この例に関する 2 つの重要な注意事項を次に示します。

- Visualforce グラフコンポーネントは、[単純なグラフ作成の例](#)（ページ 197）で使用される単純なデータオブジェクトと同じ方法で、Opportunity sObject のリストのデータ属性にアクセスする。
 - データ属性として使用されるオブジェクト項目名では JavaScript では大文字と小文字が区別されるが、Apex および Visualforce の項目名では区別されない。軸およびデータ系列コンポーネントの `label` 属性、`axis` 属性、および `series` 属性の正確な項目名を使用するように注意してください。正確な項目を使用しない場合は、グラフが作成されず、エラーメッセージも表示されません。

関連リンク

グラフデータの形式

<apex:actionSupport> を使用したグラフデータの更新

JavaScript 関数を使用したグラフデータの提供

JavaScript Remoting を使用してデータにアクセスする場合、または外部 (Salesforce 以外) データソースにアクセスする場合、データを提供する JavaScript 関数の名前を `js` コンポーネントに指定します。JavaScript 関数は、Visualforce ページで定義またはリンクされている必要があります。

この関数では、結果を [\\$`](#) に渡す前に結果を操作したり、他のユーザインターフェースまたはページの更新を実行したりできます。

JavaScript 関数はパラメータとしてコールバック関数を取り、関数のデータ結果オブジェクトを使用してコールバックを呼び出します。最も単純で有効な JavaScript 関数は次のようにになります。

g p p < a >
c p < s i >
d fg φ m llb ci c R D k a(a a a)
n e r et o e r t h r C t g o t m e P n e t u a φ t m R Di P a(a i (s v,
nt e t f a u r t l u t r i n t l u (o s r { u s a o A & s r c & s . s ==
e llb r c l u c k a a (s ;)
;)
c p < s i / >
e t r e d t p h g Φ m x a a a R p h " a & . > / a a

このグラフをサポートするには、[単純なグラフ作成の例](#)(ページ 197)で定義されている `or` メソッドに次のコントローラメソッドを追加します。

Ano	m	o	c	R	i												
bli	te	e	p c t e	e t d e	t e g	it{	g	ø	m a i s	i	D	a	a R	Di	P	a (a)	
e	t		td e	g	wd te e	Ø	W<§	i	E	g	a a	a	ø	W<s	i	D	a a)
edv̥n	edd	t	d	n g	J	a	aWa(i	D			a	(a	' a3'	; 0))		
edv̥n	edd	t e	d	g	b	a	aWa(i	D			a	Fal	', '	; 5))		
edv̥n	edd	t	d	g	r	a	aWa(i	D			a	(a1	' a,	; 0))		
edv̥n	edd	t	d	g	Ø	a	aWa(p 2	D			a	(a	', '	; 0))		
edv̥n	edd	t	d	g	r	a	aWa(2	D			a	(a	' ya,	; 0))		
edv̥n	edd	t	d	n g	uJ	a	aWa(i	D			a	(a	', '	; 5))		
tn	r	ru	d						; a a								

関連リンク

グラフデータの形式

Apex コントローラの JavaScript Remoting

JavaScript Remoting を使用したグラフデータの更新

JavaScript 配列を使用したグラフデータの提供

ページの独自の JavaScript コードで JavaScript 配列を作成して、その配列の名前をにより、Salesforce 以外のデータソースで Visualforce Charting を使用できます。

次の単純なコードは、この手法を示しています。

g p p < a a
 c p < s i >
 id t r dh err t / J r ca ap a aiya v aiS
 war A rr v a a a ya y()
 art u dt hp dt : a ad ya : (1 'na aBm , J' a a ', 66 '3a a '8 , 0 'a ' ' a ; ')
 art u dt hp dt : a ad ya : (1 'a aBm b 'a a ', 66 '3a a '8 , 0 'a F ' ' ; ')
 //..
 c p < s i / >
 tr d phdA err & t a a x =a ap a a " ya < . > / a a
 q p p < / a a x =a ap a a " ya < . >

この手法を使用すると、データを Salesforce 以外のソースから取得する場合、サーバ側の Apex コードは一切必要ないことがあります。

関連リンク

グラフデータの形式

グラフデータの形式

Visualforce グラフに提供されるデータは、特定の要件を満たしている必要があります。データ収集のどの要素にも、そのデータソースにバインドされているコンポーネント階層で参照されているすべての項目を含める必要があります。すべての項目が提供されない場合、クライアント側の JavaScript エラーが発生します。これは、Firebug などの JavaScript コンソールで参照できます。

Apex メソッドで提供されるグラフデータは一定のオブジェクトのリストである必要があります。これらのオブジェクトは、単純なラップバー、AsObject`または` `lu` オブジェクトの場合があります。データ項目は、公開メンバー変数またはプロパティとしてアクセス可能にできます。

JavaScript メソッドで提供されるグラフデータは配列の JavaScript 配列である必要があります。各内部配列は、レコードまたはデータポイントを表します。データ項目は、名前: 値ペアとしてアクセス可能にできます。例については、「[JavaScript 配列を使用したグラフデータの提供](#)」(ページ 201)を参照してください。

関連リンク

[JavaScript 配列を使用したグラフデータの提供](#)

Visualforce Charting を使用した複雑なグラフの作成

Visualforce Charting を使用して、さまざまなグラフコンポーネントから、関連データの複数のセットを表す複雑なグラフを作成します。最終的に、非常に洗練された注目を集めるグラフを作成できます。

グラフコントローラ

このトピックの後半の例では、次のコントローラを使用します。このコントローラは、[単純なグラフ作成の例](#)のコントローラを少し拡張したものです。リモート JavaScript 呼び出しでコールできる、より多くのデータおよびメソッドが含まれます。

```

blt lte fprc or k r C i ss a a
n t r u t l t o dn o tf R6/ p r i s dn aia s a a
t ble t t{ pc g L i s a a(a )
ntr trauner tor k t c gr k a . D a a;(a )
}

et t t Me dh dk l bl et// atJ r nra vama a y vai v a aiS i
t Ano m o c R i
t blt te eptc e t ct{ g oim s L a i s a a R D a a(a )
ntr trauner tor k t c gr k a . D a a;(a )

t t t t de h r a de dt d te// a a b a c;a a s s i a
e @ld d b t ce Ano m o ab/ a h y Ra i
t blt te ptc t ct{ gr i s k l a i s a a D a a(a )
t t e wd t t L i s a a a a i s a a(a )
e dwnt dd n J a a b( a(a '93 ,0 ,0 ,55)
e dwnt e dd b a a b( a(a '44 ,5 ;6)
e dwnt dd r M a2 a b( a(a '35 ,7 ;5)
e dwnt dd A r p a a b( a(a '74 8 8 ;5)
e dwnt dd M a a b( a(a 'ya ,1 9 ,6 5 ;5)
e dwnt dd n uJ a a b( a(a '33 99 4 ,5 ; ))
e dwnt dd luJ a2 a b( a(a ' , , 83 ;0)
e dwnt dd A u g a a b( a(a ' , , 78 ,8 , 4 ;5)
e dwnt e dd c a a b( a a s 3 ' 4 ,6 55)
e dwnt dd t c a a b( a(a ' , , 71 8 ,66 5 )
e dwnt dd o a a b( a a ' , , 8 ,07 ,8 5 )
e dwnt e dd c a a b( D a (a1 ' , , 7 ,7 ,0 ;0)
tn r ru d ; a a

r r l p c W // a ss a
b l p c c i D s s a a a
t e oin e r te p dg m lg i i s a ; s ;
e et nblt e te pdg ff lg i 1 a a ; s ;
e et nblt e te pdg ff lg 2 i a a ; s ;
e et nblt e te pdg ff lg i 3 a a ; s ;

```

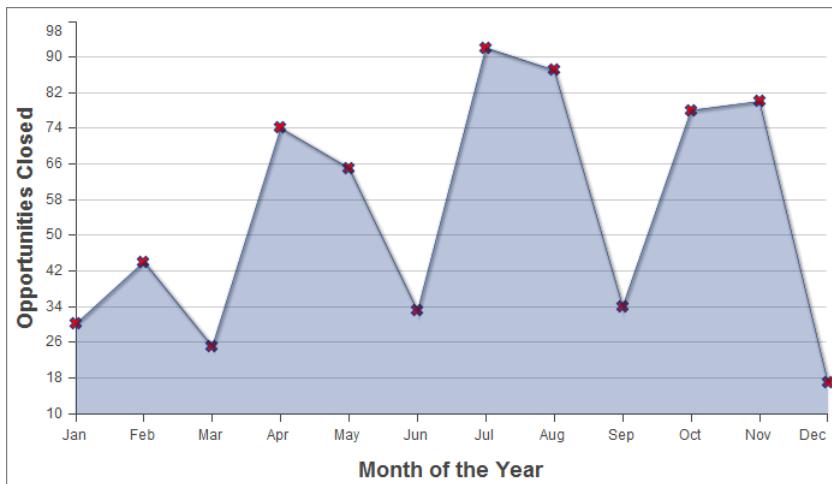
t t b l i e n r e p t c n g t m e e t r d i t a (t s d g I t a , r d i a a a a)
e n e n h m m i s . æ ; a
t t d h d i s 1. a=a1 æ a
t t d h d 2 2 i s . a=a æ a
t t d h d i s . 3a=a 3æ a



メソッドはこのトピックのグラフの例では使用されませんが、サーバ側メソッドおよび JavaScript Remoting メソッドの両方のデータ生成メソッドを再利用する方法を示します。

単純な折れ線グラフの作成

次は、1暦年で「成立した商談」というデータセットの3つのデータ系列の1つをグラフ化した単純な折れ線グラフです。

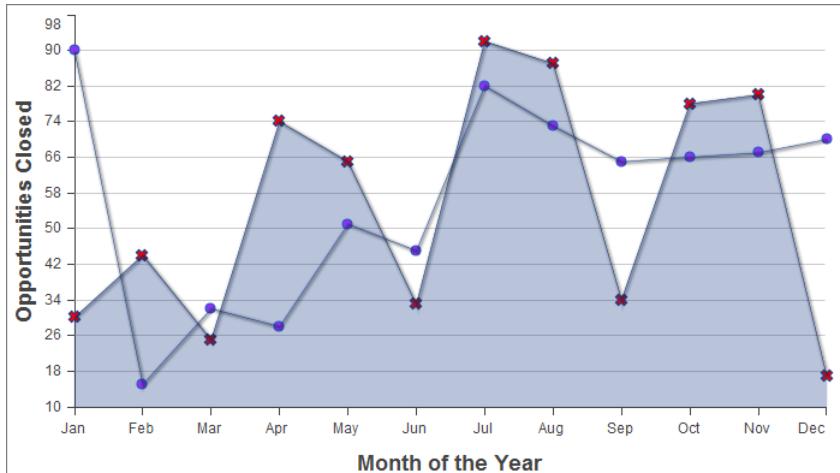


この例では、次の点を確認してください。

- ・ 折れ線グラフおよび棒グラフでは、グラフの X 軸と Y 軸を定義する必要がある。
 - ・ 縦軸はグラフの左側に定義され、その月に成立した商談の金額を示す。
 - ・ 横軸はグラフの下部に定義され、暦年の月を表す。
 - ・ 実際の折れ線グラフ、`r p` コンポーネントは特定の軸にバインドされる。
 - ・ グラフの各線の差別化に使用できるいくつかのマーカー属性がある。

2番目のデータ系列の追加

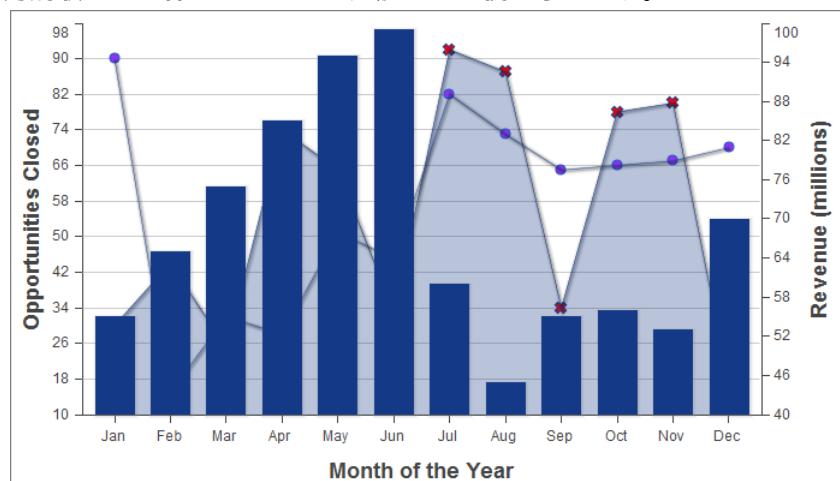
同じ測定単位を使用する2番目のデータ系列は簡単に追加できます。ここでは、「不成立の商談」データセットを2番目の折れ線系列として追加します。



重要なのは、1項目および2項目の両方がそのコンポーネントの項目属性によって縦の軸にバインドされている方法を確認することです。これにより、グラフ作成エンジンが軸の適切な目盛りと刻みのマークを特定できます。

2番目の軸を使用した棒グラフの系列を追加

他の単位でグラフを作成するもう1つのデータ系列を追加するには、2番目の縦軸を追加する必要があります。次に、データ系列「月別収益」が棒グラフとして追加された例を示します。



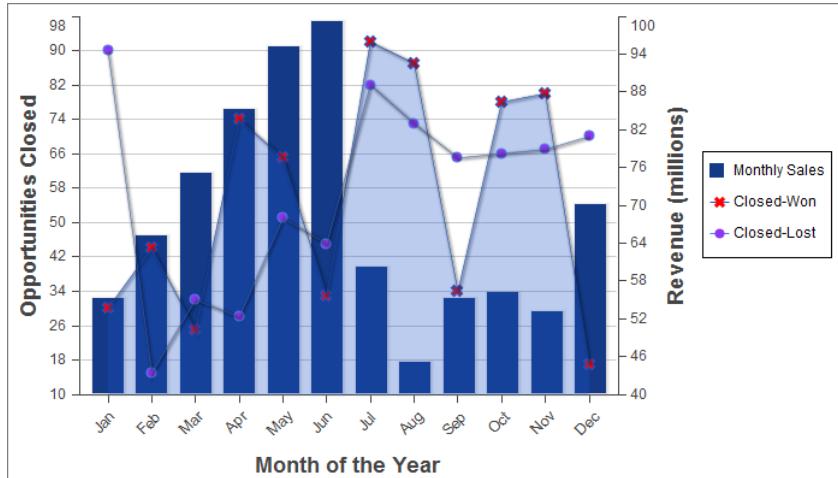
x xet e pt ur m n p o e o a i s t g y'dN h di "f s i = i " i s=" 3'a a
 te e e ln un ll m o i ="R i(i s" > /
 x xet e et p t ro g p o e oabi s a o y'mld a "y f i = " " i s=" " a
 te t n dte o o M hr fh i =" Y " a> /
 ex e en l e r t p d < e xl aefu fei Sd i s a t m " l d d=" " R = " a E y = 1 " a a
 e e r m e a o zr p m ar my il ss" aik S" " 4 aE = "#FF '0000 /
 ex e en l e r t p x le < ne ld aif t i s sldi d=" " R = " a E y = " " a a
 e e r m e a el zr p m ar my il " aik S" E 4 aE = "# 38 F" 5 > /
 re e e b r t n e p r o n t o < x at a s s i g i a w i " ai s a i "
 ex e n ld t m ld d R = " a E y = 3'a a> /
 x t r ph < /a >
 g p p < /a >

次の点を確認してください。

- 新しい測定単位を使用するデータ系統を追加するには、グラフの右側に2番目の縦軸を追加する必要がある。
- グラフの各境界に1つ、最大4つの異なる軸を設定できる。
- 棒グラフは縦方向に設定され、右の軸にバインドされている。横棒グラフは上軸または下軸にバインドされている。

凡例、ラベル、およびグラフのヒントの追加

グラフの凡例、系列ラベルを追加したり、グラフのラベルを確実に読むことができるようになりますことで、グラフを分かりやすくすることができます。



x t n gotplaten r < or Maar Ca = " a " >
 x et t r wh g h h < tad h ajid = " 4 00 = " 7 " 00 = a! " a
 e ex n tl d g top o r p g ah \$ i = " i " > /
 x xet e pt ur m n p o e o a i s t g y'dN h di "f s i = i " i s=" 1 " a a
 te t n dte o o M hr fh i =" Y " a
 ex e te et blm o < a L a a 1=a3 " s /
 x x te br r t p n l t iMaeh ronatS s i o = " lt y S i g i a w i " ai s a i "
 ex e te n ld t m ld d R = " a E y = 3'a a
 x e br r p < /a a s s >
 ex e en t t re p l ln d to eix lies sine lew f s t m " l d i d=" " R = " a E y = 1
 et ell er u fr m e a o izr p m " ar my il ss" aik S" " 4 aE = "#FF '0000 /
 ex e en t t re p l t l d e a i x i es i ne #d s t m " l d i d=" " R = " a E y = "
 ex e er m e a el zr p m ar my il " aik S" E 4 aE = "# 38 F" 5 > /
 x t r ph < /a >
 g p p < /a >

これらの追加については、次の点を確認してください。

- データ系列コンポーネントの順序が、グラフを描画する際のグラフ要素のレイヤを決定する。前の例では、棒グラフは前面に配置されました。`ex1` の例では、2つの `r p` コンポーネントの前に `s` > `r r p` コンポーネントがあるため、棒グラフが背面に配置されています。
 - `d g p` コンポーネントは、左、右、上、または下の 4 つのどの位置にも配置できる。凡例はグラフの境界線内に配置されます。この例では、凡例によってグラフ自体の横幅が圧縮されています。
 - データ系列のコンポーネント 属性を使用して凡例タイトルを追加する。
 - グラフの下軸のラベルを回転させるために `r b l ph` コンポーネントはこのコンポーネントが影響する `p` コンポーネントで囲まれている。
 - `r ph` コンポーネントにより、このコンポーネントを囲む系列の各データポイントの詳細を示すロールオーバーツールのヒントを有効にできる。

関連リンク

Visualforce Charting のしくみ

更新されたデータによるグラフの更新

新規または更新されたデータでグラフを再描画するには、[コンポーネントを使用する](#)か、JavaScript Remoting と独自の JavaScript コードを使用します。

- u p@ では、Visualforce のみを使用してグラフを更新します。JavaScript Remoting では、JavaScript コードを作成する必要がありますが、柔軟性が向上し、より円滑に更新できます。

<apex:actionSupport> を使用したグラフデータの更新

ユーザのアクションに応じて Visualforce グラフを更新するには、[コンポーネントを、S >](#)
グラフのデータに影響する Visualforce ユーザインターフェース要素に追加します。

JavaScript Remoting を使用したグラフデータの更新

Visualforce グラフは、カスタム JavaScript を使用して、定期的に、またはユーザのアクションに応じて更新します。JavaScript コードを使用すると、複雑なユーザ活動やタイマーアイベントに対応できます。また、JavaScript Remotingを使用して、新しいグラフデータを必要なときにいつでも取得できます。

`<apex:actionSupport>` を使用したグラフデータの更新

ユーザのアクションに応じて Visualforce グラフを更新するには、[コンポーネントを、](#) グラフのデータに影響する Visualforce ユーザインターフェース要素に追加します。

次のマークアップでは、グラフの横にあるメニューから新しい年を選択して更新可能な円グラフを表示します。

```

e   x t      r     <     /a    >
e   xt te    nou u  p   <     /a    Pa>

e   x      ro m p f   <     a    >
e   e ex t t e l  p t l cu e < > zr a s L i sv a' ! Ya "a $ 1 = " >
e   e ex t t len{ p o e l u p r as kh oi s p a' ! Ya Oá s" > /
e   xt ne te  t t m o  a l s s a p i a D S s á s " ya > /
e   e ex t t l p c   <     /as L i s >
e   xt t tn ot p d dn < o ma ia S s á p i a D S s á s " ya
e   x      ro m p f   <     /a    >

ex   e tB l g a p p o c <c k/a a >
ex   B l go p p c < k/a a >
g   p p < /a >

```

このマークアップでは、グラフの属性を Visualforce 式に設定することで、グラフコンポーネントをそのデータソースに添付します。式は、コントローラメソッドをユールし、コントローラメソッドからデータが返されます。グラフは、`h add` という属性を持つ `ua b i p <` でラップされます。

`ro m p f` コンポーネントは、グラフの更新が必要な場合に、ページのコントローラに新しい年を返送するため使用されます。`p c` タグは、グラフで使用可能な年を表示し、子 `o upc` タグは、メニューが変わると常にフォームを送信します。グラフの `u elp d tp <` のである `h ic Pa i` は、`eo upm o r drpp a` 属性で、ページ全体を再読み込みするのではなく、更新をグラフのみに制限するために使用されます。さらに、`o pa <` コンポーネントは、グラフの更新中に状況メッセージを提供します。短いテキストメッセージをアニメーショングラフィックやテキスト効果で簡単に置き換えることができます。

PieChartRemoteController

このページのコントローラは、「[単純なグラフ作成の例](#)」(ページ 197)で使用されている円グラフコントローラを拡張したものです。

```

e bli e t e t p c tre n o k m or l d r C ss á P Ra
t te blne t { r e t h o Tr d dh // y a Ya a
t et blne t { r p c g r dh i is Ya a
t et r ru r r e dh lh r ir dh ( Ya =æ 2N ) Ya =ä 3 0 '
} s ;
t et e rte l dolt ø e r edex Y / A ns vati a áf a p c a , < as L i s >
t et bli e t ptc e l ø et o át g f p nai < o s Oi p > Ya Oá s ( )
t et t e l n e o wa te p t i ts l s n Oi o > c y s a p i s s S Oi > ; ( )
e e ewn dd t l n o c y spa 2 a ( 2 s Oi 1 ( 3' 0 , ' 3' 0 ; ' )
e e ewn dd t l n o c y spa 2 a l 2 s 2 Oi 1 ( ' 0 , ' 0 ; ' )
e e ewn dd t l n o c y spa 2 a ( 2 s Oi 11 ( ' 0 1 , ' 0 ; ' )
e e ewn dd t l n o c y spa 2 a ( 2 s Oi 1 ( ' 0 0 ' ' 0 0 ; ' )
} e n r ru r y s; a
t e bie t e pcd tg g l W < s i D a aDi P a ( a )
e e e xli ro nrt f c o ve e pt i / s cer at r ip es s g o ar mss af a ap sp s
t e n reut tre ne e k m er t d r C tgi Pe tra r h r dh Dia P a ( a s . Ya ; a )
} t Ano m o c R i

```

```

t t bl te e e pte ede teg it st d ema<8 ir D g a a R Di P a (a S y a )
t n m nlg e e t c d tne t R/m i p sallh sh c a a s i . Dia P a (y ; a )
} e n ret reut tre ne e mportd r Cte rgi P Ra r . Dia P a (y ; a )

t tere dn rg ro // P a "a a "a
t tr te e p tee cde ten g i t re ie aks ir D g a a Dia P a (a S y a )
t e e t tde g wd te e u w i s i g a a a a w i s i D a (a S y a )
e e e ne b htm r T b tla u stu ds et nos r a s a g p i a s , sa
e t e t r n wh o do te to te ip lu c nc ar h igp a a sa
t e elen edd t d n g J a aWa( i D a(a ' a3 ' ; 0))
t e elen edd t e d g b a aWa( i D a(a1 ' ' ; 5))
t e elen edd t d g r a aWa( i D a(a1 ' a , ' ; 0))
t e elen edd t d g s r a aWa( p D D a(a ' , ' ; 0))
t e elen edd t d g a r a aWa( p D D a(a 'ya , ' ; 0))
t e elen edd t d n g uJ a aWa( i D a(a ' , ' ; 5))
}

1
t e elen edd t d n g J s a aWa( i D a(a ' a , ' ; 0))
t e elen edd t e d g b a aWa( i D a(a1 ' 3 ' ; 5))
t e elen edd t d g r a aWa( i D a(a ' a3 ' ; 0))
t e elen edd t d g s r a aWa( p i D a(a ' , ' ; 4; 0))
t e elen edd t d g a r a aWa( i D a(a 'ya , ' ; 5))
t e elen edd t d n g uJ a aWa( i D a(a1 ' , ' ; 0))
}

tn r r u d ; a a

r r l p W // a ss a
e blie l { t p c d g i W ss a D a a

t eln er te p dg m lg i i s a ; s ;
e et nb{ t e ter pdg t I lg i a a ; s ;

t e elen ne n hm m i s . a (a S a , a a)
t e t d hd i s . a a ; a a
}
}

```

このコントローラは、Visualforce グラフへのデータ提供に 2 つの異なる方法をサポートしています。

Visualforce 式 `p` を使用してインスタンスマソッド をコールする
Anon me et e ct 静的メソッド を JavaScript メソッドから回転して JavaScript Remoting を使用する

関連リンク

[JavaScript Remoting を使用したグラフデータの更新](#)
[コントローラメソッドを使用したグラフデータの提供](#)

`apex:actionSupport`
`apex:actionStatus`

JavaScript Remoting を使用したグラフデータの更新

Visualforce グラフは、カスタム JavaScript を使用して、定期的に、またはユーザのアクションに応じて更新します。JavaScript コードを使用すると、複雑なユーザ活動やタイマーイベントに対応できます。また、JavaScript Remoting を使用して、新しいグラフデータを必要なときにいつでも取得できます。

次のマークアップでは、グラフの横にあるメニューから新しい年を選択して更新可能な円グラフを表示します。

ten e gor pl ect tne n a moralà r C =i P Ra " >
 etne u ot ifc r{ killb i c iv D k a (a a a)
 er er atde o ut mn edg B v m d Eay =ar I hl u y (Y va .) a
 eet tlu ro er no ffc ngtr giamo a o . d k a áv. i ()
 eet t e Ano ent eot die ne oem b h r dt gio R Ra . R Di P a a ,'
 t n eue ofr ma y a
 ee tnt e tfue r lu tr ntli s r saro A&s rx && (s . s == y)
 e t e n to m w g r oC h R i i P a ;()
 e e e e t e n e e x tf { n p soi cv (p . y == ' i)
 et atde o ut mne dgEBet m deE r no m eo rrnm e e p Etry " n e R s M s m g i .H) L =v
 +
 e e berew r h+ ' > / v ' . ;
 e e l s
 et atde o ut mne dgEBet m deE r no m eo rrnm e e p Etry " n e R s M s m g i .H) L =v
 }
 e et } r uc :p ;)
 }
 en e neu e dr rfC o mf h ki s R a ()
 ttee rt n eil et mde o ut mn dgEBet m adeE = uI l . p y \$ D ' sá s y)
 tee ten ulnn m Er n l dsM sTg i .H L = " i a . . ;
 te t t tr n t mle we {ot fcuiv Dm R a (a i s(sa)
 e t ntr n rütn u od fc i (a a)
 ttee ten ulnn m Er s M sTg i .H L = ; ''
 tthe t n ul m E s(sa)
 }
 t r c p < s i/ >
 x te tB l gotp lp cr < kca a i = s >
 ex e tB legn p po cel & ktha ro na r fet M tJ # i s e nt gja v aiS R i "
 ex et t r wh g h hde etead h at i = " 3 " 50 = " 4 50 = a a iv D a " aa
 e en te nm to m g & n ddC rhair i i P " ai = " " >
 e e e te r qt p lde d< el bain ldss fan a = " " a a a = " " a > /
 e ex n tl dg top o r p g ah si = i " > /
 ex t r ph < /a >
 d
 ee t ete l d e ncerne de < trs kti = " o mf h " a k a" s R a;" >
 t en o o l u pt n 2 < o 2bi v p # " 3 10 > < 3 0 i/ >
 t en o o l u pt n 2 < o 2bi 2 v p # " 10 > < 0 i/ >
 t en o o l u pt n 2 < o 2bi v p # " 10 > < 0 i/ >
 ee t l c < s / >
 n tt d p l n < sp ia = " sp sá s < y a s / >a
 en e et de r no pm o rr son sp Eia = " R sp s s' > s / >a
 d < i / >
 ex e tB legn p po c <c k/a a s >
 ex B l go p p c < k/a a >
 q p p < /a >

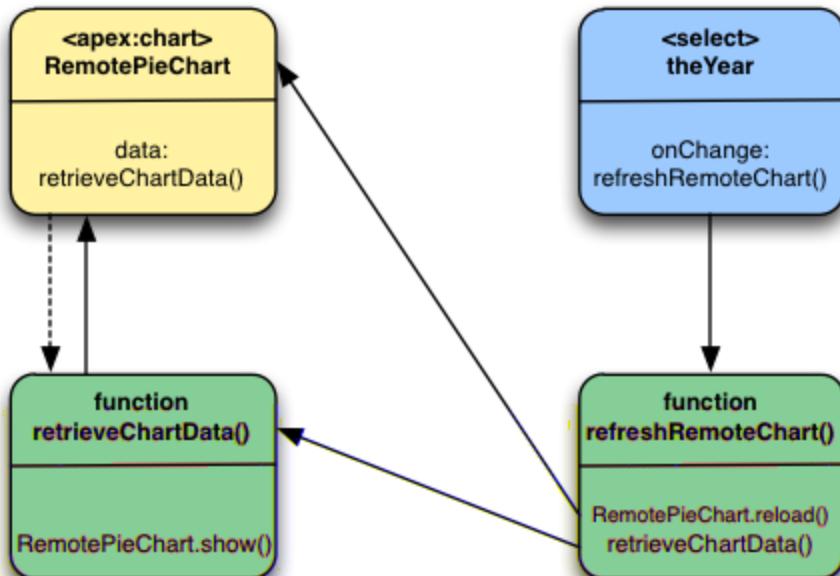
このマークアップでは、グラフの `data` 属性を JavaScript 関数 `retrieveChartData()` の名前に設定することで、グラフコンポーネントをデータソースに添付します。データはこの関数から返されます。関数の名前は文字列として指定します。

静的 HTML メニューには、グラフで使用可能な年が表示されます。メニューは、どの種類のフォーム要素とも関連付けられておらず、値がコントローラに直接返送されることはありません。代わりに、メニューの `onChange` 属性が、メニューが変更されたときには常に JavaScript 関数 `refreshRemoteChart()` をコールします。さらに 2 つの静的 HTML 要素として、ID を持つ `<p>` タグが 2 つあります。`<a>` タグは、ページ読み込み時は空で、JavaScript 経由で更新されると必要に応じて状況とエラーメッセージを表示します。

Visualforce マークアップの前にある 2 つの JavaScript 関数は、Visualforce グラフと、データを提供するコントローラメソッドをつなぎます。関数とグラフコンポーネントの間には 3 つのリンクがあります。

1. グラフコンポーネントの `data` 属性は、最初の JavaScript 関数の名前である「`retrieveChartData`」に設定されます。これにより、グラフコンポーネントに、その JavaScript 関数を使用してデータを読み込むように指示します。グラフコンポーネントは、グラフが最初に作成されてデータが初期読み込みされるときに 1 回だけ直接呼び出します。
2. 再読み込みは、2 つ目の JavaScript 関数 `refreshRemoteChart()` がコールされると行われます。これが、メニューからの 2 つ目のリンクです。年メニューが変更されると、`refreshRemoteChart()` が呼び出され、そこから `retrieveChartData()` 関数が再呼び出しされて新しいデータのセットが読み込まれます。
3. `getRemotePieChart()` が `refreshRemoteChart()` を呼び出すと、コールバックとして匿名関数が提供され、この匿名関数が `refreshRemoteChart()` をコールしてグラフを更新します。グラフ自体は `data` 属性を設定して指定した `getRemotePieChart()` であり、`getRemotePieChart()` は、作成された Visualforce グラフで使用可能な JavaScript 関数で、新しいデータを受け入れてグラフを再描画します。

この図は、ページのさまざまなコンポーネント間のリンクを示しています。



グラフの初期読み込みシーケンスは単純です。リクエストされたというリクエストが受けられると、JavaScript がコールします。こうしてグラフが表示されます。

更新はちょっと複雑です。新しい年がメニューから選択されると、メニューのイベントが起動され、匿名コールバック関数をコールします。次に匿名コールバック関数をコールし、が新しいデータを返すと、で提供されるコールバック経由でリクエストが(ajax)をコールします。こうしてグラフが更新されます。

その他に、いくつか注意点があります。

・`return` は属性を使用して表示するデータが描う前にグラフが表示されないようにします。グラフデータが読み込まれたら、匿名コールバック関数がをコールしてグラフを表示します。これとによって、ajax を使って表示する場合よりもはるかに滑らかなグラフアニメーションを表示できます。

・`onbeforeupdate` 関数は、HTML の「loading...」メッセージに設定し、データが返されてグラフが更新されたら、匿名コールバック関数が空の文字列に設定してメッセージを非表示にします。基本的に同じ効果を得るために `onprogress` を使用する場合よりも作業が若干多くなります。同じ手法を使用して「busy(処理中)」アニメーションやグラフィックを簡単に表示できます。

PieChartRemoteController

このページのコントローラは、[単純なグラフ作成の例](#)(ページ 197)で使用されている円グラフコントローラを拡張したものです。

```

e blu et e tpc tre f o m or l d r C ss a P Ra
t te blu et h b Tr d dh // y a a
t blu { r p c g r dh i s Ya a
t et r r r e dh lh r ir dh ( Ya =@ 2N ) Ya =@ 3@ '
t entr ru r r dh ; a
}
}

s ;

t t blu et e t ptc el at o at g p nari< o S Oi p > Ya Oa s ( )
e e ewn dd t l n o c y spa 2 a( 2 S Oi 1 ( 3@ , ' 3@ ; ') )
e e ewn dd t l n o c y spa 2 a( 2 S Oi 1 ( ' @ , ' 0 ; ') )
e e ewn dd t l n o c y spa 2 a( 2 S Oi 11 ( ' @1 , ' 0 ; ') )
e en r ru r o c y spa 2 a( 2 S Oi 1 ( ' 0 @ , ' 0 @ ) )

t e ble t e pcd tg g L Ws i D a aDi P a(a )
e e ex h ro nrt f c o Ve e pt i/s cer at rmp esp $ g o ar mss af a ap sp s
e n ret reut tre ne e m er th d r C tgi Pe tRa r h r dh Dia P a(as . Ya ; a )
}

t Ano m o c R i
t blu te e e pcte ede teg it st d ema k s i r D g a a R Di P a(a s y a )
t n @ m nl g e t c d t re t R/m i p sal l h sh c a a s i Dia P a(a y ; a )
e n ret reut tre ne e m er th d r C tgi P Ra r . Dia P a(a y ; a )

t tere din rg ro #/ P a "a a "a

```

```

t t r t e e p t e e c d e t e n g i t r g l e a i s i r D q a a Dia P a (a S y a )
t e e t t d e g w d t e e e W < $ i B g a a a a W i s i D P a (a ) a a
e e r { b f q 2 i (y a s l a ( 3 0 '))
e e e n e b e h t m r T b e t l a u e t u s s t n o s r a s a g p i a s ,
e t e t e t r n w h o d o t e t c d t e p l u / t c n c a x h i g p a a a
t e e d l e n e d d t d n g J a. aW a ( i D a (a ' a3 ' ; 0))
t e e d l e n e d d t e d g b a. aW a ( i D a (a1 ' , ' ; 5))
t e e d l e n e d d t d g r a. aW a ( i D a (a1 ' a , ' ; 0))
t e e d l e n e d d t d g y r a. aW a ( p 2 D a (a ' , ' ; 0))
t e e d l e n e d d t d g a. aW a ( 2 D a (a ' ya , ' ; 0))
t e e d l e n e d d t d n g u J a. aW a ( i D a (a ' , ' ; 5))
}

l s
t e e d l e n e d d t d n g J a. aW a ( 2 D a (a ' a , ' ; 0))
t e e d l e n e d d t e d g b a. aW a ( i D a (a1 ' 3 ' ; 5))
t e e d l e n e d d t d g r a. aW a ( i D a (a ' a3 ' ; 0))
t e e d l e n e d d t d g y r a. aW a ( p i D a (a ' , ' ; 4; 0))
t e e d l e n e d d t d g a. aW a ( i D a (a ' ya , ' ; 5))
t e e d l e n e d d t d n g u J a. aW a ( i D a (a1 ' , ' ; 0))
}

tn r r u d ; a a
}

e b l u e r r l p c d p g w // a w s s i s s a a a
e b l u e l { t p c d p g w i w s s i s s a a a
t e e t n b l u e r t e p d g m y i i s a ; s ;
e e t n b l u e t e t e p d g t I y i i s a a ; s ;
t e e e b l u t t d p c n g n e r e t n g t W i r d I a (a S a ,
t e e e n e n h m m i s . a ; a
t t t d h d i s . a a ; a a
}

```

このコントローラは、Visualforce グラフへのデータ提供に 2 つの異なる方法をサポートしています。

Visualforce 式でインスタンスマетодを使用する

関連リンク

<apex:actionSupport> を使用したグラフデータの更新

JavaScript 関数を使用したグラフデータの提供

Apex コントローラの JavaScript Remoting

グラフの外観の制御

Visualforce のグラフは、高度なカスタマイズが可能です。さまざまな種類のデータ系列を組み合わせたり、グラフの要素の色を制御したり、マーカーや線などのデザインを制御したりできます。

次の事項をカスタマイズできます。

- ・ データ系列要素の線と塗りつぶしの色
 - ・ 塗りつぶしの色と線の不透明度
 - ・ データポイントのマーカーの形状と色
 - ・ 結線の太さ

- データ要素の強調表示
- 軸の目盛りとグリッド線のスタイル
- 凡例、ラベル、およびロールオーバー表示されるツールチップ形式のアノテーション

この制御が可能なコンポーネントおよび属性の多くについては、「[標準コンポーネントの参照](#)」で説明します。属性とコンポーネントを組み合わせる必要がある効果もあります。これらについてはこのドキュメントでさらに詳しく説明します。

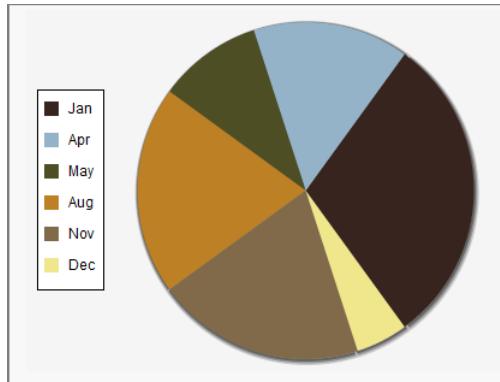
グラフの色

デフォルトでは、視覚的に一貫したダッシュボードを作成できるように、グラフの色は組み込みのレポート作成および分析グラフの色と同じになります。独自の配色を作成する場合、ほとんどのグラフ要素の色をカスタマイズできます。

データ系列要素(棒、円グラフの系列など)を描画する一連の色の定義を指定するには、`c` 属性を使用します。グラフのすべてのデータ系列に使用する色を指定するには、`color c < a` を設定します。データ系列コンポーネントに `c` を設定すると、その系列のみに色が指定されます。

`c` は HTML スタイルの 16 進数の色定義をカンマで区切ったリストの文字列です。たとえば、`A C B A A2 E A #E4B3 E8 #21` です。色は順に使用されます。リストの最後に達すると、再び最初の色から開始されます。

円グラフの系列の色にカスタム配色を使用している例を次に示します。



グラフ全体の背景色を設定するには、`c` 属性を使用します。

`c` は `color` 属性を除くすべてのデータ系列コンポーネントに使用できます。`c` の詳細および他のグラフ要素の色を設定するオプションについては、特定のデータ系列コンポーネントのセクションで説明します。

グラフのレイアウトとアノテーション

わかりやすいグラフを作成するには、凡例、有意義な軸範囲とラベル、データ要素のヒントまたはラベルを追加します。

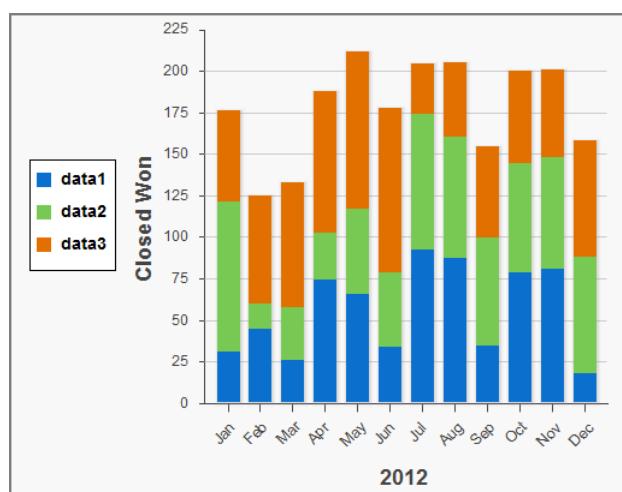
デフォルトでは、すべてのグラフに凡例が表示されます。デフォルトの凡例を非表示にするには、``graph f`` に設定します。凡例の配置と凡例項目の間隔を制御するには、グラフに `graph p` コンポーネントを追加します。`p` 属性を使用して、グラフの四辺の境界のいずれかに凡例を配置します。凡例で使用するテキストスタイルを制御するには、`tf` 属性を使用します。`tf` 属性は、CSS スタイルの短縮フォントプロパティを指定する文字列です。たとえば、`graph p o 1 p a f font="bold 24px 'Helvetica'"` です。

軸の目盛りとラベルを適切に設定しているかどうかは、理解不能な誤解を招くグラフになっているか、明確で説得力のあるグラフになっているかの違いとして現れます。デフォルトで、`p ur m p c a` コンポーネントには `f` 属性で設定したデータ項目に基づいて自動的に目盛りが設定されます。自動目盛り設定を使用すると、グラフ上にすべてのデータが収まりますが、開始値または終了値が有意義な数値ではない場合があります。自動目盛り設定を上書きするには、`m m` 属性と `m i` 属性を使用します。刻みマークの間隔を設定するには、`s` 属性を使用します。この属性は、軸の両端間のステップ数を指定する整数です。測定値と目盛りを比べやすくするためにグラフに線や網掛けを追加するには、`hr d g` 属性、`rd` 属性や `ia` 属性を使用します。

軸とデータ系列にはグラフラベルを適用できます。`bxl ph` が `p L` の子の場合、ラベル `sa` は軸の外側に描画されます。`b l ph` がデータ系列コンポーネントの子の場合、ラベルはグラフのデータ要素の上か近くに描画されます。ラベルのテキストを設定するには、`f` 属性を使用します。ラベルを描画する場所を設定するには、`p` 属性を使用します。グラフ上に收まるようにラベルのテキストを調整するには、`r o` 属性や `i` 属性を使用します。

 メモ: `b l ph` が `p L` の子の場合、ラベル `sa` は軸の外側に描画されます。`b l ph` がデータ系列コンポーネントの子の場合、ラベルはグラフのデータ要素の上か近くに描画されます。ラベルのテキストを設定するには、`f` 属性を使用します。ラベルを描画する場所を設定するには、`p` 属性を使用します。グラフ上に收まるようにラベルのテキストを調整するには、`r o` 属性や `i` 属性を使用します。

次のサンプルグラフでは、これらのコンポーネントと属性を数多く使用して、見た目にわかりやすいデザインを作成しています。



```

x xet e      p t ure m n pote to à ei sa  y="N lfiode"   à Cir d=g r "u i =W s " " i ="
x te ld d t fd d n      2 ixm u B"mle t an am an à à 22 zd=" " p i a h" " $5 s=" " 8 s ia S"
x te r bl ph      < a L a a > /
x xet e t p t ro g p Coe oabi sepo yHeld a "yf à i ="
x te et b l ph o      < a L a a 1=à 3 " $ / 
x x p      < /a i sa>
x e e b rt t ne front o< r à aS à i i a à f i " ai sa" "
x e n ld t m ld d t d te à d =t "2 à à à luc a à a à 3 à à a=" " > /
t r ph      < /a a

```

棒グラフ

棒グラフは、Visualforceで使用できる線形データ系列グラフの1つです。線形系列グラフは、標準の長方形グリッドに対してプロットされるグラフです。

線形系列の各データ要素は、`x, Y`座標で記述されます。グリッド上に座標を描画する方法はデータ系列によって定義されます。`r p` グラフには、原点軸から`X, Y`座標まで伸びる棒が描画されます。原点軸が左軸(Y)なのか、または下軸(X)のかは、`o` 属性により決まります。`a` グラフの左辺を棒の始点にする場合は`tr n pron t zor ro alh à ss i` に設定し、棒がグラフの底辺から上に伸びる縦棒グラフの場合は`t ne front o< r à aS à i i a à f i " ai sa" "` に設定します。`i " à`

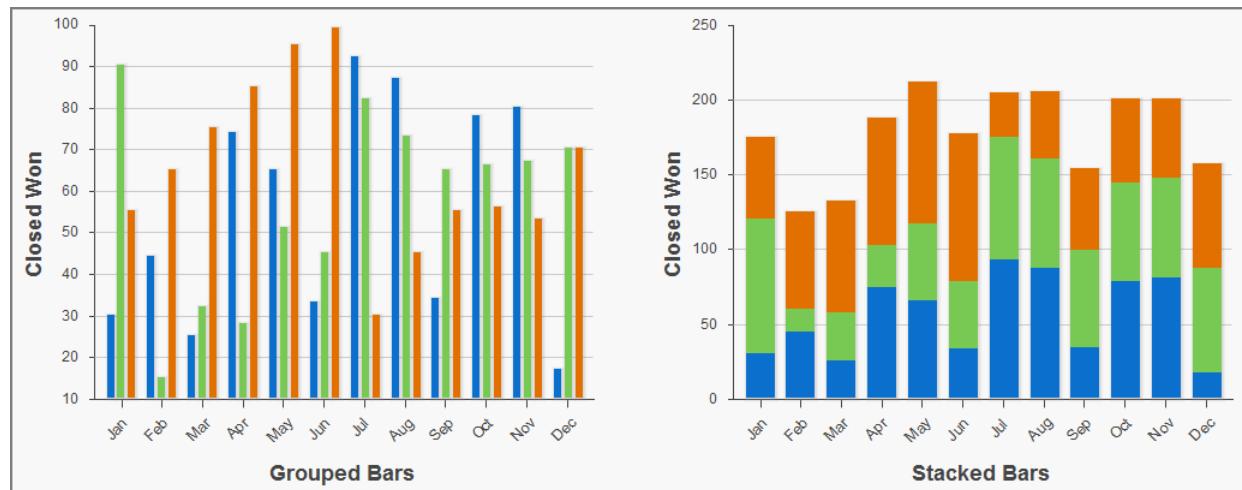
各棒の区間に複数のデータポイントをプロットするには、`b`つの`r p` タグ内に棒をグループ化または積み上げます。1つのグラフに複数の`r p` タグを含めると、重なり合って描画されるため、最後のデータ系列以外はすべて覆い隠れてしまいます。縦棒グラフを作成するには、グループ化して、または積み上げて表示するすべての項目を`属性に追加します。xy`

```

e e b rt t ne front o< r à aS à i i a à f i " ai sa" "
e n ld m yField="data1",data2,data3" > /

```

デフォルトでは、`r p` のデータ項目はグループ化されてグラフに表示されます。データ項目を相互に積み上げる場合は、`r uc` に設定します。`"`

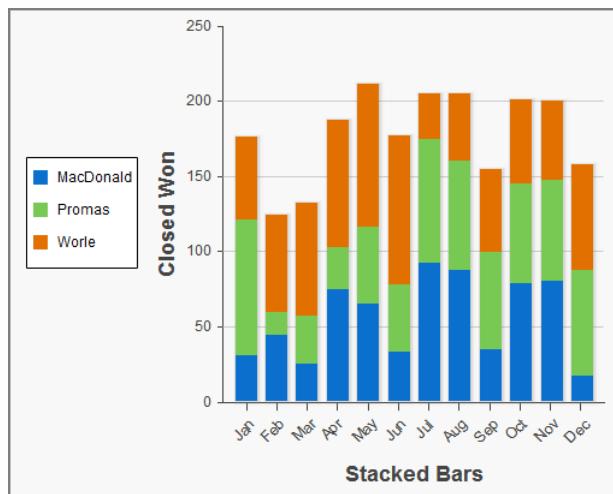


グループ化される各棒の間隔を調整するには、`g` 属性を使用します。グループの間隔を調整するには、`g n dd` 属性と `g` 属性を使用します。

デフォルトでは、積み上げまたはグループ化棒グラフの凡例のタイトルには、`属性の項目の名前が使用されます。上の例では、デフォルトのタイトルは「data1」、「data2」、「data3」になります。凡例によりわか`

りやすいタイトルを付けるには、コンポーネントの属性を使用します。各項目はカンマで区切ります。たとえば、

コンポーネントの属性を使用します。各項目はカンマで区切ります。たとえば、



関連リンク

[グラフの色](#)

[グラフのレイアウトとアノテーション](#)

その他の線形系列グラフ

他の線形データ系列グラフには、面グラフ ()、折れ線グラフ () および散布図 () があります。

および散布図 () があります。

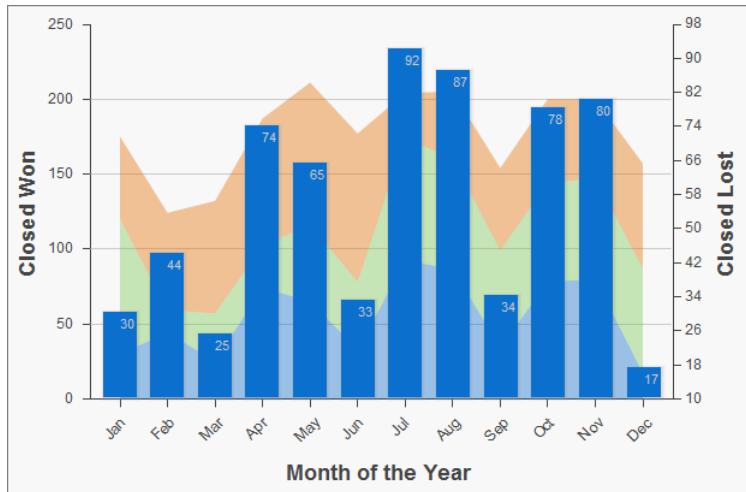
同じグラフで複数の線形データ系列グラフを組み合わせることができますが、わかりやすいグラフを作成するには、次の点に留意してください。

- データ系列グラフは、Visualforce マークアップで定義した順に重ねて描画される。

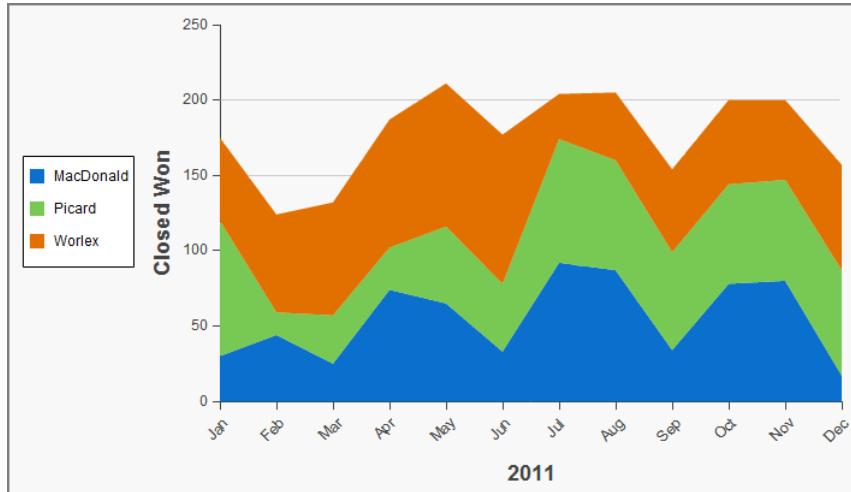
- 通常、グラフは背景に表示する必要があり、透明にできないため、最初に定義する。

コンポーネントは積み上げ棒グラフと似ていますが、個々の棒の代わりに、系列の各点を線で結んで定義される網掛け領域として描画されます。

- には、属性を使用して面グラフを部分的に透明にします。 属性は、0.0 ~ 1.0までの浮動小数点の数値です。0.0は完全に透明、1.0は完全に不透明を表します。棒グラフの系列と組み合わせた面グラフの系列を次に示します。



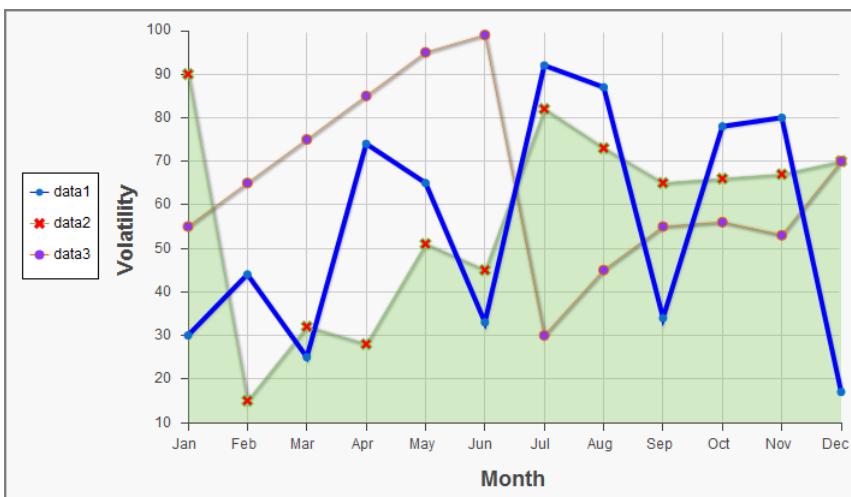
デフォルトでは、面グラフの凡例のタイトルには、属性の項目の名前が使用されます、上の例では、デフォルトのタイトルは「data1」、「data2」、「data3」になります。凡例によりわかりやすいタイトルを付けるには、コンポーネントの `label` 属性を使用します。各項目はカンマで区切ります。たとえば、`label: "Closed Won", value: "#B33"` のように使用します。

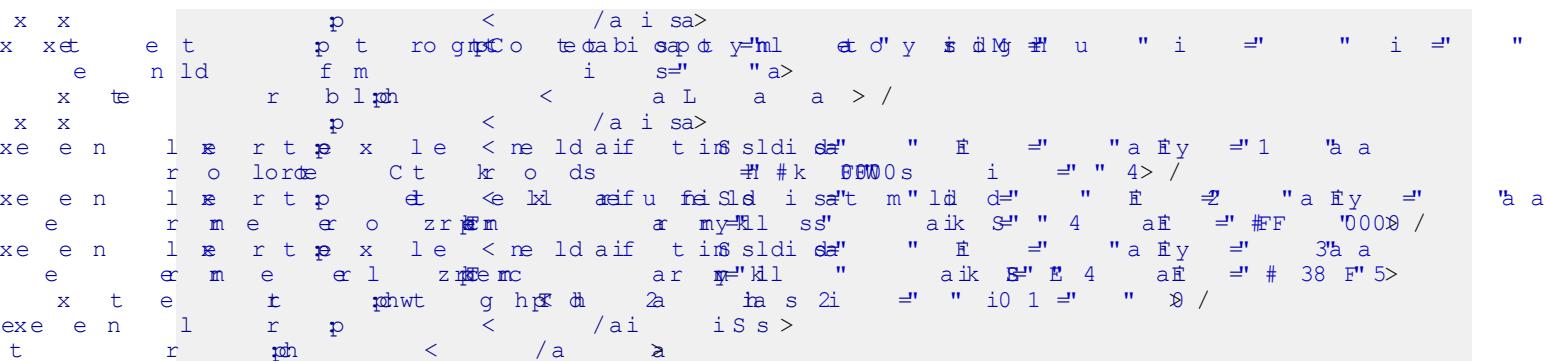


グラフと同様に、
の下の領域を塗りつぶす
ことができます。

グラフでは一連の点を結ぶ線が使用されます。線
グラフとは違い、

グラフは積み上げられません。
グラフが塗りつぶされていない場合は、同じグラフに複数の系列
を表示することができます。折れ線グラフの系列にはデータポイントのマーカーを表示でき、マーカーと結線の
両方の色とサイズを定義できます。3つの系列を組み合わせ、そのうちの1つが塗りつぶされている折れ線グラ
フを次に示します。





関連リンク

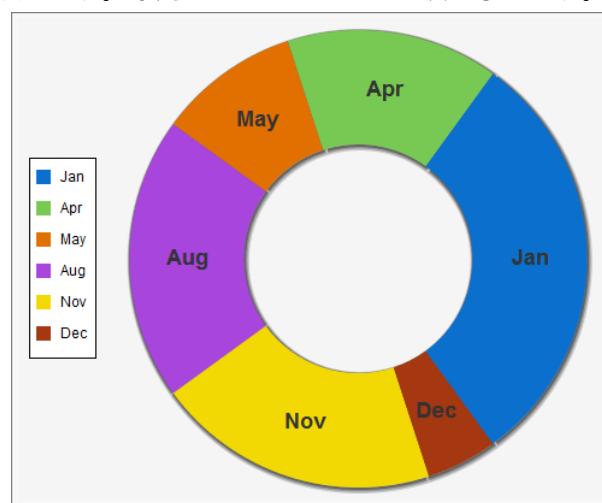
[グラフの色](#)

[グラフのレイアウトとアノテーション](#)

円グラフ

属性とコンポーネントを使用します。

円グラフの代わりにドーナツグラフを作成するには、`rt` 属性を設定します。で、穴の半径のパーセントを表します。単純なドーナツグラフを次に示します。



属性は、0 ~ 100 の整数

re r p p < /ai iss>
t r ph < /a a

関連リンク

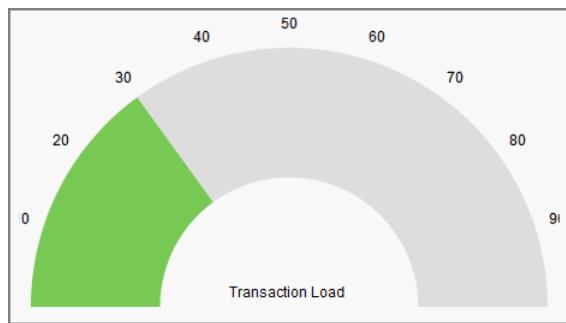
[グラフの色](#)

[グラフのレイアウトとアノテーション](#)

ゲージグラフ

ゲージグラフには、定義された軸または目盛りに対し 1 つの測定値が表示されます。グラフに表示されるのは 1 つの数値ですが、軸およびグラフの色を変えて、その数値の意味を伝えることができます。

値の範囲を定義するには、`range` 属性と `color` 属性を使用します。`isvalid` 属性を使用します。現在の値が適切か不適切かを示すには、`isvalid` 属性と `color` 属性を使用します。測定値が許容範囲内であることを示すグラフを次に示します。



 メモ: ゲージグラフでは、凡例やラベルはサポートされません。

関連リンク

[グラフの色](#)

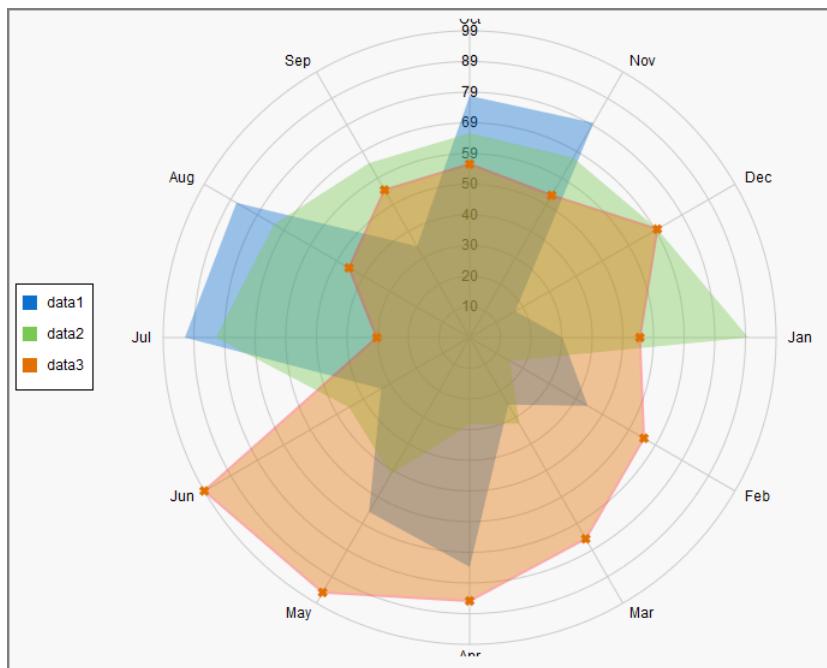
[グラフのレイアウトとアノテーション](#)

レーダーグラフ

レーダーグラフは、折れ線グラフに似ていますが、線グリッドの代わりに円形の軸が使用されます。

マーカーのスタイル、サイズ、色を設定するには、`markerStyle`、`markerSize`、`markerColor` 属性を使用します。結線の色と太さを設定するには、`lineColor`、`lineWidth` 属性を使用します。必要に応じて `fillColor` 属性を設定して系列で囲まれた領域を塗りつぶし、`fillOpacity` 属性を設定して他の系列が表示されるようにこの領域を透明にします。`fillColor` 属性は、0.0 ~ 1.0 までの浮動小数点の数値です。0.0 は完全に透明、1.0 は完全に不透明を表します。

レーダーグラフの例とグラフを作成するマークアップを次に示します。



```

t      t   r   wth g  ke e hd<    etn al h tdaq r  #d3 "5i0} d=" 7 " 00  ="   "   =a a!   'a a
x     n tl e dgt op o l p< a f $ i ="   " > /
x xet pt d l n ps o radi pba y" R i a" a s i =" i a" a
x te r b l ph < a L a a > /
x x   p < /a i sa>
x e e er dr re p ne ld < t m t ad data a S s F t r=" u "ba ph y =" 1 c p'a a s=" "
x e e er dr re p ne ld < t m t ad data a S s F t r=" u "ba ph y =" c p'a a s=" "
x e e er dr re p ne ld < t m t ad data a S s F r=" u "a ph y =" 3'a a s=" "
e   r m te r @ r p d ar hy" kloresW s C o k =F " s cp =" # 33" i a=y " 04> /
t     r ph < /a a

```

関連リンク

[グラフの色](#)

[グラフのレイアウトとアノテーション](#)

第 16 章

Visualforce でのフローの表示

VisualWorkflow では、システム管理者は、ユーザが画面を順に進み、データの収集と更新を行える「フロー」というアプリケーションを構築できます。たとえば、VisualWorkflow を使用して、カスタマーサポートセンターへの電話のスクリプトを作成したり、営業組織用にリアルタイムの見積を生成したりできます。

フローを実行するための標準のユーザインターフェースはカスタマイズできません。Visualforce は、アプリケーションの作成とアプリケーションのデザインのカスタマイズを行うための強力な手段を開発者に提供する、コアテクノロジーです。VisualWorkflow では、この技術を活用して、フローを実行するときにユーザインターフェースをカスタマイズできます。

次のトピックでは、Visualforce ページにフローを埋め込み、設定する方法を説明します。

- [Visualforce ページにフローを埋め込む](#)
- [フローの高度な例](#) >
- [フローの属性の設定](#)
- [フローのユーザインターフェースのカスタマイズ \(ページ 228\)](#)

Visualforce ページにフローを埋め込む

フローのデザインをカスタマイズするか、機能を強化するには、Visualforce ページにコンポーネントとして埋め込むことができます。組織のサイトとポータルでフローが有効になっている場合、Force.com サイト、カスタマーポータル、またはパートナーポータルのユーザにフローを提供できます。



メモ: ユーザは、アクティブなバージョンを含むフローしか実行できません。埋め込んだフローにアクティブなバージョンが含まれていない場合は、エラーメッセージが表示されます。埋め込んだフローにサブフロー要素がある場合、サブフロー要素から参照およびコールされるフローは有効バージョンである必要があります。

フローを Visualforce ページに追加するには、次のように [コンポーネントを使用してフローを埋め込みます。](#)

1. フローの一意の名前を検索します。
 - a. フローリストのページに移動します。[設定] で、[作成] > [ワークフローと承認申請] > [フロー] をクリックします。
 - b. 埋め込むフローの名前をクリックします。
2. 新しい Visualforce ページを定義するか、編集するページを開きます。
3. [コンポーネントを追加します。](#)

m4. 属性をフローの一意の名前に設定します。次に例を示します。

```
te n lew nr g p p < a >
x e : n ew < ul qM i vmi Ua y F N " a > /
g p p < /a >
```



メモ: 管理パッケージのフローである場合、**属性**は `l no u c pu q am` の形式で指定してください。

5. フローを含む Visualforce ページのページセキュリティを設定することで、フローを実行できるユーザを制限します。

フローを含む Visualforce ページがサイトまたはポータルに外部的に配信される場合、Visualforce ページへのアクセスを持つユーザは埋め込まれているフローを実行できます。

カスタム Web タブ、リンク、またはボタンを使用して、フローを含む Visualforce ページを組織内のユーザに提供する場合は、ユーザにそのページへのアクセス権がある必要があります。また、「フローを実行」権限、またはユーザ詳細ページで `l mo c` ユーザ `F` 項目が有効化されている必要があります。

6. フローの完了動作を設定することにより、フロー画面の [完了] をユーザがクリックすると実行される処理を指定します。

フローの変数値の設定

この例では、カスタマーサポートエージェントがケースを作成してモデムの問題をトラブルシューティングできるようにする、単純なフローを作成します。`mp p` コンポーネントを使用してフローを開始するときに変数の値を設定できます。この例では、フローの読み込み時に `r nc` というケース番号変数を初期値の `01212212` に設定するために、次のマークアップを使用します。

```
w te n lew nr g p p < a >
x e : e mdo t & bimM i oov i Thg =a Si " >
w te n lew r f : < b<r nc l au a a v'a 2saNa 2 v 1a" 1 0 " > /
g p p < /a >
```



メモ: インタビューの開始では変数の設定のみを行うことができます。`mp p` タグはフローが開始されるときに一度のみ評価されます。

入力アクセスを許可する変数の設定と、出力アクセスを許可する変数の取得のみ行うことができます。それぞれのフロー変数では、入出力アクセスは次の項目によって制御されます。

- Cloud Flow Designer の **入力 出力種別** **変数項目** [/]
- Metadata API `ln` `u` の `tVI F` 項目および `a is` 項目
- `i so`

入力アクセスまたは出力アクセスを許可しない変数では、変数の設定または取得の試行は無視され、Visualforce ページ、その `g p p` コンポーネント、または Apex クラスのコンパイルが失敗する可能性があります。

また、標準 Visualforce コントローラを活用して変数を設定することもできます。たとえば、Visualforce ページが `c` コントローラを使用している場合、標準コントローラからデータを渡すようにページの機能を強化できます。

```
t n tegndprde por h d etC ea b a a C=" s " a a Sy=" s " a >
w te n lew nr f : e mdo t & bimM i oov i Thg =a Si " >
x e n e e mp mp { b<r nc d aue a a v'a } QaNaNc v a" ! s a s Na " > /
```

```
ew t e n l wo      r   f :      <     / i   v i   >
    g   p   p       <     / a   a
```

finishLocation 属性の設定

モデルのトラブルシューティングの例に基づいて、ユーザがフローの終わりにある[完了]ボタンをクリックしたときに Salesforce ホームページにリダイレクトされるように `foh c` 属性を設定することもできます。

```
t      n  tegndprde por id setC ea b a a C s " a a Sy=" s " a >
te n lewo  rer f :e mdot r< minM iom vti(Tg n =b feh e Si "o i mos h m jh aL ! R FpB ( / /
x      e  n er e mp mp { bce mcl aue aa va } cbaManc "v a" ! s a s Na " > /
ew t e n l wo      r   f :      <     / i   v i   >
    g   p   p       <     / a   a
```

`no foh c` のその他の設定例については、「フローの `foh c` 属性の設定」(ページ 226)を参照してください。

<flow:interview> を使用した高度な例

コンポーネントは、複雑な Visualforce の相互作用を簡単に開発できるようにするために設計されています。ただし、カスタムコントローラを作成してフロー内で利用できる機能も多くあります。カスタムコントローラを使用すると、相互にやりとりできる複数のコンポーネントを含むページを作成できます。独自の Apex 型によって組織内のすべてのフローを個別に参照でき、フロー内の変数にはメンバー変数としてアクセスできます。



メモ:

入力アクセスを許可する変数の設定と、出力アクセスを許可する変数の取得のみ行うことができます。それぞれのフロー変数では、入出力アクセスは次の項目によって制御されます。

- Cloud Flow Designer の [入力/出力種別] 变数項目

Metadata API の [項目および属性] 項目

入力アクセスまたは出力アクセスを許可しない変数では、変数の設定または取得の試行は無視され、

Visualforce ページ、そのコンポーネント、または Apex クラスのコンパイルが失敗する可能性があります。

次の例では、一意の名前「ModemTroubleShooting」を持つフローは、として参照されます。マークアップでは、ページの他の部分でフロー変数の値を表示する方法を示します。

```
ten e gorpldep C dotr minMa eo a Tg at =m em b l p Si s Ci S " a Sy=" s " a >
w t e n l wo  ner f :e mdot < minM ei nfv w Tg r =aw } l mo Si f" i v i =" ! y " > /
x et txe ou e ptle up <Tu a if ro C vt D a":ro c a s ai E y ! s a E "y > /
g   p   p       <     / a   a
```



メモ: 管理パッケージのフローである場合、属性は `l no u c pu q am` の形式で指定する必要があります。

上記のマークアップのコントローラは、次のようになります。

```

t ew ble n l ewo p re Ido t r blinM 空 te.T h y timo . lgf Si y ; s ;
t e blin r t pcg r ro c i is s a 空 ; y
t e blin r t pcg g r ro c i is s a 空 y( )
e A w e l occre bel fe l e/m bsm nev pabl et at s sa g h 空 i ah si s /
e w en l tmo lfun rf ru i g ( yh == ) H i ' ; '
e t l n r weu l mot rf ro G y v . sa a 空 ; y
}

```

カスタムコントローラを使用する場合、フロー構造タのフローの先頭で変数の初期値も設定できます。

`mp p` タグを使用して値を設定する場合、コンストラクタを使用して変数を渡すことは省略可能で、必須ではありません。

次は、コンストラクタのフロー変数の値を設定するカスタムコントローラの例です。

```

e bl le pdotc r blinM do eT h y o ssmaC bl Si vs Si a 空
ew ble n l ewo p re Ido t r blinM 空 te.T h y timo . lgf Si y ; s ;
e t en dot r p blinM do eT h y{oi mC bl Si vs Si a 空 ( )
t e ne u M p h r nc y a2 v ( 'saNa 1 , '3 ' 4 5 6)
ew wnl new te n lfewo r e Ido t y blinM 空 .T h y in . M p Si ( y ; a )
}
t e blin e f e pcg br m} i is s Na s ;
t e blin re [pcg g br m i is s Na ( )
n r weu el mo f br nc y v . saNa ;
}

```

次のコードでは、`l.u` メソッドを使用して、Visualforce コントローラがフロー変数の値にアクセスできるようにすることができます。変数は、Visualforce ページに埋め込まれたフローか、サブフロー要素でコールされる別のフローに含まれている場合があります。変数値は、これらのうちインタビューが現在実行されているフローから返されます。指定された変数がフロー内に見つからない場合、メソッドは `null` を返します。このメソッドは、コンパイル時ではなく実行時にのみ変数の存在を確認します。

次のサンプルでは、`l.u` メソッドを使用して Visualforce ページに埋め込まれたフローからブレッドクラム(ナビゲーション)情報を取得します。そのフローにサブフロー要素が含まれ、参照される各フローにも `varBreadCrumb` 変数が含まれる場合、どのフローでインタビューが実行されているかに関わらず、すべてのフローのブレッドクラムを Visualforce ページから取得できます。

```

h l te {npclc mo l d r p i ss a Sa
n net w d d o fh f /s/ a
ew ble n l wo e pwr e t Ieo l m{ blw r teTp v timo F. } g _ a G a y F y ; s ;
t e blin r R o c g r d r b m C i s a ( )
e n r B rg d r b m C i s a a ;
w { nel mt file n r ru } io m(Fy == ) H i ' ;
e l B r t d r b n m erow t gl moe g e b d i S r d F y b . m V i C a a a ( ' a a ; ')
e n r B r d n r e b mlaC o B m r (d ar :ba == C H ? ' ' a a ; )
}

```

次の表に、フローと Apex との間での、サポートされているデータ型の名前付けの違いを示します。

フロー	Apex
text	String
number	decimal
currency	decimal
date	date、dateTime
Boolean	Boolean

Apex コードのテストを記述する適切な方法として、次に示す `Test Apex` の例を参考して、独自のクラスを作成する単純な例を挙げます。

```
re le p do tc r m i n t e d o e T h g o { s t m a c b l
t e t t e n e g r e e f g c e p g l o f t R a m h M e o o R a T h e r b l P a
e t t e t e n e e n T C g g s p s f Pa( Ra ; )
e dot r m i n t e d o e T h g o e t m r t b l t e n m S i o r l d s r C S i a a s y s =
ewn e dot r m i n t e d o e T h g o m c b l S i V s S i a a s ; ( ) =
t e t e t e n r t h e q n e E m t e o r l a y r C . s s g b s a n ( y s 2 2 2 2 N a 1 ( 1 ) 1 ' 0 ; ' ) }
```

reRender 属性の設定

属性を使用することにより、`reRender` フローを再表示します。

```
w t e n l e o n e r d r e t t x t e t t l e u n r d r f : e n r n r f : e n r n r f :
```

警告: 属性を設定しない場合、フロー内の他の画面に移動するボタンをクリックすると、コンポーネントのみでなく Visualforce ページ全体が更新されます。

フローの finishLocation 属性の設定

コンポーネントを使用して Visualforce ページにフローを埋め込むと、Force.com サイトユーザー、カスタマーポータルユーザ、およびパートナーポータルユーザが、Visual Workflow を使用して作成された動的アプリケーションにアクセスできるようになります。Visualforce で使用可能なその他のフローのカスタマイズに加えて、最後の画面で [完了] をクリックするときに実行される内容を指定することもできます。

次のセクションでは、`finishLocation` 属性を設定する方法を示します。

- 関数を使用した
- 変数を使用した
- コントローラを使用した

の設定

の設定

の設定

**メモ:**

n t n o foh c が指定されない場合は、[完了] をクリックしたユーザはフローの最初の画面に戻ります。

- フローと同じページにレコードを表示するために標準コントローラを使用すると、[完了] をクリックしたユーザは、レコードのない、フローの最初の画面に戻ります。これは、クエリ文字列パラメータがページ URL で維持されないためです。必要に応じて、ユーザをレコードに戻す場合は、n t n o foh c を設定してください。
- Salesforce 組織の外部 URL にフローウェーブをリダイレクトすることはできません。

URLFOR 関数を使用した finishLocation の設定

関数を使用して、次のとおりを実行できます。

- Salesforce ホームページなどの相対 URL にユーザの経路を設定します。

```
ew te n lwo g p p < a >
nr &w: n m on< t {mfi n voi foh &a "ai lms }h m jjh a" L! R F p ( / .s
g p p < /a >
```

- ID を使用して特定のレコードまたは詳細ページにユーザの経路を設定します。たとえば、ID が 001D000000IpE9X である詳細ページにユーザの経路を設定する場合は、次のように指定します。

```
ew te n lwo g p p < a >
nr &w: n m on< t {mfi n voi foh &a "ai l s } i a" L! R F p 1000000X
g p p < /a >
```

についての詳細は、「[関数](#)」(ページ 567)を参照してください。

\$Page 变数を使用した finishLocation の設定

を使用することなく、ユーザが [完了] をクリックすると他の Visualforce ページにリダイレクトされるようにする場合には、foh c を単にリダイレクト先のページの名前に設定します。

```
w te n lwo g p p < a >
nr &w: n m on< t {mfi n voi foh &a "ag l s p i a" ! Pa a "a > /
g p p < /a >
```

についての詳細は、「[グローバル変数](#)」(ページ 541)を参照してください。

コントローラを使用した finishLocation の設定

コントローラを使用したい場合の異なる方法で foh c を設定できます。ページのサンプルコードの一部を次に示します。

```
ten or plw pe n l &o or l&r Ca &" Fy " >
tn t or nhju C o e <1 em > du Tai asg! > pi si s y a.
w te n lwo nr &w: n m on< t {mfi n voi foh &a "aipl s i a" ! a" > /
g p p < /a >
```

コントローラのサンプルコードを次に示します。

```

bli    w t e ip d mo or l d r C i ss a Fy
e e e e b le e n te gr& Af c g i Ra Pa ( )
c e newre e reu e n gr f c Ra (3 ' 00 ')
}
t e blute f p cgBg g i is Pa ( )
n r ru   m p Ci fgh is p 'a Ay_is _a '
e n r xru   m p
}

```

例は3種類あります。

- Ag • g は、場所を定義するための文字列を渡して、新しいページ参照をインスタンス化します。
- Bg • g は、PageReference のように処理される文字列を返します。
- g • g は、PageReference に翻訳された文字列を返します。

 メモ: Salesforce 組織の外部 URL にユーザをリダイレクトすることはできません。

フローのユーザインターフェースのカスタマイズ

Visualforce ページにフローを埋め込んだ後、CSS を使用してカスタムスタイルを適用することによって、実行時のフローのデザインをカスタマイズできます。フローの属性と CSS クラスを併用することによって、ボタンの場所、ボタンのスタイル、背景、画面の表示ラベルのデザインなど、フローの個々の部分をカスタマイズできます。

フローボタンの属性

フローの [次へ] ボタン、[前へ] ボタン、および [完了] ボタンの表示方法を変更するには、次の属性を使用します。

属性	説明
b u o o o c	L フローのユーザインターフェースでのナビゲーションボタンの場所を定義します。選択可能な値は次のとおりです。 t o • p tt b o o • m t b o • h

次に例を示します。

```

e ex g p p < a > L i a "
e ex f :w tt m l & b M at ov i o b o =b Fy " L i a "
e ex g p p < /a >

```

属性	説明
	 <p>メモ: 指定されていない場合、○。□ 値はデフォルトの <code>ih a</code> に設定されます。</p> <p>フロー ナビゲーション ボタンにセットとしてスタイルを割り当てます。CSS クラスではなく、インラインスタイルでのみ使用できます。</p> <p>次に例を示します。</p> <pre> e ex g p p < a > ew te n lwo nr f :w tt m& b M á lv i loro =a cFy ":" Sy" bn er o� edoro d d� rd:r f �od : -p # ; 1 s i ;" > e ex g p p < /a > </pre>

フロー固有の CSS クラス

これらの事前定義済みのフロースタイルクラスは、独自の CSS スタイルで上書きできます。

フロースタイルのクラス	適用先...
FlowContainer	フローを含む <code><div></code> 要素。
FlowPageBlockBttns	フロー ナビゲーション ボタンを含む <code><div></code> 要素。  <p>メモ: システム内の別の場所で適用されたボタンのスタイルによって、フローナビゲーションボタンの CSS スタイルが上書きされないようにするには、CSS スタイルをフローナビゲーションボタンに適用するたびに、このフロースタイルクラスを指定することをお勧めします。</p> <pre> e w { たとえば、& fu e w の代わりに、& fu s { と or n B ou & fu s </pre>
FlowPreviousBttn	[前へ] ボタン。
FlowNextBttn	[次へ] ボタン。
FlowFinishBttn	[完了] ボタン。
FlowText	テキスト項目の表示ラベル。
FlowTextArea	テキストエリア項目の表示ラベル。
FlowNumber	数値項目の表示ラベル。
FlowDate	日付項目の表示ラベル。
FlowCurrency	通貨項目の表示ラベル。
FlowPassword	パスワード項目の表示ラベル。
FlowRadio	ラジオボタン項目の表示ラベル。
FlowDropdown	ドロップダウンリストの表示ラベル。

第 17 章

Visualforce でのテンプレートの使用

Visualforce では、複数の Visualforce ページ間で類似の内容を再利用する方法が複数用意されています。どのメソッドを選択するかは、再利用するテンプレートに必要な柔軟性に応じて異なります。テンプレートメソッドが柔軟なほど、そのメソッドを使用するテンプレートの実装が変更しやすくなります。次のテンプレートメソッドを使用できます（柔軟性の高い順）。

カスタムコンポーネントの定義

メソッドでコードをカプセル化すると、プログラムでそのメソッドを複数回利用できるのと同様に、カスタムコンポーネントで共通のデザインパターンをカプセル化することにより 1 つ以上の Visualforce ページでそのコンポーネントを複数回利用することができます。カスタムコンポーネントの定義は最も柔軟なテンプレートメソッドです。これは、有効な Visualforce タグであればどれでも含めることができ、制限なしでどの Visualforce ページにもインポートできるためです。ただし、カスタムコンポーネントは、再利用可能な Visualforce ページの定義には使用しないでください。Visualforce ページ全体の内容を再利用する場合、他の 2 つのテンプレートメソッドのいずれかを選択してください。

<apex:composition> を使用したテンプレートの定義

基本テンプレートを定義して、テンプレートの一部を実装ごとに変更できるようにする場合、

□ `om p <コンポーネント>` を使用します。このテンプレートメソッドは、ページの全体的な構造を維持し、個々のページの内容を変える場合に適しています。例として、同じページレイアウトでさまざまな記事を表示する必要があるニュース記事用の Web サイトなどがあります。

この技法によって、コントローラが返す PageReference からテンプレートを定義することもできます。

<apex:include> を使用する既存ページの参照

Visualforce ページの内容全体を別のページに挿入する場合、`du p c` コンポーネントを使用します。このテンプレートメソッドは、複数の領域で同じ内容を複製する場合に適しています。例として、Web サイトのどのページにも表示されるフィードバックフォームなどがあります。

□ `er x p t および om s p > <で作成したテンプレートは、すでに存在する Visualforce ページを参照する場合にのみ使用する必要があります。1 つのコンポーネントセットのみを複製する必要がある場合は、カスタムコンポーネントを使用します。`

<apex:composition> を使用したテンプレートの定義

□ `om p < を使用して定義したすべてのテンプレートには、1 つ以上の子の r p タグが必要です。r p タグは、そのテンプレートをインポートするページに対し、そのセクションで定義が必要であることを示します。 om p < を使用してテンプレートをインポートする Visualforce ペー`

次に、`apex:composition` タグでは、`apex:repeat` タグを使用してテンプレートの各セクションのコンテンツを指定する必要があります。

スケルトンテンプレートを作成すると、それ以降の Visualforce ページが同じ標準構造内で異なるコンテンツを実装できます。そのためには、`apex:repeat` タグを使用してテンプレートページを作成します。

次の例は、 `apex:repeat` タグおよび `apex:repeat` タグを使用してスケルトンテンプレートを実装する方法を示します。

最初に、 `apex:repeat` というコントローラを使用する `myFormComposition` という空のページを作成します。

```
 apex:repeat>
    <apex:repeat>
        <apex:repeat>
```

ページを保存すると、 `apex:repeat` の作成を要求するメッセージが表示されます。次のコードを使用して、そのカスタムコントローラを定義します。

```
public class myFormComposition {
    public String name {get; set;}
    public String address {get; set;}
    public String city {get; set;}
    public String state {get; set;}
    public String zip {get; set;}
    public String phone {get; set;}
    public String email {get; set;}
    public String notes {get; set;}
    public String id {get; set;}
}
```

```

e   e blB tn  e@ov  p c{  gnor      h     g           a   S   Gi    ( )
e   ee w  tru    nor      h     g           s   G   i   ;
}

```

次に、スケルトンテンプレートを作成します。

```

x   t en      gorpl@ pc  exa <om o a lpm E   #   $ i       a" >
x   ro m p f   < a ss sa > /
x   xt te     eou uebt@ nl  y<@  ouE le nm rva : a" m@l y   a" ="
x   et n tx    e eu n pd e {p m@l e @l áu } iind=" @a "v   a" !   @a " > /
x   ex tne    en r p m g   < a s   =a "a > /
x   ex tne    en r p m l m  < a s   =a " a > /
x   ex tne    en r p m l m  < a s   =a " a > /
x   te e      t n rhp T r g h<g >h sa ' p y   i < ?>/
x   tit oBndm@ pno  @d e a l eu a tt d i Bn=" @o "k   a" v   s a ="
x   ro m p f   < /a a >
x   g p p   < /a a >

```

2つの `age` 項目が `age` と `meal` のコンテンツを必要としています。これらの項目のマークアップは、この構成テンプレートをコールするすべてのページに定義されます。

次に、`myFullForm` というページを作成し、そこで `age` と `meal` のマークアップを定義します。

```

x   t en      gorpl@ pc  exa <om o a lpm E   #   $ i       a" >
x   ro m p f   < a ss sa > /
x   xt te     eou uebt@ nl  y<@  ouE le nm rva : a" Fy   $ i " >
x   et n tx    e eu n pd e {p m@l e @l áu } iind=" @a "v   a" !   @a " > /
x   ex n d    p f   < /a i >
x   ex e en d  p m@l m   < a i   =a " a >
x   xt te     eou uebt@ nl  y<@  ouE Lro e v@dm a" ro : l mylfv ia   a" ="
x   et n tx    e eu n pd e {p m@l e @l áu } iind=" @a "v   a" !   @a " > /
x   ex n d    p f   < /a i >
x   t te     eou uebt@ nl  y<@  ouE Lro   @fro a" ro   lory @d cia   " ="
x   et n tx    e eu n pd e @fro Td cel áu   lorj@ @d c   "v   a" !   " > /
x   ex t      a om p   p   < /a   $ i >
x   ex t tx    eeu ut  p d e @p {g <T r g@aw dt   n@r H@ h@ i g@le ue nll =b ls } iadi   "v   H a" !
x   e e o i  eo } g@d r k l@e Y @f o@i @a @p @n s@o@ l@c   "v   a" !   " > /
x   g p p   < /a a >

```

このマークアップでは、次の点に留意してください。

- `myFullForm` を保存すると、前に定義した `age` と `meal` のマークアップと [保存] ボタンが表示されます。
- 構成ページには `age` 項目および `meal` 項目が必要であるため、`myFullForm` はこれらをテキスト入力項目として定義します。項目がページに表示されている順序は重要ではありません。`myFormComposition` は `age` 項目が常に `meal` 項目よりも前に表示されるように指定します。
- 一致する `age` 項目がない場合でも、`name` 項目はインポートされます。
- 項目のコントローラコードが存在する場合でも、`color` 項目は無視されます。これは、構成テンプレートが `color` という名前の項目を必要としないためです。
- `age` および `meal` 項目は、テキスト入力である必要はありません。`age` と `meal` のマークアップ内にコンポーネントを含むことができます。

この例では、`<apex:include>` タグで有効な Visualforce を使用する方法の例として、`newPage` という新規 Visualforce ページを作成して次のマークアップを使用します。

```

<apex:page controller="NewPageController">
    <apex:include page="oldPage"/>
</apex:page>

```

構成テンプレートで必要なのは、`<apex:include page="oldPage"/>` タグが存在することだけです。この例では、`age` がテキストとして定義されています。

動的テンプレート

PageReference を介してテンプレートを割り当てるには、動的テンプレートを使用します。テンプレート名は、使用するテンプレートが含まれる PageReference を返すコントローラメソッドに割り当てられます。

たとえば、スケルトンテンプレートを定義する `myAppliedTemplate` というページを作成します。

```

<apex:page controller="MyAppliedTemplateController">
    <apex:applyTemplate templateName="myAppliedTemplate"/>
</apex:page>

```

次に、このページへの参照を返すメソッドを使用する `getTemplate()` というコントローラを作成します。

```

public String getTemplate() {
    return 'myAppliedTemplate';
}

```

最後に、このコントローラと動的テンプレートを実装する `MyAppliedTemplateController` というページを作成します。

```

public class MyAppliedTemplateController {
    public String getTemplate() {
        return 'myAppliedTemplate';
    }
}

```

<apex:include> を使用する既存ページの参照

何も変更を行わずに他のページのコンテンツ全体を複製する場合は、`<apex:include page="oldPage"/>` タグを使用します。この技術を使用すると、複数の場所で同じように使用する既存のマークアップを参照することができます。



メモ: コンポーネントの複製のみを行う場合は、`<apex:include>` を使用しないでください。コードのセグメントを再利用可能にするには、**カスタムコンポーネント**の方が適しています。

たとえば、ユーザの名前を取得して表示するフォームを作成するとします。最初に、`fmf` というページを作成して、再利用可能なフォームを表し、`fmfmpf` というコントローラを使用します。

```
t en e t goer plex l< m à mpæ p = a a' >
g p p < /a à
```

`< fm f mpf >` が存在しないというメッセージが表示されたら、次のコードを使用してカスタムコントローラを定義します。

```
e tbli e tl ex p clc m l mpf i p ss a a a
e n n r g m is sa ; a G i = s a
B ee w dt nør h gl f sa ; a
e e e e ble en { grc f c i Ra v a ( )
ee w t stor h g u s G i = ;
} n n ru lu ;
e hdi e d pc e ind n t igv ind s N Ha (is Ha )
n e e n m ind a Ha ;
t e blin er { pcg g ind i is N Ha ( )
e n n ru m ; a
e e bltn eelov pc{ gnor h g s a S Gi ( )
e e w tru nor h g s G i ;
```

次に、`fmf` に戻り、次のマークアップを追加します。

```
x t en e t goer plex l< m à mpæ p = a a' >
x t te eou uebtpl nl < m à mpæ p = a a' >
x et nt x e eu n pd e { p à mpæ p = a a' >
x tit obindm pmo à mpæ p = a a' >
x ro m p f < /a >
g p p < /a à
```

[保存] をクリックしても何も起りません。これは予期される動作です。

次に、`fmf` を含む `displayName` というページを作成します。

```
x t en e t goer plex l< m à mpæ p = a a' >
x e ne el due p oegt mro p à mpæ p = a a' >
x t ne te à upro n à mpæ p = a a' >
t { en o } c à mpæ p = a a' >
e e e ee r t drnr g g à mpæ p = a a' >
x et tx eau ut p dñr ep { g < T r gedaw dt à mpæ p = a a' >
```

このページを保存すると、^{m f} ページ全体がインポートされます。名前を入力し、[保存] をクリックすると、フォームから ^{nor} 値が ^h ^{e g} ^x 項目に渡されて ^{uG} ^立 ^p ^p [&] ユーザ名が表示されます。

別の Visualforce ページを作成し、^{m f} ^{pm} ^{IP} を使用して異なる挨拶文を表示することもできます。

たとえば、^{ld} ^{pm} ^{IP} というページを作成し、次のマークアップを使用します。

```

<template>
    <mp> <apex:include pageName="HelloWorld" />
</template>

```

表示されるテキストが変更されても、^l ^{mp} ^E ロジックは同じままです。

第 18 章

モバイルデバイスの開発

開発者は Visualforce および Apex を使用することで、Force.com プラットフォームでネイティブに実行される高度で強力なアプリケーションを記述できます。Force.com プラットフォームで作成されたアプリケーションをモバイルデバイスに拡張するために、開発者は Visualforce Mobile を使用できます。Visualforce Mobile は、Salesforce Classic の速度と信頼性、salesforce.com のネイティブクライアントアプリケーションを、完全にカスタマイズ可能なブラウザベースのユーザインターフェースと組み合わせています。

Visualforce Mobile は、開発者が、Salesforce Classic によって提供されるオフラインのデータアクセスと Visualforce および Apex によって提供される柔軟性と迅速な開発とを併せて活用できる、クライアント側プログラミングとオンデマンドプログラミングのハイブリッド版です。

Salesforce Classic for BlackBerry と Salesforce Classic for iPhone は、Visualforce ページと Web ページを組み込みブラウザのクライアントアプリケーション内で直接表示できます。Visualforce Mobile ページでは、Salesforce Classic にデータの同期と組み込みブラウザのクローズを行わせるための JavaScript コードを実行することもできます。

Salesforce Classic とは?

Salesforce Classic は、BlackBerry、iPhone、または Windows Mobile デバイスからユーザがデータにアクセスできる salesforce.com 提供のクライアントアプリケーションです。Salesforce Classic クライアントアプリケーションを使うと、無線通信事業者ネットワークを介して Salesforce とデータを交換したり、ユーザのモバイルデバイスのデータベースにあるデータのローカルコピーを保存したりできます。デバイスに送信されるデータはモバイル設定で決定されます。モバイル設定は、ユーザの Salesforce レコードの関連するサブセットを定義するパラメータのセットです。

モバイルデバイスを使って Salesforce にアクセスする各ユーザには、Salesforce Classic ライセンスが別途必要になります。Performance Edition、Unlimited Edition、および Developer Edition を使用している組織には、Salesforce ライセンス 1 つにつきモバイルライセンスが 1 つ提供されます。Professional Edition または Enterprise Edition を使用している組織は、モバイルライセンスを別途購入する必要があります。



メモ: Mobile Lite は、Professional Edition または Enterprise Edition を使用しているモバイルライセンスを持たないお客様が利用できる、無料バージョンのモバイルアプリケーションです。Mobile Lite では、Visualforce Mobile はサポートされません。

Salesforce Classic および Visualforce Mobile を実行できるデバイス

Salesforce Classic は、BlackBerry、iPhone、Windows Mobile デバイスで実行できますが、Windows Mobile クライアントアプリケーションは、現在 Visualforce Mobile をサポートしていません。BlackBerry デバイスおよび iPhone デバイスは次の要件を満たす必要があります。

BlackBerry

Salesforce Classic アプリケーションは、BlackBerry オペレーティングシステムバージョン 4.3 ~ 7.0 で実行できます。パフォーマンスを最適化するためには、Salesforce では、バージョン 4.6 ~ 4.7 がインストールされた BlackBerry スマートフォンで Visualforce Mobile を実行することをお勧めします。BlackBerry オペレーティングシステムを最新バージョンにアップグレードすると、デバイスの全体的なパフォーマンスが向上します。デバイスには、最低 5 MB の空きメモリが必要です。モバイルクライアントアプリケーションは、オペレーティングシステムに関する要件を満たす以下の BlackBerry スマートフォンでサポートされます。

- BlackBerry 8100 Series (Pearl)
- BlackBerry 8300 Series (Curve)
- BlackBerry 8800 Series
- BlackBerry 8900 Series (Javelin)
- BlackBerry 9000 Series (Bold)
- BlackBerry 9500 Series (Storm)

iPhone

Salesforce Classic には、iTunes で使用できる最新の iPhone オペレーティングシステムが必要です。モバイルクライアントアプリケーションをインストールする前に、デバイスには少なくとも 5 MB の空きメモリがある必要があります。モバイルクライアントアプリケーションは、次のデバイスでサポートされています。

- iPhone
- iPhone 3G
- iPhone 3GS
- iPod Touch



メモ: iPhone または BlackBerry デバイスを持っていない開発者はシミュレータを使用して [Visualforce Mobile ページ](#) をテストすることができます。

モバイルアプリケーションの機能と制限事項

Salesforce Classic は、クライアントアプリケーションと Visualforce ページの間で情報を渡すことのできる組み込みブラウザを備える、ネイティブクライアントアプリケーションです。組み込みブラウザは、デバイスのインターネット接続を使用して Salesforce と通信し、ネイティブクライアントアプリケーションは、SOAP API を使用して Salesforce と非同期に通信します。組み込みブラウザは JavaScript を実行できますが、ネイティブクライアントアプリケーションは JavaScript を実行できません。

次のリストに、ネイティブクライアントアプリケーションの機能と制限事項の概要を示します。

使用可能なオブジェクト

システム管理者は、取引先、納入商品、取引先責任者、商談、リード、ToDo、行動、価格表、商品、ケース、ソリューションおよびカスタムオブジェクトをモバイル化できます。カスタムリンク、S コントロール、マッシュアップ、差し込み項目、画像項目はモバイル化できません。ワークフロールール、入力規則、数式項目、および Apex トリガは、モバイルクライアントアプリケーションでは実行されず、レコードが保存されて Salesforce に送信された後で、サーバ側で実行されます。

権限、レコードタイプ、およびページレイアウト

ユーザ権限、レコードタイプ、およびページレイアウトは、Salesforce から継承されます。システム管理者は、必要に応じて、モバイルユーザの権限をさらに制限したり、モバイルページのレイアウトから不要な項目を除外したりして、[モバイル化されたオブジェクトのプロパティを変更することができます。](#)

関連リスト

システム管理者が関連オブジェクトをモバイル化すると(つまり、親データセットに子データセットを追加すると)、そのオブジェクトは自動的にモバイルデバイス上の関連リストになります。

ダッシュボードとレポート

ダッシュボードは、BlackBerry および iPhone クライアントアプリケーションで使用できます。レポートは、BlackBerry クライアントアプリケーションで使用できます。レポートは Excel 形式でデバイスに送信され、基本テーブルで表示されます。モバイルアプリケーションのレポートビューアは、並び替え、集計、小計、またはグルーピングをサポートしていません。

カスタムリストビュー

BlackBerry ユーザはモバイルクライアントアプリケーションのカスタムビューを作成できます。BlackBerry ユーザおよび iPhone ユーザは、モバイル管理コンソールで Salesforce のシステム管理者によって作成されるカスタムビューにアクセスできます。モバイルアプリケーションでは、カスタムビューは 2 列に制限されます。

Visualforce タブと Web タブ

iPhone ユーザおよび BlackBerry ユーザは、Visualforce タブおよび Web タブが Salesforce のシステム管理者によってモバイル化されている場合、モバイルクライアントアプリケーション内のそれらのタブにアクセスできます。ネイティブクライアントアプリケーションではユーザがデータにオフラインでアクセスできますが、Visualforce タブおよび Web タブでは、タブが組み込みブラウザで起動されるためワイヤレスネットワークへの接続が必要です。

Visualforce Mobile を使用する必要がある状況

よく使用されているコンシューマおよびエンタープライズモバイルアプリケーションの大多数は、インストールしたりデータの送受信のためにサーバに定期的に接続したりする必要のある、クライアント側のアプリケーションです。モバイルクライアントアプリケーションがモバイルオンデマンドアプリケーションよりも広く普及している理由には、主に次の 2 つがあります。

接続

モバイルデバイスは常時ネットワーク接続を維持しません。クライアントアプリケーションでは、ユーザはオフラインで作業することが可能で、その場合もデータに中断なくアクセスできます。

速度

ワイヤレスデータネットワークは、依然として非常に低速です。クライアントアプリケーションは応答が迅速です。

Visualforce Mobile はモバイルデバイスのカスタムインターフェースおよびビジネスロジックを作成する方法を提供しますが、開発者が Visualforce Mobile で作業する必要があるのは、ネイティブクライアントアプリケーションの機能を使用してニーズを満たすことができない場合のみです。たとえば、開発者は、カスタムオブジェクトを作成し、カスタム項目を作成し、さらにレコードの更新時にサーバ側で実行する Apex トリガを記述することで、

Visualforce ページと同じ機能を複製することができます。ワイヤレスネットワークの速度と信頼性が改善されるまでは、モバイルユーザにとって最良の操作性が得られるのはクライアントアプリケーションで操作を実行した場合です。

ただし、ネイティブクライアントアプリケーションでは顧客の要件を満たせない状況があります。次の場合は、Visualforce Mobile を使用してください。

正しく表示されません。次の属性定義を使用して、Visualforce Mobile ページで標準スタイルシートを使用しないようにすることを検討してください。

```
< t net eeg dprd pl < lha a f a a Sys s=" s 'a >
```

ページにスタイルシートを追加する場合、**g p p コンポーネントのすぐ下に**セクションを含めるのが最良の方法です。

```
< t net eeg dprd pl < lha a f a a Sys s=" s 'a >
  et e t tx p <c s y y=" ss' >
  et l l h < ! -- s ys >-
  ex g p p < s / y >
  ex l g p p < / a >
```

ページ間でスタイルを再利用するには、スタイルを定義する別の Visualforce ページを作成します。次に、**l du p c** タグを使用してスタイルページを組み込みます。たとえば、**这样一个名前のページを定義するとします。**

```
< t net eeg dprd pl < lha a f a a Sys s=" s 'a >
  et e t tx p <c s y y=" ss' >
  et l l h < ! -- s ys >-
  ex g p p < s / y >
  ex l g p p < / a >
```

次のように他のページにこれらのスタイルを含めます。

```
< t net eeg dprd pl < lha a f a a Sys s=" s 'a > /
  ex x e n e e l du etp cg m < /a > /
  ex g p p < /a >
```

モバイル用に最適化されているスタイルシートを静的リソースとして保存し、ページでそれを参照することができます。ただし、スタイルシートは、ページを表示するためにクライアント側の Visualforce マークアップとペアになっているため、静的リソースとしてスタイルシートを追加するとページの読み込み時間が長くなります。



メモ: iPhone 用のページを構築し、iPhone の標準 UI を模倣する必要がある場合は、iPhone に似たインターフェースを Web アプリケーションに提供するサードパーティのライブラリ iUI を使用することで、時間を節約して開発の手間を省くことができます。

ルックアップ

u ldp p で提供される参照項目セレクタは BlackBerry での適切なユーザの操作性を提供しません。また、iPhone では機能しません。レコードの保存時に参照項目のエントリを検証する Apex トリガを作成すると、この問題を回避できます。また、可能な場合は項目の種別を変更することもできます。

次のトピックでは、iPhone および BlackBerry 用のページ開発についての詳細を説明します。

- iPhone の考慮事項
- BlackBerry の考慮事項
- クロスプラットフォームの互換性のあるページの開発

- JavaScript ライブラリの使用

関連リンク

[Visualforce ページのスタイル設定](#)

[静的リソースの使用](#)

iPhone の考慮事項

モバイルアプリケーションでは、組み込みブラウザで Visualforce Mobile のページを起動します。iPhone に組み込まれたブラウザは、デフォルトの Web ブラウザに使用されるのと同じフル機能の Safari ブラウザです。優良な JavaScript サポートと十分な性能を備えています。

iPhone 用のページを開発するときは、次の考慮事項が適用されます。

ページのズーム

デフォルトで、iPhone ブラウザは、ページの幅を 980 ピクセルに設定します。この値は、さまざまな Web サイトとの互換性を最大限にするために選択されています。iPhone ブラウザで初期ページを表示する幅を指定するには、`<meta>` タグを使用します。

```
wnt e tm tn ronwte oe <p dwt d a mainde "c l h =ki e=v ir mi u-m bi,i i csaF =a . ,0i a s
e ur nl bl o s s - a a " > /
```

このタグは、他のブラウザでは無視されます。

iPhone 特定のアプリケーションでは、ページの幅をデバイスの幅に設定する必要があります。`viewport` メタキーに複数のプロパティを提供する場合、カンマで区切った割り当てステートメントのリストを使用します。次の表は、`viewport` プロパティを説明したものです。

プロパティ	説明
<code>width</code>	<code>viewport</code> の幅(ピクセル単位)。デフォルト値は 980 です。範囲は、200 ~ 10,000 です。ページをデバイスの幅(ピクセル単位)に設定するには、 <code>width</code> 値を使用します。
<code>height</code>	<code>viewport</code> の高さ(ピクセル単位)。デフォルトは、 <code>width</code> プロパティの値とデバイスのアスペクト比に基づいて計算されます。範囲は、223 ~ 10,000 ピクセルです。ページをデバイスの高さ(ピクセル単位)に設定するには、 <code>height</code> 値を使用します。
<code>initial-scale</code>	乗数としての <code>viewport</code> の初期スケール。デフォルトは、表示領域の Web ページにフィットするように計算されます。範囲は、 <code>0.0</code> ~ <code>1.0</code> です。このプロパティによって決定されます。Web ページが初めて表示されるときの <code>viewport</code> の初期スケールのみを設定できます。 <code>user-scalable</code> を設定しない限り、ユーザはそれ以降ズームインおよびズームアウトできません。

プロパティ	説明
viewport-fit: cover	ズーム操作は、プロパティおよび viewport の最小スケール値を指定します。デフォルト値は 0.25 です。範囲は 0 ~ 10.0 (0 を含まない) です。
viewport-fit: none	viewport の最大スケール値を指定します。デフォルト値は 1.6 です。範囲は 0 ~ 10.0 (0 を含まない) です。
user-scalable: yes	ユーザがズームインおよびズームアウトを実行できるかを決定します。拡大縮小を許可するには yes、許可しない場合は no を設定します。デフォルトは no です。user-scalable を yes に設定すると、入力項目にテキストを入力する際にページをスクロールできなくなります。

画面の回転

モバイルアプリケーションでは、画面を回転してもページをフリップしたりサイズ変更したりすることはできません。

URL ターゲット

組み込みブラウザは、URL ターゲットは読み込まれません。

属性をサポートしません。これをページで使用すると、URL ター

ファイルへのアクセス

組み込みブラウザは、ファイルシステム、カメラ、場所、または他のデバイスデータへのアクセスをネイティブにサポートしていません。

静的リソースのキャッシュ

モバイルアプリケーションでは、静的リソース (画像、JavaScript、CSS など) はキャッシュされません。このため、低速の接続ではパフォーマンスが影響を受ける可能性があります。組み込みブラウザは、キャッシュ機能をサポートしません。

モバイル開発の一般的な規則として、次のコンポーネントを使用しないようにしてください。

- アクションの実行を JavaScript に依存するコンポーネント
- Salesforce.com スタイルシートに依存するコンポーネント

Visualforce Mobile ページが前述のカテゴリに該当するかどうかは、ページの HTML ソースを調べて確認できます。JavaScript () を参照する script タグ、またはスタイルシート () を参照する style タグがある場合、ページが期待どおりに表示されることをテストする必要があります。

BlackBerry の考慮事項

モバイルアプリケーションでは、組み込みブラウザで Visualforce Mobile のページを起動します。Research in Motion (RIM) は BlackBerry オペレーティングシステムバージョン 4.3 のリリースで組み込みブラウザをアップグレードしましたが、組み込みブラウザの JavaScript のサポートには依然として制限があります。BlackBerry Bold (バージョン 4.6) および BlackBerry Storm (バージョン 4.7) にはより強力な標準ブラウザがありますが、組み込みブラウザは Visualforce Mobile を完全にサポートできるほど十分には改善されていません。

BlackBerry スマートフォン用のページを開発する場合は、次の考慮事項が適用されます。

JavaScript のサポート

BlackBerry の組み込みブラウザでの JavaScript のサポートは非常に限定されています。インライン DOM イベントはまったく機能しません。BlackBerry 用の Visualforce Mobile ページでは、可能な限り JavaScript は使用しないようにしてください。

フォームとビューステート

Visualforce ページは、要求間でデータベースの状態を維持するためにビューステートに依存しています。

Visualforce ページで `m p f` タグを使用すると、多くの場合、ビューステート変数は BlackBerry の組み込みブラウザには大きすぎて、非常に単純なフォームでも効果的に処理することができません。

フォームを作成する必要がある場合は、標準 HTML フォームを使用するようにしてください。フォームから送信されるパラメータをコントローラ内の `ApexPages.currentPage.getParameters()` で取得できます。HTML フォームを使用するときは、次の点に留意してください。

- ・ ページ間の状態の維持は手動で実行する必要があります。
- ・ 他のページへのリダイレクトは手動で実行する必要があります。

また、`FileInput` タグおよび `FileUpload` タグはコンポーネントではありません。

ユーザがファイルをアップロードできる Visualforce Mobile ページの場合、`m p f` コンポーネントと `FileInput` タグを使用することが最良の選択です。2つのコンポーネントはこの制限のある使用事例で適切に機能します。たとえば、アップロードフォームを作成するには、次のように Apex コントローラメソッドを2つのタグと併用します。

```
<apex:form>
    <apex:input type="file" name="file" />
    <apex:fileupload value="file" />
</apex:form>
```

実装環境では `transient` 変数を使用することでさらに利点を得ることができます。キーワードは、`Transient` です。ポストバック時に保存する必要のないデータに使用されます。前の例では、添付ファイル自体は非 `transient` である必要がありますが、添付ファイルの内容は非常に大きくなる可能性があるため、ビューステートに内容を保存することは妥当ではありません。

解決策は、Blob ファイルの種類を取得するように `FileInput` タグの値を次のように変更することです。

```
<apex:form>
    <apex:input type="file" name="file" />
    <apex:fileupload value="file" />
    <apex:outputText value="File Type: <code>{!file.ContentType}</code>" />
</apex:form>
```

次に、このページの Apex コントローラで `h` を transient として定義します。

最後に、メソッドで、`h` の値を使用して添付ファイルを定義します。

添付ファイルの内容は適切なデータで更新されますが、データは保持されません。添付ファイル自体は非 transient であるため、保存後は `in doch lhi` を `transient` に設定します。

Visualforce タグ配置の誤り

Visualforce タグには、コンパイルして HTML に解決するときに誤って解釈されたり、解釈されなかつたりするものがあります。

- コンポーネントは、コード内で表示されている場所に配置されます。必ずページでの表示位置にコンポーネントを配置してください。たとえば `<apex:component>` コンポーネントはセクションの最下部に配置します。
- コンポーネントと `apex:repeat` コンポーネントで提供される Salesforce の標準スタイルは、崩れてしまうか、あるいは無視されます。より単純なタグを使用するか、純粹な HTML を記述してください。

ページのスタイル

Visualforce Mobile ページのスタイル設定でのベストプラクティスに必ず従ってください。また、BlackBerry の組み込みブラウザは、などの一部の共通の CSS プロパティを無視するため注意してください。

改行

タグは、改行なしスペースなど、行に何かが存在ない限り無視されます。

ナビゲーション

BlackBerry クライアントアプリケーションの組み込みブラウザには、組み込みナビゲーションはありません。Visualforce ページがウィザードである場合、ユーザが以前のページに戻ったり次のページに移動したりできるナビゲーションリンクを使用する必要があります。また、Visualforce ページはタブに埋め込まれます。このため、モバイル Visualforce ページ内のナビゲーションにタブは使用しないでください。

クロスプラットフォームの互換性のあるページの開発

BlackBerry と iPhone の組み込みブラウザの両方で見栄えがよくて正常に動作する Visualforce Mobile ページを作成することは、困難な作業になる場合があります。Salesforce では、次のいずれかのアプローチを使用することをお勧めします。

分割とリダイレクト

BlackBerry 用のページと iPhone 用のページを別々に作成します。Visualforce Mobile ページのタブを作成する場合は、そのタブが BlackBerry 用に最適化されたページをポイントするようにします。その Visualforce

Mobile ページの最上部に、接続しているデバイスが BlackBerry スマートフォンでない場合は、iPhone ページに自動的にリダイレクトする JavaScript を含めます。

```
< n   l   utg g p p < t p a j p a v   saiac "   py="   v / saia " >
w   w e { d o b l f b r r c i k . a y )
w   w t e d o e o r e tne a f g o . h m j r = ! ' E a   P O i i   s ;
} r   c p < s i/ >
t   ex   g p p < /a   a
```

このアプローチでは、すべてのデバイスに最高のユーザ操作性を提供し、開発が長期にわたるという問題を最小限に抑えます。ただし、2つの個別のアプリケーション（デバイスタイプごとに1つのアプリケーション）を管理する必要があります。

最低限の共通機能

最低限の共通機能を組み込み、存在を感じさせない、最小限の JavaScript のみを含めます。タグ内にインラインイベントを含むスクリプトは使用しないようにします。お客様の組織のデバイスによっては、JavaScript を一切使用しないようにする必要がある場合があります。旧型の BlackBerry スマートフォンでは、どのような JavaScript でも、使用するとページが正常に機能しなくなる可能性があります。

条件コード

デバイス条件コードおよびスタイルを作成します。ブラウザによってサーバに渡されるヘッダーに含まれているユーザエージェント文字列で、接続しているデバイスが BlackBerry であるか iPhone であるかを識別します。Visualforce Mobile ページのコードはユーザエージェント文字列を評価し、接続しているデバイスに適したコンテンツを表示します。Visualforce の利点は、マークアップがサーバ側で解釈され、クライアントは条件ステートメントの評価に基づいて、表示できるマークアップのみを取得する点です。条件コードの使用は、最も高度なアプローチですが、コードがさらに複雑になるため必ずしも長期的な最良のソリューションとはいえません。



※モード： 使用した静的リソースへの動的参照（ページ 162）では、代わりに使用できるアプローチを説明しています。このアプローチでは要求の特性に基づいてさまざまな画像を動的に表示します。

たとえば、次のマークアップでは、静的リソース内に保存されている画像の表示のみを行うカスタムコンポーネントを作成します。ただし、表示する画像は、コンポーネントコントローラの調査に従ってブラウザがレポートしたユーザエージェント値に基づいて、実行時に決定されます。

```
< n   l   nom p e o r l d e r c< bolm na l m o p = " i S a " >
e   e { m d e p u e & e s u a i e b o l m e b p m l g R H Q R s ) c . j g t s a p + i y
x   e t n n o o m p < /a
```

```

e   e      d   e      n   c   o   p   h   v i      y = i   ' P ; '
}     }      }   e   e   e l   t   nAur n   ngE o   B{   dB   s ir   sc   , . ki   sa ( ' a   y ')
e   e      d   e   B   dB   rT   c   v i   k   y =   ' a   y '
}
}

```

次の例では、接続しているアプリケーションに応じて、異なるスタイルシートを読み込みます。まず、複数のデバイスで表示するページを作成できます。

```

e   e      li   er o  it   do   fc   &V   !   i-s pp   a   si      >-
d   h      <   >a
n et lee r t et le t tx keh < i   pE s cys h " f y" ou   ls5ol c U! R F Q R s   . G   a
n   l bogc   c   i   y ss/   /   ss   " )> /
tn   etn &e   teic le { er < keu   i i   oua Sysbol sc $s   =! R s   . G   "a > /
n et lee r t et le t tx keh < i   pE s cys h " f y" ou   sd' d U! R F Q R s   . S i   a,
n   l olc c   c   c   i   y ss/   /   ss   " )> /
tn   etn &e   teic le { er < keue i i   oua Sysd   sd' s   E=! R s   . S i   "a > /
d   h      <   / >a
...

```

z1 bol e n ファイルおよび は CSS ファイルは参照されている CSS ファイルを含む静的リソース
tn etn &e です。teic l h カスタムコンポーネントの場合、ブラウザの種類に基づいて表示される
複数の CSS 宣言を定義できます。

```

e   e   h   ro n   o   ofm   ddoo   pi/s   a
x e   t n n &e   nom   p   or   ldp m c< nte a   olC   or   l& r C   =" y i i   a   " >
xtt e   t e   en b   eu   pe   rm   &u   el   ea   cia o   &   r   p   oash   t" T   scm   i   r =!   op s   "a   y"   is "
et r   n &f   r u   i   =!   " > /
e   rB   t1B   fr e   wed   rd   br   lo h/   Ba   gald   s   a   a   s   '   .   ,   '   a   y"   is "
xt   te   nou tue   lpdn   &pu   {n< o   redawel   br   Pa   er   ya=B   m   1B   }   rr   =! c   !   s   Nk   &   '   a   y" >
n et lee r t et le t tx keh < i   pE s   cys h " rf   y" ou   ss' c   U! L! R F Q   (s   ,
n   e   wBBB   r cocr}   ci   y ss/   /   s   .ss   " )> /
xt   te   nou u   lp   p   <   /a   Pa>
e   edB   e   1B   fr e   bcdmd   brn   lo h/   a   aro   yl   fc   Cs   ic   Ssa   s$ a
ex A   t   edn   &   eet   p   wntgdn   //   h   e   re   li   ri   d   i   b   ddmd   br   o   r   a   a
xt   te   nou tue   lpdn   &pu   {n< o   redawel   br   Pa   er   ya=h   m   "   o   }   ro   =!   fcs   N   &   Ssa   " >
n et lee r t et le t tx keh < i   pE s   cys h " rf   y" ou   ss' c   U! L! R F Q   (s   ,
n   e   BB   bcdmd}   E   d   y ss/   /   .ss   " )> /
xt   te   nou u   lp   p   <   /a   Pa>
t   e   ro n   ef   w   rk   br   ne   r f   &   d   i   l   Pro   Sta   a   f   ro   d   q   sc   Ssa   s$ a
xt   te   nou tue   lpdn   &pu   {n< o   redawel   br   Pa   er   ya=h   m   "   o   }   ro   =!   sfN   &   i   ' P   -Sia   a" >
en   wmt   e   m   tronwt   o   <   p   d   n   a   wai   h   12"   lx   e   =!   &   m=3u   m   &   i   sa   =a   ;   Oi   a   s   &   a
e   ur   l   bl   c   <   / > a
n et lee r t et le t tx keh < i   ep=   s   e   ysh   "   rf   y" ou   e   ss/   nc   U! L   k   RCF   Q   Ih(s   c   ,i   y ss/   / p
xt   te   nou u   lp   p   <   /a   Pa>
x e   t n n   o   om   p   p   <   /a   >

```

e we br 最後に、 m 値は前の例と同じ方法で Apex コントローラで決定されます。



メモ: Salesforce Classic は、テキスト `fc` を BlackBerry の組み込みブラウザ用の文字列の末尾に追加します。また、一部の BlackBerry スマートフォンでは、ユーザはユーザエージェント文字列を変更できます。

```

ex A e   do      pc          pp //      si
t t  bh e   it ep rwe B cg r g o r i m i a i s s N a( )
e e   rt enAuge x Age te negat e sc eg t ed re Ba gnAr PgH. ) a ( U ) ( s ' - ; '
e e   t nAur n nf o n c oi s i sa( ' P ' )
t t   en r rm o rh f i . i sa( ' Sia a ' )
e e   t nAur n enf oe cl iro fc . i sa( ' Ssa ; ' )
t t   en re ru l ro fc . i sa( ' a y ' )
t t   t nAur n nge o B dB irr sc . ki sa( ' a y ' )
t t   ne r rBu lB rr c k ' a y '
e tn r ru o r h ' ; '
}

```



メモ: Salesforce Classic 用の JavaScript ライブリのコマンドは、iPhone デバイスと BlackBerry デバイスの両方で使用できます。

JavaScript ライブリの使用

Visualforce Mobile ページを開発するとき、Salesforce Classic でアクションをトリガするコマンドを含む JavaScript ライブリを活用できます。このライブリを使用すると、Visualforce Mobile ページとネイティブクライアントアプリケーションとの間のシームレスな操作性を提供できるようになります。

新しい JavaScript ライブリのアクションを、どの Visualforce ページでも使用できます。これらのコマンドは Visualforce をサポートする JavaScript 対応デバイスで機能します。現在、これらのデバイスには、iPhone と BlackBerry スマートフォンが含まれます。BlackBerry スマートフォンで表示されるページに JavaScript ライブリを使用する場合、Salesforce では、デバイスにバージョン 4.6 以降の BlackBerry オペレーティングシステムをインストールすることをお勧めします。



ヒント: 共有 JavaScript ライブリを使用することには、iPhone と BlackBerry の両方のオペレーティングシステムでコマンドが機能するという利点があります。

ライブリの関数をコールするには、少量の JavaScript コードが必要です。次の関数があります。

`mobileforce.device.sync()`

モバイルクライアントアプリケーションを Salesforce と強制的に同期し、デバイスのデータレコードを更新します。

`mobileforce.device.close()`

Visualforce ページが表示されている組み込みブラウザを閉じ、元のタブまたはレコードにユーザを戻します。

`mobileforce.device.syncClose()`

モバイルクライアントアプリケーションを Salesforce と強制的に同期し、Visualforce ページが表示されている組み込みブラウザを閉じます。

`mobileforce.device.getLocation()`

デバイスの現在位置の GPS 座標を取得します。



メモ: HTML リンクを使用して sync コマンドと close コマンドをトリガすることもできます。JavaScript のサポートが制限される BlackBerry スマートフォンで使用できる、すぐれた代替手法です。コマンドをトリガする HTML を使用するには、`r` タグに `h` 属性の値として、次の文字列を含めます。

- ・ クライアントにデータの同期を強制的に行わせるには。`n fc : c` を使用します。 `//sync`
- ・ 組み込みブラウザを強制的に閉じるには。`l fc : c` を使用します。 `//close`
- ・ 組み込みブラウザを強制的に閉じ、クライアントにデータの同期を行わせるには、
`e bolm ren fcl: o c c i` を使用します。 `//sync/close`

Visualforce ページで、次の静的リソースを使用して JavaScript ライブリをポイントします。

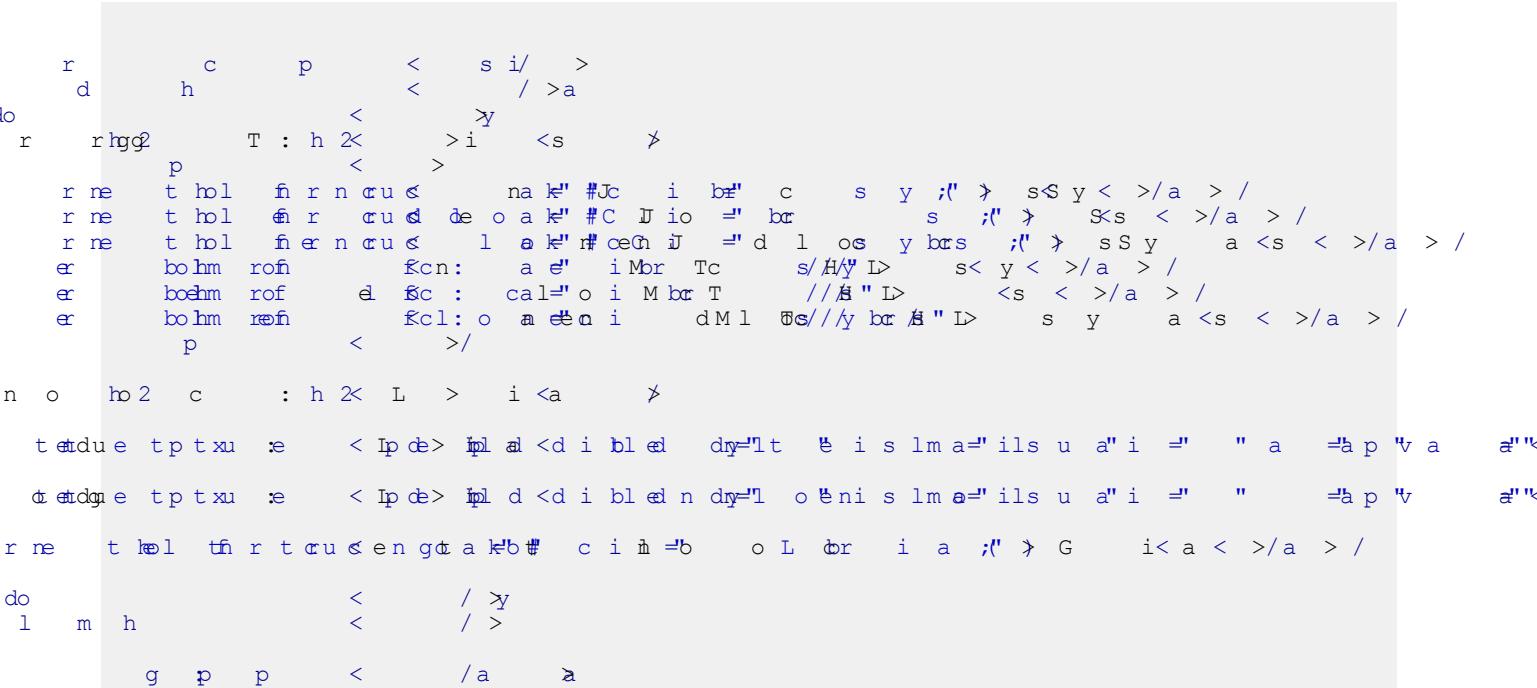
```
r t c xl p p <pp sj e eryt balm a ev/-sablm so y/ fci i ca i/ p .s' >
```

外部 Web サイトは、`c` パラメータにインスタンス名を含める必要があります。

```
r r t c xl p p <pp sj i yt ia pi a v/-sai " r en e c h l e proe ds balm 1e //sa balm ro.p y/ fci i ca i/ p .s' > s i/ >
```

次のコードは、JavaScript ライブリで使用できるすべてのコマンドを使用した Visualforce ページの例です。

```
e ew eg opd ph h < a s a' s'a >
mtl in www w h tsop g ls=" m h // 3 99 / / " >
d h < >a
e l e lu re e bol tfce M r gg is T a Ti i <s i / >
en wmt e tn tnronwte oe < p dwt d a hairnde "c lh="k e=v ic m u-m; m i i csal =a ; Oi a s
ur l bl c s s - a a;" 0 > /
et tnt e g r ou c< U! c-s s i a s >-
r t c xl p p <pp sj ie eryt balm a ev/-sablm" so y/ fci i ca i/ p .s'
c r c p < s i >
n n un o fc c i s y ( )
e e bolm ro dn fc c i s y ( )
t e nr ru l f s a
}
n en u od dfco C i s ( )
e e bolm eo d fd o c c i v. i . s ;( )
t e nr ru l f s a
}
n neun{ o fd o c C i s y s ( )
e e bolm ro en n fc l o i c s y v. i s y s ;( )
t e nr ru l f s a
}
t d n d n o ptn fu al flc d a i a = i ( a )
et atde o ut mn dg B en de l t Il u l. y ( a va . ) a= ;
et atde o ut mn dg B en de l o Il n u l. o y ( v . ) a= ;
}
en tn ut {o nfgo o c i L i a ( )
e e bolm rd dt f o n tgo c to d c n o v. o p Lc i a (L a i a ; )
e nr ru l f s a
```



Visualforce ページのモバイル化

モバイルブラウザで実行できる Visualforce ページを開発したら、ユーザが Salesforce Classic で Visualforce ページにアクセスできるようにセットアップを実行する必要があります。

次のトピックでは、Visualforce ページをモバイル化する方法を説明しています。

- モバイル準備完了の Visualforce タブの作成
- モバイル設定への Visualforce タブの追加
- Visualforce Mobile ページのテスト

モバイル準備完了の Visualforce タブの作成

Visualforce ページをモバイル化するには、カスタムタブを作成してモバイル準備完了として定義し、モバイル設定に追加できるようにします。

Visualforce Mobile タブを作成する手順は、次のとおりです。

- [設定] で、[作成] > [タブ] をクリックします。
- [Visualforce] タブ関連リストで、[新規] をクリックします。
- カスタムタブに表示する、モバイル用に最適化された Visualforce ページを選択します。
- タブに表示する表示ラベルを指定します。
- タブスタイル ルックアップアイコンをクリックして、[タブスタイルの選択] を表示します。

タブスタイルがすでに使用されている場合は、タブスタイル名の横に角括弧 ([]) で囲まれた数字が表示されます。スタイル名上にマウスポインタを置き、そのスタイルを使用しているタブを参照します。タブで未使用のスタイルのみを表示するには、他のタブで使用されているスタイルを非表示] をクリックします。

6. タブスタイルをクリックして、カスタムタブの配色とアイコンを選択します。
7. [モバイル準備完了] チェックボックスをオンにして、Visualforce ページがモバイルブラウザに正しく表示されて機能することを示します。
このチェックボックスをオンにすると、モバイル設定で使用可能なタブのリストにそのタブが追加されます。
8. 最初に表示されるスプラッシュページとして使用するカスタムリンクを選択しないでください。モバイルアプリケーションは、スプラッシュページをサポートしません。
9. 必要に応じてタブの説明を入力し、[次へ] をクリックします。
10. 新規カスタムタブを使用できるユーザプロファイルを選択する手順は、次のとおりです。
 - ・ [1つのタブ表示をすべてのプロファイルに適用する] を選択し、ドロップダウンリストから [デフォルトで表示]、[デフォルトで非表示]、または [タブを隠す] を選択します。
 - ・ または、[プロファイルごとに異なるタブ表示を適用する] を選択し、各プロファイルのドロップダウンリストから [デフォルトで表示]、[デフォルトで非表示]、または [タブを隠す] を選択します。
11. Salesforce デスクトップユーザにタブが表示されないように、利用できるすべてのアプリケーションから新しいタブを削除することを検討します。Visualforce Mobile ページからは、通常、多くの標準の Salesforce 要素が取り除かれているため、ユーザがデスクトップブラウザからページにアクセスできるようする必要はありません。
 - ・ 利用できるすべてのアプリケーションの横にあるチェックボックスをオフにします。
 - ・ 各ユーザのカスタマイズ設定にタブを追加する] チェックボックスをオフにします。
12. [保存] をクリックします。

モバイル設定への Visualforce タブの追加

Visualforce ページをモバイル化するには、Visualforce タブをモバイル設定に追加する必要があります。モバイル設定とは、Salesforce がユーザのモバイルデバイスに転送するデータと、そのデータをモバイルデバイスで受信するユーザを決定するパラメータのセットです。複数のモバイルユーザの異なるニーズを同時に満たせるように、組織で複数のモバイル設定を作成できます。たとえば、あるモバイル設定ではリードと商談を営業部門に送信し、別の設定ではケースをカスタマーサポート担当者に送信する、というように設定できます。

モバイル設定を設定する手順は、次のとおりです。

- ・ モバイル設定を作成する
- ・ データセットを定義する
- ・ モバイルオブジェクトプロパティを編集する
- ・ モバイルタブをカスタマイズする

モバイル設定についての詳細は、『[Salesforce Classic Implementation Guide](#)』を参照してください。組織でモバイル設定がすでに作成されている場合は、タブのカスタマイズのステップに進んでください。

モバイル設定を作成する

モバイル設定を作成する前に、ご自分のユーザアカウントがモバイルライセンスに割り当てられていることを確認してください。確認するには、ユーザレコードを編集します。[モバイルユーザ] チェックボックスがすでにオ

ンになっている場合は、その他の操作は不要です。[モバイルユーザ] チェックボックスがオンになっていない場合は、プロファイルまたは権限セットの「モバイル設定の管理」権限を有効にします。



メモ: Developer Edition、Unlimited Edition、および Performance Edition を使用している組織では、デフォルトですべての Salesforce ユーザにモバイルライセンスが割り当てられています。

モバイル設定を作成する手順は、次のとおりです。

1. [設定] で [モバイル管理] > [モバイル設定] をクリックして、モバイル設定リストページにアクセスします。
2. [新規モバイル設定] をクリックします。
3. モバイル設定の名前を入力します。
4. [有効] チェックボックスをオンにします。モバイル設定は、このチェックボックスをオンにするまでは機能しません。
5. 必要に応じて、モバイル設定の説明を入力します。
6. 必要に応じて、[最近使ったデータのモバイル利用] チェックボックスをオンにして、Salesforce で最近使用したレコードにデバイス同期化のマークを付けます。
7. [最近使ったデータのモバイル利用] チェックボックスをオンにしている場合は、[最近使ったデータの最大数] ドロップダウンリストから値を選択します。
8. [選択可能なユーザ] ボックスでユーザ名を選択し、[追加] 矢印をクリックして、モバイル設定にユーザーアカウントを追加します。
プロファイル全体または個別のユーザをモバイル設定に追加できます。
9. 合計データサイズの制限を設定するには、[データサイズの最大値] ドロップダウンリストを使用して、このモバイル設定に割り当られているユーザのモバイルデバイス上で常時使用可能なメモリ量を指定します。
Visualforce Mobile ページのテストのみを行う場合は、デフォルト設定が適切なサイズです。
10. [保存] をクリックします。

データセットを定義する

モバイル設定を設定する手順における次のステップでは、モバイルデバイスと自動的に同期するオブジェクトとレコードを決定します。Visualforce Mobile ページのテストのみを行う場合は、データセットの定義は不要です。ただし、オブジェクトのレコード詳細ページから Visualforce Mobile ページへのリンクを作成する場合は、ネイティブのレコードと Visualforce Mobile ページとの統合をテストできるようにそのオブジェクトをモバイル化する必要があります。レコードから Visualforce Mobile ページへのリンクの作成方法を確認するには、『[Salesforce Classic Implementation Guide](#)』の「Creating Mobile Links」というタイトルのトピックを参照してください。

データセットを追加する手順は、次のとおりです。

1. モバイル設定の詳細ページを開きます。
2. [データセット] 関連リストで、[編集] をクリックします。
3. 階層内で、[データセット] を選択して親データセットを作成するか、既存のデータセットを選択して子データセットを作成します。
4. [追加...] をクリックします。
5. ポップアップウィンドウで、モバイル化するオブジェクトを選択します。

1. モバイル設定の詳細ページを開きます。
2. 初めてモバイルタブを定義する場合は、[モバイルタブ] 関連リストで [タブのカスタマイズ] をクリックします。モバイルタブがすでに設定されている場合は、[編集] をクリックします。
3. [選択可能なタブ] リストでモバイル化する Visualforce タブを選択し、[追加] 矢印をクリックしてモバイル設定に追加します。Visualforce タブが [選択可能なタブ] リストに表示されない場合は、タブを編集してモバイル準備完了とマークします。

標準オブジェクトまたはカスタムオブジェクトをモバイル化した場合、タブをカスタマイズするときに忘れずにそれらのオブジェクトを選択してください。また、[ダッシュボード] タブをモバイルアプリケーションに表示するには、このタブを選択する必要があります。

4. [選択されたタブ] リストでタブを選択し、[上へ] と [下へ] の矢印をクリックして、モバイルアプリケーションに表示する順番にタブを配置します。



メモ: iPhone ユーザは、モバイルクライアントアプリケーションでタブ順序をカスタマイズできます。ユーザがタブ順序をカスタマイズすると、管理者によるモバイル設定のタブ順序の変更はクライアントアプリケーションで無視され、新たにモバイル化されたタブはユーザの既存のタブの下に追加されます。

5. [保存] をクリックします。

Visualforce Mobile ページのテスト

Visualforce Mobile ページを開発したら、モバイルアプリケーションでテストし、期待どおりに表示されて機能することを確認します。BlackBerry スマートフォンまたは iPhone にモバイルアプリケーションをインストールして実行する方法については、『[Salesforce Classic User Guide for BlackBerry](#)』または『[Salesforce Classic User Guide for iPhone](#)』の「Installing Salesforce Classic」を参照してください。

Salesforce Classic デバイス要件を満たす iPhone または BlackBerry スマートフォンがない場合は、iPhone または BlackBerry シミュレータでモバイルアプリケーションを実行できます。シミュレータのインストールと実行方法については、『[Salesforce Classic Implementation Guide](#)』の「Mobile Device Simulators」を参照してください。

Visualforce Mobile ページをテストするとき、次のいくつかの管理タスクを実行する必要がある場合があります。

データの同期

モバイルアプリケーションは、Salesforce のスキーマ変更や新しいデータを 20 分ごとに確認します。場合によっては、モバイル設定の編集後または Salesforce でのレコード作成後のデータを同期して、その変更が直ちにアプリケーションに反映されるようにします。モバイルアプリケーションを強制的に Salesforce と同期することができます。

iPhone のデータを同期する方法については、『[Salesforce Classic User Guide for iPhone](#)』の「Synchronize Data」を参照してください。BlackBerry スマートフォンのデータを同期する方法については、『[Salesforce Classic User Guide for BlackBerry](#)』の「Refreshing Data」を参照してください。



メモ: Visualforce Mobile ページで JavaScript ライブリのコマンドを使用し、モバイルアプリケーションがデータを同期するように強制できます。

別のユーザーアカウントのテスト

多くの場合、開発者は Salesforce 組織に複数のアクティブなユーザーアカウントを作成しています。Salesforce Classic でユーザーアカウントを有効にしている場合、別のユーザーアカウントを登録する前にそのアカウントを無効にする必要があります。

シミュレータの代わりにモバイルデバイスを使用して Visualforce Mobile ページをテストする場合、モバイルアプリケーションからアカウントを無効にできます。iPhone から Salesforce アカウントを無効にする方法については、『[Salesforce Classic User Guide for iPhone](#)』の「Erase Data」を参照してください。BlackBerry スマートフォンからアカウントを無効にする方法については、『[Salesforce Classic User Guide for BlackBerry](#)』の「Removing Salesforce Data from Your Device」を参照してください。

シミュレータを使用して Visualforce Mobile ページをテストする場合、Salesforce でアカウントを無効にする必要があります。Salesforce でアカウントを無効にする方法については、『[Salesforce Classic Implementation Guide](#)』の「Deleting Mobile Devices」を参照してください。

Sandbox アカウントのテスト

デフォルトでは、モバイルクライアントアプリケーションは本番組織のトランスポートに接続しますが、sandbox 組織でテストすることができます。Sandbox アカウントを有効にする方法については、『[Salesforce Classic Implementation Guide](#)』の「Activating a Sandbox Account in Salesforce Classic」を参照してください。

例: iPhone 用の対応付けアプリケーションの作成

モバイル開発の概要を説明するために、この章には iPhone 用のアプリケーションを作成するプロセスを説明するいくつかの例があります。このアプリケーションは、Google Maps Web API を使用して顧客の優先度順に見込みのある取引先を対応付けます。これらの例を実際に動作させるには、必ず次の要件を満たしている必要があります。

- Developer Edition の組織: Developer Edition を取得していない場合は、Developer Force で Developer Edition の組織にサインアップします。
- テストデータ: Developer Edition を使用している組織で、ユーザの取引先に有効な住所が含まれていることを確認します。次の 2 つの取引先の住所（請求先）を編集して、会社が自分の住所の近隣になるようにします。
 - ◊ Edge Communications
 - ◊ United Oil & Gas Corp.

これらの住所が互いに近くなるようにすることで、例をテストしているときに、マップにすべての取引先をより簡単に表示できます。

- UI ライブリ: Web アプリケーションで iPhone の標準の UI を簡単に模倣できるようにするサードパーティのライブラリ iUI をダウンロードします。
- Google Maps API: Google Maps API にサインアップして、Maps API キーを取得します。
- iPhone Simulator: モバイルアプリケーションで Visualforce ページをテストできるように iPhone シミュレータをダウンロードします。
- モバイル設定: 例が完成したら、忘れずに Visualforce タブおよび取引先オブジェクトをモバイル化するモバイル設定を作成します。

この章の Visualforce Mobile の例には、次のものが含まれます。

- カスタムコントローラの作成
- 対応付けおよびリストビューの作成

- #### ・ 詳細ページの作成

カスタムコントローラの作成

対応付けアプリケーションを作成するには、まず、マップおよび対応する取引先のリストを表示する Visualforce ページで参照されるカスタムコントローラを作成する必要があります。このコントローラでは、評価が「見込み有り」のユーザの取引先を取得して、Visualforce ページで対応付けを行う JavaScript ルーチンで使用するための、区切り文字で区切られた取引先の文字列配列を作成します。さらに、Salesforce ページで Google マップを使用するために必要な Maps API キーの getter メソッドも定義します。

次の Apex クラスは、ユーザの「見込み有り」の取引先を対応付ける Visualforce ページのコントローラです。

blu tle f pc mor l d r C p i ss a a ;s
t bln tr p dgr r i is a ;s
t bln r pc m i is y Ky
t An t oti(p c Ac g] M dc [s y s ()
n er rd nt r I o gif difS s UI = s U . s ;()
t An t eue dc e] lcc td ncet ll [sga or [sm Gro N ar i ac , s i i i yj i y , s a,
e w n etBn t ll I r gBn ll OetBt li G g S i i s P a
Bt ell ogl do C i i F
t rAno mou cc
eew t n t h g o R i aH= '
e nwA n d r d n DI : I] O = s ;
t Amt oti f cc cc : cc (a s)
t ddt r dd r cc a s a S + aN a '
t et tu occrmGro ca . s : i + R y '
t dt cc BtBn t:ll tcrg at. ~ ~ ' :ai it s ~ ~ '
t Bn ll cc g +t C etBt: al+icc ig + y ~ ' : ai ti s a ~ '
t Bt ell cc ogl + do C ai :i + s P a ~ ~ ' ;'
}
tr ru cc s;
t e blnt f pc g tm e i u eiSg g my Ky() p// s k saia a y
tt m e h m cdg g o p y Ky d ' pfa p 2 /s a ? = iv a=& ;& '
et n we n deInch m fg e r s u odg gi mi , p y k sa y
t eno g tet e p bAk thgA a a Myi ip I G sa \$P
ct e e dd cdg g o m tc /m p y / lp m h p i /sa /s a / .
e em e m tn e r eodg Ky my Ky < #ip sp h _ k _ saia a _ y _ >; '
e nr ru m y Ky
t e blnt r t Ap c Agdr r r i is s ()
t ddr r t cc t] hA gc a s ; ''
t Ane ou t cc t] hA gc M cc [R s = y s i()
n t ru ddr r a ;s

関連リンク

カスタムコントローラの作成

対応付けおよびリストビューの作成

対応付けアプリケーション作成における次のステップは、マップとそれに対応する取引先のリストを表示するVisualforceページの作成です。Visualforceページは、Googleマップオブジェクトのパネルを定義し、取引先のリストを表示するグループサブパネルを作成します。さらに、JavaScriptを使用して取引先住所を取得し、顧客の優先度に基づいて色付けされたマーカーをマップに入力します。JavaScriptは次のロジックを実行して、マップオブジェクトを設定します。

文字列配列から対応付ける住所を取得する

- コントローラに定義された区切りを取得し、住所配列を解凍する
- すべての取引先住所と現在のユーザーに対し、c をキーとして使用し、緑、黄色、赤のうちどのマーカーを使用するかを決定する

静的リソースに保存されたカスタム画像マーカーを取得する

JavaScriptコードは、複数の場所で参照する必要がある場合に備えて静的リソース内に配置しておくのが適切な方法です。M という静的リソースを作成します。

```

n n ea t ddnfc d n u E fic L a v a ( ) 
et nr lor d nd n d o d av a =ai . ; a 
w w n o n fd o dtf dh ni u (oy fic . 
w n d o d nd u fci . =a ; 
w l n d o t d fd n u o fci . =a i ( ) 
n ld d d fc ;( ) 
} 

e t ddn d E L a v a ( 
f n u o fc i ( ) 
e w B rteo r{ f o inl I C ip (G s s i a ( )) 
e e re me goret enc l v o dx y c = G ;( ) 
e in ne et stop o utM in p dg B =an d E G ma t . p y" ( " a; ) 
e w m e t nc T oM lov p C a = G a y ;( ) 
tt t do o on m gn th or in o C a A R B = z s M IT H GN _ T Q O_ OHR _ G , iG S 

t nr om C v a = ; 0 
t en idwn or lo p ct ml m or loM ac p w ll( mrrG $ a a ( )) // s a as 
t n dd tar lo pt o o C m g atha n b(Mu, o p R p ; ) // a y s 

t n n ue φ r fc in u T iL i v ( a ) 
e r \ \ v * a += * $ / (( $ ) ) / 
e t e ner eue l ur l r φ v a. $ a (1 ", " ; ) 
} 

t n n ue φ r fc in u T i R i v ( a ) 
e r \ \ v * a += * $ / (( $ ) ) / 
e t e ner eue l ur l r φ v a. $ a (1 ", " ; ) 
} 

e e e ea in d n endg ed g R / h i ac p a i i s sa 
t n t n ue { o r fc in u i v ( a ) 
e r ru er rm lm uT T L i R (i v ( a ; ) ) 
} 

t n n u t Ad ddn eote o oeet t Mt Thpt t i at{ C ia SN a, S pi py, Z S ai , y 
eet r tddr r t v et t a @= s " +", i + y " #, p + S a " Z" i ; 

```

e e t m ogor n g dgr y _ L . L a (a ,
t n ft u n o ofc p i i)
{ t n o f p T v a =a ; ''
e n rn m g n Ic o A vIc a Ey = I C T G D VGLF_ Q ;)

t e { lf p f i (y== s ')
e n negb T m b +t <dr > +i SN +ac" >< " > i s N /a . "
re en rn vIc z c h meadgy giao w ab nr o fp/s a g . p a i s
e e { l s
t t f p cc i (y== 'a ')
t n om c ++ ;
e t ar r t p in v ap : = i SN a i " (;)
t r r o m Ar r p e n v o] a Nc a pi ; [0 // ay //
t r r r Ar pr t vpr kro a = ip i1 f) a y //

```

        t Ad ddn o o o cr lo M r plo ] r Gao ] a Cas 2 [ Q las [ , as [ ,
        r lo rt do ] C ] c cas 3 [ , as , a ; )
    }

e te e t Ad ddn o e o o c te { M e T pl /G s ass a a
{ e { e est t{ Mr e} rt t $ n ( U ), ! 's C . $ S U , ! 's i y , ' '
e et t{ re} t e $re o yl !dos $ l $ a U , ! 's f . s P a s ' ; )

;
)

```

次のコードは、対応付けアプリケーションのリンク先ページを定義します。

```

x t en gorpde pnc etorwld ra Cep a d r h # a f " s a" s 'a >
x t et et e dm p lpm < u pa s if =a v " > /
{ er rect p t pax < ts i jsp e ! ycky py" c v p saia "< > s i/ >
x e n t d du rep c d a < e p ca boll ic$ v M a! R s . M i s i " > /

c lut dAen e touet dc de r ic d =" o ms" hs = " i = " >
ew e r nt tur e m no < o D l -- pf a p a a a >-
1 1 r og < p i ss=a " >
e { e r te { r er } te $ rs y ! s m E $s N U ! B . s Na a
1 < i >

e et e rn e lc roe opt< gf ! be- M a p ac G s a >-
ed lnt d l n p d v g sp a : " a s p y" i a ;'0 >
d et d wt m l e xd p < v h x" " y ap y" i 3 p 00 i 3 p0 >
d < v />

e et t r e r d bu ou l < p ! l-p l a ps < s i >-
1 1 t A p ou $ p ci ss=a " > - a i s jas >-
e ex t e t ren e nou wcc r $ oD -- p a a s , >-
{ r kpu Ap} < Mr ac v a a! p ! y s "v = a" >
1 < i >
e { etr tn { hou f dc d } < d a = " Ipa i ia ? ! . " >
tt r e d m p < i i = " ! . " >
{ e } m p < >/a
1 < i >
e ex t r p p < /a >a
lu g p p < /a >a

e ページのマークアップは、om w p <コンポーネントを使用して、テンプレートを参照します。テン
ブレートは、iUIフレームワークを利用して、iPhoneのようなスタイルをページに適用します。iUIフレームワー
タは、c $I 静的リソースから組み込まれます。テンプレートを定義することによって、iPhoneプラッ
トフォーム用に作成するすべてのVisualforceページに簡単に同じスタイルを適用できます。

```

次のマークアップは、テンプレートとして使用される ページを定義します。

```

< !
-->
t n d g o* fi : fPa i i i i v
e et elit ro k fe ne e p mlsflu la dcd ro ad i fi p // .
n ue gw utt*h eron $ ddh adic g o anuc a . i / > /
n e e lu r$ fyc iv p ay s a a
>-

g p p < a >
x wmt e m tronwt o < p d n a vai h l2" l = a =3 ; i 0 i sal =a > 0
x e neum h e cur l bl i a s l =a ; $ s a a ;" 0> /
x e n t d du rp c le a < e p a ou i Sv c u II R FuB s y . i il ' -30 i a i / s " "

```

```

    t e { l      p le h e <      as ouy   S      c u I R Fu s U . c i il ' -30 i A i / .ss " " >
e {   l t @ m etn & r & p s x :y> }      :i pl p v a; ;@ s / y>
      g p p < /a a

```

テンプレートについては、次の点に注意してください。

- マークアップは、iUI ライブリリースの `h` スタイルを上書きします。そうすることで、ページ上部に目立ったギャップを作ることなく Salesforce Classic にアプリケーションが表示されるようにしています。
- マークアップは、`apex:composition` 要素の使用を回避します。Salesforce Classic に組み込まれたブラウザでは、ページの上部にナビゲーションツールバーが表示されるため、2つ目のツールバーを表示するとユーザを混乱させる可能性があります。iUI フレームワークに提供されるボタンスタイルを使用する場合は、ボタンを表示するために `button` クラスを使用しないようにしてください。

関連リンク

[Visualforce ページでの JavaScript の使用](#)

`apex:composition`

[静的リソースの使用](#)

詳細ページの作成

対応付けアプリケーション作成の最後のステップでは、リストビュー内の取引先の詳細ページを作成します。まず、取引先情報を取得するコントローラを作成します。

```

t blt l t pAuto in mou orcadair C ss a s
t r n t Ah pt onf cou vcc a a ;
t blt t Aut @ pncfou orcadair C i s ( )
t ene tou e cd td ncet a gao er [SmGro tN aenir acn,os c o ip h y _ si i
eetBn t ll r gBn ll etB@ li Gi BtSell@ i ogl do@ rAno C mSousaf icc s P _ a
e re x d A eh te negrntpe e t g e =gt rm Ba. gd Pa(.) ] Pa a s (.) i( ' ; ')
} etbilat t pAn cg ou cc i ( )
t n rt rm ou cc i a ;
} ee e e bie en { grc f c i Ra a ; a ( )
e t td n ou p cc a a ;
t n m ru lh ; ;
}

```

次に、ユーザがリストビューから選択した取引先の電話番号および評価を表示する Visualforce ページを作成します。このページに iPhone に似たスタイル設定を追加するには、iUI フレームワークの `frame` クラスおよび `style` クラスを使用します。>

次のコードでは、対応付けアプリケーションの取引先の詳細ページを定義します。

```

x e ew eg op d pte m < dr ad r c Aut @ n attribute sohdair C l@ As ou M cc " i = Y
x t e t ne dm po lpm < u pa $ if =@ v " > /
x ed ln et dt de e ep @ det ccl set@ r "aic =" D n a dd "as g p=" : " sp y" i a ; 0
t n r mate @ o t lAp npo u i aco - mo44fi ="

```

```
t eAn      du2 cc m      < h 2 > !      N < a >
e t et    ld      lfn r m g < i : s 2s y" ia 0 0 0;" 
e t et    d w l r oc l bhl : < ss=a " >
e t et    l bhl o l t pAn < a> <P / a>
e t et    e t txue { p l t pAn < inou} oy" Mv a! . P " > /
d          < ss=a " >
e t et    d w l r oc lgbl : < aR i <a / a>
e t et    l bhl o l t pAn < inou} oy" g v a! R i a" > /
d          < ss=a " >
e t      ld      f      < i / s >
d          < ss=a " >
g p p < / a >
```

第 19 章

Force.com AppExchange アプリケーションへの Visualforce の追加

Visualforce ページ、コンポーネント、またはカスタムコントローラを、AppExchange 用に作成するアプリケーションに含めることができます。

Apex クラスとは異なり、管理パッケージにおける Visualforce ページのコンテンツは、パッケージがインストールされている場合は非表示なりません。ただし、カスタムコントローラ、コントローラ拡張、およびカスタムコンポーネントは非表示になります。さらに、`cc` 属性によって、カスタムコンポーネントがご使用の名前空間でのみ実行されるように制限できます。

Salesforce.com は、Visualforce コンポーネントまたは Apex コンポーネントを配布する場合にのみ管理パッケージを使用することをお勧めします。これは、管理パッケージが、ページ、コンポーネント、クラス、メソッド、変数などの名前の先頭に自動的に追加される固有の名前空間を受け取るためです。この名前空間のプレフィックスによって、インストーラの組織内で名前が複製されることを防ぐことができます。

Visualforce ページを使用してパッケージを作成する場合は、次の警告を考慮する必要があります。

- 管理パッケージに含められるコンポーネントの `cd bo` 属性が `global` に設定されている場合は、次の制限事項に注意してください。

◊ コンポーネントの `cc bl` 属性は `pc` に変更できません。 [i](#)

◊ すべての必須の子 `b u p` コンポーネント (`true` に設定された必須属性を持つコンポーネント) の `cc` 属性は `global` に設定されている必要があります。

◊ 必須の子 `b tu p d lu` に `f a` 属性が設定されている場合、それを削除または変更することはできません。

◊ 新規の必須の子 `b u p` コンポーネントを追加することはできません。

◊ 子 `b u p e` コンポーネントの `cd bo` 属性が `global` に設定されている場合、その属性を `bl` `pc` に変更することはできません。

◊ 子 `b u p e` コンポーネントの `cd bo` 属性が `el` に設定されている場合、`a` 属性を変更することはできません。

- global でないコンポーネントを持つパッケージがインストールされている場合は、ユーザがそのコンポーネントを [設定] で表示しようとすると、コンポーネントのコンテンツの代わりに「Component is not global (コンポーネントは global ではありません)」と表示されます。さらに、コンポーネントはコンポーネントの参照に含まれません。

- パッケージをインストールする組織で高度な通貨管理が有効になっている場合、`u ldp p` と `a` `ou u ldp p` を使用する Visualforce ページはインストールできません。
- Force.com AppExchange アプリケーションの一部として含まれる Apex は、いずれも累積テストカバー率が 75% 以上である必要があります。パッケージを AppExchange にアップロードすると、すべてのテストが実行され、

エラーがない状態で実行されていることが確認されます。テストは、パッケージがインストールされている場合にも実行されます。

- バージョン 16.0 以降、管理された Apex クラスを Visualforce コントローラとして使用している場合、登録者が次のメソッドとプロパティを使用できるように、これらのアクセスレベルを に設定する必要があります。
 - カスタムコントローラのコンストラクタ
 - 入力および出力コンポーネント用などの getter メソッドと setter メソッド
 - プロパティの get 属性と set 属性



ヒント: Apex クラスまたは Visualforce ページが翻訳を含むカスタム表示ラベルを参照している場合、その翻訳をパッケージに組み込むには、個々の言語を明示的にパッケージに含める必要があります。

Visualforce ページを含むパッケージが組織にインストールされている場合、ページは、ドメインではなく ドメインから配信されます。これにより、パッケージの悪意のあるコードがデータに影響を与えるのを回避します。

関連リンク

[カスタムコントローラおよびコントローラ拡張のテスト](#)

Visualforce ページとコンポーネントのパッケージバージョン設定の管理

Visualforce マークアップがインストール済み管理パッケージを参照する場合、Visualforce マークアップが参照する各管理パッケージのバージョン設定が、下位互換性を補助するために保存されます。これにより、管理パッケージのコンポーネントが次のバージョンのパッケージにアップグレードした場合にも、特定の既知の動作を行うバージョンにページが引き続きバインドされるようになります。

パッケージバージョンは、パッケージでアップロードされる一連のコンポーネントを特定する番号です。バージョン番号の形式は `majorNumber.minorNumber.patchNumber` (例: 2.1.3) です。メジャー番号とマイナー番号は、毎回のメジャーリリース時に指定した値に増えます。`patchNumber` は、パッチリリースにのみ生成および更新されます。公開者は、パッケージバージョンを使用して、後続のパッケージバージョンをリリースすることにより、そのパッケージを使用する既存の顧客の統合を分割することなく管理パッケージのコンポーネントを強化することができます。



メモ: パッケージコンポーネントと Visualforce カスタムコンポーネントの概念は大きく異なります。パッケージは、カスタムオブジェクト、Apex クラスとトリガ、カスタムページおよびカスタムコンポーネントなどの、多くの要素で構成されます。

Visualforce ページまたはカスタムコンポーネントのパッケージバージョンを設定する手順は、次のとおりです。

- Visualforce ページまたはコンポーネントを編集して、[バージョン設定] をクリックします。
- Visualforce ページまたはコンポーネントが参照する各管理パッケージの [バージョン] を選択します。より新しいバージョンの管理パッケージがインストールされても、バージョン設定を手動で更新しない限り、ページまたはコンポーネントではこのバージョンの管理パッケージが引き続き使用されます。インストール済み管理パッケージを設定リストに追加するには、使用可能なパッケージのリストからパッケージを選択します。

ページまたはコンポーネントに関連付けていないインストール済み管理パッケージがある場合にのみ、リストは表示されます。

3. [保存] をクリックします。

パッケージバージョン設定を使用する場合は、次のことについて注意してください。

- ・ 管理パッケージのバージョンを指定せずに、管理パッケージを参照する Visualforce ページまたはカスタムコンポーネントを保存する場合、ページまたはコンポーネントは、デフォルトで最新のインストールバージョンの管理パッケージと関連付けられます。
- ・ パッケージがページまたはコンポーネントにより参照されている場合は、管理パッケージの Visualforce ページまたはコンポーネントのバージョン設定を削除することはできません。管理パッケージの参照元を調べるには、[連動関係の表示] を使用します。

関連リンク

[Visualforce のバージョン設定方法](#)

[カスタムコンポーネントのバージョン設定の管理](#)

第20章

Visualforce ページでの JavaScript の使用

Visualforce ページで JavaScript を使用すると、幅広い既存の JavaScript 機能 (JavaScript ライブラリなど) や、ページの機能をカスタマイズするその他の方法を利用できます。 `e a upxcto cn<t` 、 `oa uipra F i pp` などの action タグは Ajax 要求をサポートしています。



警告: ページに JavaScript を含めることで、Visualforce を使用するときは発生しないブラウザ間やメンテナンスの問題が発生する可能性があります。JavaScript を記述する前に、既存の Visualforce コンポーネントでは問題を解決できないことを確認してください。

Visualforce ページに JavaScript を含める最適な方法は、静的リソースに JavaScript を配置してそこからコールする方法です。次に例を示します。

t el du rep c d g v p gai te Jigs w M l a t i R s . v y saia h " > /

これで、`c` タグを使用して、`s` ページ内のその JavaScript ファイルで定義されている関数を使用できるようになります。



ヒント: 式内で JavaScript を使用する場合はバックスラッシュ (\) を使用して引用符をエスケープする必要があります。次に例を示します。

JavaScript を使用したコンポーネントの参照

どの Visualforce タグにも 属性が含まれています。他のタグでタグの 属性を使用することにより、その 2 つのタグをバインドすることができます。たとえば `u b lp e p rox` タグの `af` 属性を `u ldp p` タグの `ap` 属性と併用することができます。`e a upxcto cnkt`、`oa ujra F i pp`、およびその他の活動指向型のコンポーネントの `ap` 属性および `af` 属性もその他のコンポーネントの `ap` 属性の値を使用します。

この ID は、複数の Visualforce コンポーネントのバインドに使用されるだけでなく、ページを表示する際のコンポーネントのドキュメントオブジェクトモデル(DOM) ID の一部の形成にも使用されます。

JavaScript または他の Web 対応言語で Visualforce コンポーネントを参照するには、そのコンポーネントの 属性の値を指定する必要があります。DOM ID はコンポーネントの 属性とその要素を含むすべてのコンポーネントの 属性の組み合わせで構成されます。

コンポーネントアクセスの例

次の例では、`<input type="checkbox">` タグの DOM ID を使用します。このページには 2 つのパネルがあります。最初のパネルには DOM イベントをトリガするチェックボックスがあり、2 番目のパネルにはイベントに応じて変わらいくつかのテキストが含まれています。

ページの上部には、`<input type="checkbox">` HTML タグ内に含まれる JavaScript があります。イベント (`input`) をトリガした要素および影響を受けるテキストを含む対象のパネルの DOM ID (`textid`) を引数としてとります。

```

et e dg p p g h < a ia="Pa>
e A t n l m u on prof< t! nf-dg o hi f i a . >-
t n en u n t o n t g zh u d ip & i ( , i )
t n e { u fd ph c i ki ( . ) gf byldh i sw y . i = " ";
e et atde o ut n tag tEx netdE ed In t .o n gf yro kli m sw y . i = " ";
e l s
e et atde o ut n tag tEx netdE ed In t .o n gf yro kli m sw y . i = " ";
t r c p < s i/ >
t te nou e h l T bl rp< nfc Pag psa h i , ss a i
ext b et d lc d e tk fe ho e Mi Ir$ n fng,o Dm DO p a . >-
xt te nou tu b pl qu bl< o a c Pa k ya=" " >
e l bl xo tb d h tc < e ac th atbo=" t keng "h>io libsl f : a < / a>
t n e u dt e bpo d h x < ib lo i h="k y=" " >
x t n e ol nt t n c c g h t ne ho ionn=" C} $ h i s , ! ' . Pa ;" ')> /
xt te nou u b p < /a Pa>
t te ne t hte l T p m h dn o Pa s a , a i sa
xt te we e llnbh d g dn ia a . >-
xt te nou ue b dn pt < l h aou bl o Pa c Pa k ya=" " >
n et w tg kmo g h a y i !
xt te nou u b p < /a Pa>
g p p < /a a

```

式は、コンポーネントによって生成される HTML 要素の DOM ID を取得するために使用されます。`$Component` グローバル変数を使用すると、Visualforce コンポーネント用に生成される DOM ID の参照が簡略化され、ページ構造全体での連動関係の一部が削減されます。

関連リンク

[コンポーネント ID へのアクセスのベストプラクティス](#)
[\\$Component](#)

Visualforce での JavaScript ライブラリの使用

Visualforce ページに JavaScript ライブラリを含めると、これらのライブラリが提供する機能を活用できます。JavaScript ライブラリを含める場合、静的リソースを作成してから、ページに `du rp c < コンポーネント` を追加してライブラリを含めるのが最適な方法です。

たとえば、MooTools (<http://mootools.net/>) を使用する場合、
を作成し、次のようにページ内で参照します。

という名前でライブラリの静的リソース

```
en t el du rep c el m < tp a a>
g p p < /a a>
```

その後、ライブラリから関数をコールする `c` を追加することで、`i` ページ内でその静的リソースを使用できます。

```
en t el du rep c el m < tp a a>
r c p < s i >
n en u n t o n t e a t g(x) u d i p E i ( , i )
t e n e e t t x f d e t ph o e d n i t k i ( o $ . g f b l d h i s w y . i = ' ; '
e t t x l e t e d n k o $ n g f r o h i m s w y . i = " ; a
}
r c p < s i / >

tn t or n h g u C o <1 h > ai <a> >
t te nou tu pl qu b l< o a c Pa k ya=" " >
e l b l ro tb f d t c < e d c t a t h o = t k e n g " h > io libsl f : a < / a>
t n e u d e b p o d h x < ib lo j h = t " k y = " "
n e o l n t h c c g e h t n e h o i o m n = C } g $ h i s , ! ' . Pa ; " ') > /
x t t e nou u b p < /a Pa>
t te not i m e t l n T t o p t b w e - i s h k l n b h a t e n i d s a g d s e h b i a o d h d a k c i s
}
t te no eut u e b d n p t < l l h a e u b l n o Pa = " c g k m P a k ya = " > a !
x t t e nou u b p < /a Pa>
g p p < /a a>
```

Visualforce ページで JavaScript ライブリを使用し、そのライブリで `\` が特殊文字として定義されている場合、JavaScript を変更して、この特殊文字としての使用を上書きする必要があります。たとえば、jQuery を使用する場合、`o ol` 関数を使用して `\` の定義を上書きできます。

```
n t el du rep c el m < p a a>
l m h < >
d h < > a
r c p < s i >
n o ol C f c Q y i ; ( )
e t e j n d o u t m r h d n u Q o f c y ( . ) a y ( i ( )
t l w l l t o r d d r f c k " y " a i ( , a ! ; )
}
r c p < s i / >
g p p < /a a>
```

 **メモ:** ExtJS コンポーネント `edemp h` `de a`、`rg al` `lk a id s &C` `T`、または `s a` を使用するページでは、バージョン 3 より前の ExtJS バージョンを使用しないでください。

Apex コントローラの JavaScript Remoting

JavaScript から Apex コントローラのメソッドをコールするには、Visualforce の JavaScript Remoting を使用します。これにより、AJAX 機能を実装した標準 Visualforce コンポーネントでは実現できない、複雑で動的な動作を行うページを作成できます。

JavaScript Remoting は、次の 3 つで構成されています。

- JavaScript で記述される、Visualforce ページに追加するリモートメソッドの呼び出し。
- Apex コントローラクラスのリモートメソッド定義。このメソッドは Apex で記述されますが、通常の action メソッドとはいくつかの違いがあります。
- JavaScript で記述される、Visualforce ページに追加または含めるレスポンスハンドラコールバック関数。

Visualforce ページへの JavaScript Remoting の追加

Visualforce ページで JavaScript Remoting を使用するには、要求を次の形式の JavaScript 呼び出しとして追加します。

```
    namespace ] controller method . . .
    rm p ] [ a a s ... ,
    callbackFunction
    onrugc o f ] [ i i a
    ; )
```

m c p はコントローラクラスの名前空間です。組織に名前空間が定義されている場合、またはクラスがインストール済みパッケージに基づく場合は必須です。

or l d r c は Apex コントローラの名前です。

mo . h はコールする Apex メソッドの名前です。

r rm p はメソッドが取るパラメータのカンマ区切りのリストです。

t lab n a oc c はコントローラからの応答を処理する JavaScript 関数の名前です。匿名関数をインラインで宣言することもできます。a oc c k ではメソッドコールの状況と結果をパラメータとして返します。

onrugc o f は、リモートコールと応答の処理を設定します。Apex メソッドの応答をエスケープするかどうかを指定するなど、リモートコールの動作を変更する場合にこれを使用します。

リモートメソッドコールは同期して実行されますが、応答が返されるのを待機しません。応答が返されると、コールバック関数は非同期で応答を処理します。詳細は、「[リモート応答の処理](#)」を参照してください。

JavaScript Remoting 要求の設定

Remoting 要求の宣言時に構成設定を使用してオブジェクトを指定することで、Remoting 要求を設定します。たとえば、デフォルトの設定パラメータは次のようにになります。

```
ret u f#: tr uc : pom } , s a i 3 0000
```

これらの設定パラメータに順序はありません。また、デフォルトから変更の必要がないパラメータは省略できます。

JavaScript Remoting では、次の設定パラメータをサポートしています。


```
r      c      <    s i/   >    //..
```

A u n d i cc を一度設定すると、すべての JavaScript Remoting 要求で OAuth が使用されます。残りの JavaScript Remoting コードには変更は必要ありません。

A u n d i cc は、ページのコードで取得される OAuth 認証トークンです。アクセストークンの取得と更新は簡単な OAuth で直接行われ、追加部分が 1 つあります。JavaScript Remoting の OAuth 認証では "visualforce" スコープが要求されるため、このスコープを使用するか、これを含むスコープ ("web" や "full") を使用してトークンを生成する必要があります。OAuth 要求で、`ic ro p fcc` (あるいは "web" または "full") と設定します。

アクセストークンの取得および Force.com プラットフォームでの OAuth の使用についての詳細は、Salesforce オンラインヘルプの「リモートアクセスアプリケーションの認証」および

[ed l er ro kee pfc g it ggpfr A o e. mpD / ao Ado 2 o imc _c o](#) を参照してください。

名前空間および JavaScript Remoting

特にパッケージで提供されるメソッドに Remotoing コールを行うページで名前空間との連携をより簡単に使うため、`o c $` グローバル変数を使用して、リモートアクションの正しい名前空間(ある場合)を自動的に解決できます。この機能を使用するには、JavaScript Remoting を明示的に呼び出す必要があります。これを実行するパターンは次のとおりです。

```
th ro er no enc ngt r giMbo a o . d k a av. i ( 
  fully_qualified_remote_action
  invocation_parameters
); )
```

完全修飾リモートアクションは、`namespace BaseClass] ContainingClass] ConcreteClass Method` のように、名前空間、ベースクラスなどを含むリモートアクションメソッドへの完全なパスを表す文字列です。名前空間を自動的に解決するには、たとえば、`o c $` のように、式に

Anon m o c \$ を使用します。

呼び出しパラメータは、リモートメソッドの呼び出しの実行に使用される引数であり、標準リモートコールを行うために使用される引数と同じです。

Anon m o c メソッドに送信するためのパラメータ(ある場合)。

- 返される結果を処理するコールバック関数。
- 呼び出しに関する設定詳細(ある場合)。

たとえば、次のような取引先を取得するリモート呼び出しを定義します。

```
e tr tx c t p jpr < s ic py" v / saia " >
tn e ut o gAno m ou cc i R ( )
te n eou et cte o ut mnv dgE a mden e a x cc l udn y ( 'a S av .') a
e e th ro er no enc ngt r giMbo a o . d k a av. i (
e t t Anotme n e t o$ latMAGh ! ou i cc . y . ,
te n ou ccm a N a,
t en n etue {o t fia i v (s v, )
ee t tnt{ f u i v (s sa )
e et etde o ut mn dg Be m d E nn deI crtI r luMyd (T 'a ii.H) L = s .
e et etde o ut mn dg Be m d E nn I e mcc rte r lyM ( TAm ai.H) L = s .
e ee t etn e e x tf { n p soi cv ( p . y == ' i ')
e et etde o ut mn dg Be m d E r eoI rrnmoe e p E et ne y" ( s M s Th sg i.H) L =v .
e e{ l s
e et etde o ut mn dg Be m d E r eoI rrnmoe e p E et ne y" ( s M s Th sg i.H) L =v .
e } ss
} ,
```

```
    } r uc : p ; )  
r c p < s i/ >
```

この JavaScript Remoting コールでは、コントローラが定義されている名前空間、自分の名前空間内にあるか、インストールされたパッケージが提供する名前空間内にあるなどの詳細を把握している必要がありません。また、組織に定義済みの名前空間がない状況にも対応します。

メモ: このコール時に発生したエラーは JavaScript コンソールでのみレポートされます。たとえば、`o c$` で複数の名前空間に一致する `o c` メソッドが見つかった場合、最初に一致したメソッドを返し、JavaScript コンソールに警告を記録します。一致するコントローラまたはアクションが見つからない場合は、そのコールはエラーを表示することなく失敗し、エラーは JavaScript コンソールに記録されます。

リモートメソッドの宣言

コントローラでは、Apex のメソッド宣言は、次のように `o c` アノテーションが先頭に付加されます。

メソッドでは、引数として、Apex プリミティブ、コレクション、型指定された `sObject`、汎用 `sObject`、ユーザ定義された Apex クラスおよびインターフェースを取ることができます。汎用 `sObject` では、実際の型を特定するために `ID` または `sObjectType` の値を指定する必要があります。インターフェースパラメータでは、実際の型を特定するために `apexType` を指定する必要があります。

メソッドでは Apex プリミティブ、`sObject`、コレクション、ユーザ定義された Apex クラスおよび列挙、`luteeret`、`lutepl`、`vt`、`ReasU`、`l nse`、`Rece`、`pR` または `S offc` を返すことができます。

JavaScript Remoting に使用されるメソッドは、名前とパラメータ数によって一意に識別される必要があります。オーバーロードは不可能です。たとえば、上記のメソッドでは、`ng`一緒に `mt eI r Ig do u br npc` (メソッドを持つことはできません。代わりに、異なる名前で複数のメソッドを宣言します。

Apex メソッドは `lc` かつ `bl` または `a pc` のいずれかである必要があります。グローバルに公開されるリモートアクションで繊細な操作を実行したり、非公開のデータを公開したりしないようにしてください。リモートアクションは他の `a` メソッドのみをコールできます。`pc` リモートアクションはコンポーネントでは使用できません。`bl` 一方、スコープでは使用できます。`a` スコープのエスカレーションはコンパイルエラーになります。または、実行時に解決される参照の場合は、実行時エラーになります。次の表では、これらの制限を詳細に説明します。

@RemoteAction のス Visualforce Page コード	非グローバルコンポーネント	グローバルコンポーネント	iframe
グローバルリモート メソッド 使用可能	使用可能	使用可能	使用可能
公開リモートメソッド 使用可能	使用可能	エラー	エラー

コンポーネントを `<div>` タグまたは `` タグによって間接的に含まれるマークアップを介してリモートアクションにアクセスする場合、リモートメソッドのスコープは最上位コンテナ（スコープのエスカレーションルールに準拠する必要のある包含階層の最上位項目）に継承されます。

最上位コンテナ			
@RemoteAction のア クセス元	Visualforce Page	非グローバルコン ポーネント	グローバルコンポー メント
グローバルコンポー メント	使用可能	使用可能	使用可能
非グローバルコン ポーネント	使用可能	使用可能	非グローバルコン ポーネントが公開リ モートメソッドを含 まない場合にのみ使 用可能。
<code>l du p c m o m p p</code>	<code><同じ名前空間内では></code> なし <code><使用可能。名前空間 が異なる場合および 含まれるページまた はその子階層に公開 リモートメソッドが 含まれる場合は工 ラー。</code>	なし	エラー

リモートメソッドと継承

メソッドが検索またはコールされる場合、Visualforce ではページコントローラの継承階層を調べ、コントローラの上位階層のクラス内でメソッドを検索します。

この機能を示す例を次に示します。次の Apex クラスでは 3 階層の継承階層を形成します。

```
1 body n re h d tledcn o Kmaor d i R ss ai R
en et ent rte p o m or ldr C s PaR
2 body r lu n er ethleight rtie n oimorsldir aC ss a PaR
n e redt erndt rte p o mpor ldr G G a aR
3 body r lu n r e hlight erntde rte fa osmpor d d r ss a G a aR
t Ano m o c R i
4 t t l body en r t eg l h p r s dgi oao hs Hs ya T (i S )
e n r eu ll o lle to o h re oh +tm Th ' p ' G a a ;
}
```

この Visualforce ページでは簡単な

リモートアクションをコールします。

```
t en gorppl ipc tledcn o Kmaor ldar C = i R " >
et e tr tx c t p jpr < sic py v / saia " >
t n en ue o { fdl o ll o o h i Hs ya ( )
e et tledn oem or ldr C ll o tllRo on hte { ot Hs ya ( , i (s v,
ee t tnt{ fu i v ( s sa )
e et etde o ut mre dg B t m adE r nmu ei rt. r luMy" (T s " i H L = s ;
}
} ; )
t r c p < s i/ >
```

```

e n b n u e o l e c l l c < d u J k t t i l n e & s u y & br ( ' ; " & H S y a < / < > > /
e d t e d r t l u l u & & i ] = " s " > R s [ < s & / >
g p p < / a &

```

リモートメソッドはクラス内には存在しません。代わりに、
クラスから継承されます。

インターフェースパラメータによるリモートメソッドの宣言

具象クラスに制限するのではなく、インターフェースパラメータと戻り値のデータ型を使用して
メソッドを宣言できます。これにより、たとえば、パッケージプロバイダは、リモートメソッドと関連付けられた
インターフェースをパッケージ化できます。登録者組織は、パッケージ化されたインターフェースを実装する
独自のクラスに渡すことによって、Visualforce ページからコールできます。

次に、簡単な例を示します。

```

e h d t l t e f p c o m o r l d r C i ss R
e t e n b l e t r e f p c t r e f M I t r i f ng g r a y M a i s y i s ;( )
e t b l e e p d t c r e M e n m i p r s s M I y s s i f a s y i s ;
e t t r n t r n p g r m g & a i s y i s ;
t e b l u t t r { p e g g t r M g r i u n i s g y i s ( ) y i s ;
e b l u t d t p c { r n M g n v i r m s } g y i s ( i s s ) y i s = s;
}

t t A n o m o c R i
t t b l e t e n e p c e c M e l t e n f { s g M i r a M I s s a ( y i a ) }
t t e l w n M n C l C M C y s s a y = y s s a ;( )
t t m m r e C M l e M C t y s t f y i s n g ( r y s s a ; y s y a i ' . + y i s ( ) " . ; "
n r r u m C y

```

JavaScript Remoting コールからインターフェースパラメータを宣言するに送信されるオブジェクトは
値を含む必要があります。この値は、具象クラスへの完全修飾パス、つまり、
namespace BaseClass] ContainingClass] ConcreteClass である必要があります。たとえば、上記のコン
トローラへの JavaScript Remoting コールを実行するには、次のコードを使用します。

```

t t h r o e r n o & c n g t r g i a M o a o . & k a & i v . i ( 
t t e & n d m t e n e o & m o e l d r ! & } i g R M s s a , ' 
x e t e & e e p t & h e n o m c p a l d y r C t l ' s a n M & r m g u o m : y s s a , ' y i s L ' ' s !
n t d l h u a R s ; )

```

組織内にクラス定義がある場合は、Remoting コールを単純化し、デフォルトの名前空間も使用できます。

```

e n e o e m o e l d r C g R M s s a ( 
x e e t t e n p o & f o r l d r C t l ' a y n M & r m g u o m : y s s a , ' y i s L ' ' s !
n t d l h u a R s ; )

```

リモート応答の処理

リモートメソッドコールに対する応答は、リモートメソッドコールで提供されているコールバック関数によって
非同期に処理されます。コールバック関数は、リモートコールの状況を表すオブジェクト、およびリモー
トメソッドで返されるオブジェクトをパラメータとして取得します。関数は返されるデータに基づいてページの情報およびユーザインターフェース要素を更新できます。

オブジェクトは、リモートコールの成功または失敗に対応するうえで役立つ値を提供します。

- `success` は、成功のときは `error` のときは `failure` になります。
 - `isSuccess` は応答の種別です。成功したコールは `true`、リモートメソッドが例外を返した場合は `false` のようになります。
 - `message` には、返されたエラーメッセージが含まれます。
 - `response` には、リモートメソッドにより生成された場合は、Apex スタック追跡が含まれます。
- e string または number など、によって返される Apex プリミティブデータ型は対応する JavaScript に変換されます。返される Apex オブジェクトは JavaScript オブジェクトに変換され、コレクションは JavaScript 配列に変換されます。JavaScript は大文字と小文字を区別することに注意してください。そのため、`A`、`a`、`A` および `a` は異なる項目であるとみなされます。

JavaScript リモートコールの一部として、Apex メソッド応答に同じオブジェクトに対する参照が含まれる場合、そのオブジェクトは返される JavaScript オブジェクトには複製されません。代わりに、表示される JavaScript オブジェクトには同じオブジェクトへの参照が含まれます。たとえば、同一オブジェクトを 2 回含むリストを返す Apex メソッドです。

デフォルトでは、リモートコールの応答は、30 秒以内に返される必要があります。これを超えると、コールはタイムアウトになります。要求の完了にこれ以上の時間を必要とする場合は、長いタイムアウトを 120 秒以内で設定します。

リモートコールの応答の最大サイズは 15 MB です。

JavaScript Remoting コードがこの制限を超える場合は、次のように対応できます。

- 各要求の応答サイズを削減する。必要なデータのみを返します。
- 大量データの取得を、小さなチャunkを返す複数の要求に分割する。
- バッチ以外の要求を使用していることを確認する。`{Remoting 要求の設定プロックで}` と設定します。
- バッチ要求の使用頻度を抑え、バッチサイズを削減する。



メモ: JavaScript Remoting を使用時に開発中の JavaScript コンソールを開いたままにします。JavaScript Remoting で発生するエラーや例外は、JavaScript コンソールが有効化されている場合はそれに記録され、有効化されていない場合は無視されます。

プログラミングエラーまたはその他の失敗により `MethodException` メソッドで例外が発生すると、Apex スタック追跡がブラウザに返されます。JavaScript デバッガコンソールでスタック追跡を検査するか、応答コールバック関数のエラー処理でスタック追跡を使用します。次のコールバック関数では、例外がある場合にスタック追跡を表示します。

```

e tr tx      c t p jx < s ic     py"      v / saia " >
tn e ut o    ff gAno m ou cc i R      ( )
te m         eou et cnde o ut mnv dgeB a madEN e a n cc l udn      y ( 'a S av .') a
e e          tli ro er no ec ngtr giamo a o . d k a iv. i ( ,
e t t        Anotme n e to $ latMAGh !原 ou i cc . y . , '
te n         ou ccm a N a,
t e n etue {ot fca i (s v, )
ee t tnt{   f u i v ( s sa )
e et etde o ut mn dg Be m dE nndeI crtI r luMyd (T 'a ii.H) L = s .
e et etde o ut mn dg Be m dE nnIe mcc rte r lyM ( Tm ai.H) L = s N ; a
e e }ee t etn e e x tf { n p soi cv ( p . y == ' i ')
e et etde o ut mne dg Be m dE r eoI rrnmoe . p Er y" ( s Ms T s' i.H) L =
e e et ne     re g obre ter enw + e. ss r a" hr / > v p+ . < "
e e { l           s
e et etde o ut mne dg Be m dE r eoI rrnmoe e p Et re y" ( s Ms Th s' i.H) L =v
e e }
}
;
```

r c p < s i / >

JavaScript Remoting および `<apex:actionFunction>`

通常、`<apex:form>` タグ内に `<apex:actionFunction>` を使用すると、JavaScript によりコントローラアクションメソッドをコールすることもできます。この 2 つの相違点の一部を次に示します。

通常、`<apex:form>` タグ内に `<apex:actionFunction>`

- ◊ ユーザが再表示ターゲットを指定できる
- ◊ フォームを送信する
- ◊ JavaScript の記述は不要

- JavaScript Remoting:

- ◊ ユーザがパラメータを渡すことができる
- ◊ コールバックを提供する
- ◊ 一部の JavaScript を記述する必要がある

通常、`<apex:form>` の方が使いやすく、必要なコードも少なくなります。一方、JavaScript では、より高度な柔軟性を提供できます。

JavaScript Remoting の例

Visualforce ページでの JavaScript Remoting の使用方法の基本的なサンプルを次に示します。まず、次のように Apex コントローラを作成します。

```
1 body n r e h l e n t ou d e m s i a s s a R
t bln ter{ n ep cg t e u t ccm gi is a N a ; s ;
t t bln t An p t e o m te cdu ac gs i a a ; s ;
e bln { ou p e o c m t nti m o r R ip (c ) // y s
t An o m o c R i
t t l bo g et A n d u An cg au ter f ea i gaou ccm (is a N a )
t n ou e cc e d e n E C m o T a h e h S r m p N m a , H P p y N O y s
t An e ou M cc te nne E F a u W : c c h R ] N a N a
n rt rm ou cc a ;
}
```

Ano m o c アノテーション以外、これは他のコントローラ定義と同じように見えます。

このリモートメソッドを使用するためには、次のような Visualforce ページを作成します。

```
t en g o r e p t d a n t su o e m a = " R " >
e t e tr tx c t p jpr < s ic py=" v / saia " >
e t en e ut o t f g A n o m ou cc i R ( )
te m e ou et c a n o ut mnv d g e B a m a d E e a h cc l u h y ( ' a S a v . ) a
e e tlu ro e r n o a c n g t r g i a M o a o . a k a a v . i ( ,
e e t An o e t a n d e d g o r m a n l R g ou i cc R .
te n ou cc a N a ,
t en n e u e { o r f n a i ( s v , )
e e t t n t { f u i v ( s s a )
e t r a M b e d e f e M u / a n D o l t s H f a n l v a s h a k s i = i s .
e t et de o ut mne d g e B t m d E r a o m n d e . c r t I r y Y u M d T
e t et de o ut mn d g B m d E I . y (
```

```

        {
            e      e      t n n o   oen    blCt oe blp $n e d r' o b t id     eek cm   Iu   l)mcn is   Es   p   .
            e      e      nn e   rt e   r   lueM   bTr m   i .Hm L   H = ps   N   O   ys;
            e      e      }ee   t et n   e   x tf {   n   p soi cv ( p . y === ' i   ')
            e      e      etde o ut   mne dg Be   m dE r   eo I rrnmoe . p E r   y" ( s M s T   s' i .H) L   =
            e      e      etde o ut   mne dg Be   m dE r   eo I rrnmoe e p E et   ne y" ( s M s T   s' i .H) L   =v
            e      e      etde o ut   mne dg Be   m dE r   eo I rrnmoe e p E et   ne y" ( s M s T   s' i .H) L   =v
            }
            e      e      et      r uc   :p   ,   s   a
            ;
        )
    }
    t      r      c      p      <   s i/   >
    n      et   ud et   e pr txcc   < di   i p"   a s'a   y"   "   > /   "
    n      bneuet   olt   tcgao   oukt   Antti   un bR xc o   "( ) G   <   /   >
    e      ed   dn r   o rr ro   pdE   i = s   s   s" > i />
    x      ex   B   lgo   pd   lpl   o   <   k a   a k i   "   >
    ex   ex   tB   lgn   pe   potd   b&   nka   ncalocu   hci   "   2   s   "   s=" " >
    ex   ex   tBe   lga   p   poect   dm   k a   a fm   ls   i   "   i   s   "   >
    ex   ex   enet   d   r A   opm   d   sc   ls   ia="   "   > /
    ex   ex   tBe   lga   p   poc   <c   m   k/a   a   ls   >
    ex   ex   tBe   lga   p   poc   <c   m   k/a   a   acm   ls   i   "   s   "   >
    ex   ex   xtet   tx   ou   t   pdee   p   &T   lmcm a   E   pi   "   s   "   >
    ex   ex   tBe   lga   p   poc   <c   m   k/a   a   ls   >   y"   > /
    ex   ex   B   lgo   p   poc   <c   k/a   a   ls   >
    ex   ex   g   p   p   <   /a   a   >

```

このマークアップについては、次の点に注意してください。

- JavaScript は、**明示的な** `call` Remoting コールを使用し、`c$` グローバル変数を活用してリモートアクションメソッドの正しい名前空間を解決します。
- `u` 変数は、`var` コールが成功した場合にのみ `undefined` になります。この例で示したエラー処理は意図的に単純にしてあり、エラーメッセージとスタッタ追跡が、`temperr` 値と `v_h.s` 値からそれぞれ出力されます。要求でメソッドコールが成功しない場合は、これに代わるさらに強固なロジックを実装することをお勧めします。
- `t` 変数は、Apex の `success` メソッドから返されるオブジェクトを表します。
- ブラウザ HTML 要素の DOM ID へのアクセス方法は単純で、項目の ID を使用するだけです。
- ID が一意になるようにするために、Visualforce コンポーネントの DOM ID は動的に生成されます。上記のコードでは、`c_p` グローバル変数を介してコンポーネントにアクセスすることによってその ID を取得するために、「[JavaScript を使用したコンポーネントの参照](#)」で説明した方法を使用しています。

第 21 章

ベストプラクティス

Visualforce ページでは次のベストプラクティスを使用できます。

- Visualforce のパフォーマンス向上のためのベストプラクティス
- コンポーネント ID へのアクセスのベストプラクティス
- 静的リソースのベストプラクティス
- コントローラおよびコントローラ拡張のためのベストプラクティス
- コンポーネント facet の使用のためのベストプラクティス
- ページブロックコンポーネントのベストプラクティス
- PDF を表示するためのベストプラクティス
- lb r p p のベストプラクティス >a

n

Visualforce のパフォーマンス向上のためのベストプラクティス

Visualforce は、標準の Salesforce ページの機能、動作およびパフォーマンスに合わせる機能を開発者に提供するために設計されました。遅延、予期しない動作や、その他の問題(特に Visualforce に関するもの)がある場合は、いくつかの対処法を実行することにより、操作性を改善できるだけでなく、コーディングの改善にも役立てることができます。

まず、次を確認して、Visualforce に問題があるかどうかを特定します。

- 予測される Visualforce の機能を他のマシンや他のブラウザを使用してテストし、その問題が 1 台のユーザのコンピュータに限らないことを確認する。
- 他の Salesforce ページの読み込み時間を確認して、読み込みに時間がかかることがネットワーク上の問題ではないことを確認する。他のページの読み込みにも時間がかかる場合は、Salesforce における帯域幅や待ち時間の問題が原因である可能性があります。Salesforce サーバの状況を確認するには、trust.salesforce.com を参照してください。また、ネットワーク接続の状況をチェックして、適切に機能しているかどうかを確認する必要があります。
- JavaScript および CSS の縮小、Web 画像の最適化、できる限り iframe の使用を避けるなど、一般的な Web 設計のベストプラクティスに従っていることを確認する。
- 開発者コンソールを使用して、要求をステップごとに実行し、要求内のどの項目がシステムリソースを最も消費したかを調べる。Salesforce オンラインヘルプの「開発者コンソールの使用」を参照してください。

次のリストは、よく発生する Visualforce のパフォーマンス上の問題と解決策を示したものです。

ビューステートのサイズ

Visualforce ページのビューステートのサイズは、135 KB 未満である必要があります。ビューステートのサイズを縮小することにより、ページをより迅速に読み込み、表示が停止する頻度を削減します。

開発モードフッターの [ビューステート] タブを使用し、次の対処法を実行すると、ビューステートのパフォーマンスを監視できます。

- 状態の維持に不可欠ではなく、ページの更新時にも不要な変数には、Apex コントローラでキーワードを使用する。
- ビューステートの大部分をコントローラまたはコントローラ拡張で使用されているオブジェクトから取得していることが分かった場合は、Visualforce ページに関連するデータのみを戻すように SOQL コールの絞り込みを検討する。
- ビューステートが大規模なコンポーネントツリーの影響を受けている場合は、ページが依存しているコンポーネント数の削減を試みる。

読み込み時間

サイズが大きいページは読み込み時間に直接影響します。Visualforce ページの読み込み時間を改善するには、次の対処法を実行します。

- アイコンの画像など頻繁にアクセスするデータをキャッシュする
- Apex コントローラの getter メソッドで SOQL クエリを使用しない
- 次の対処法を実行して、ページ上に表示するレコード件数を削減する
 - Apex コントローラで SOQL コールから返されるデータを制限する。たとえば、`E E句で Wス H R` テートメントを使用したり、`の結果を削除`したりします。
 - リストコントローラでページネーションを活用して、ページあたりの表示レコード件数を削減する。
- Apex オブジェクトを遅延読み込みして要求時間を削減する

タグ外に JavaScript を移動し、`g p p` 終了タグのすぐ前の `ca` タグに配置することを検討する。`du r p c c < タグは、JavaScript を閉じ要素 h のすぐ前に配置します。つまり、Visualforce ではページ上のその他のコンテンツの前に JavaScript が読み込まれます。ただし、ページにマイナスの影響を及ぼさない確信がある場合は、JavaScript をページ最下部に移動するだけにしてください。たとえば mr c またはイベントハンドラを必要とする JavaScript コードスニペットは、h 要素に配置したままにする必要があります。`

どの場合でも、Visualforce ページは 15 MB 未満である必要があります。

複数の同時要求

同時要求は他の保留中のタスクをブロックする可能性のある実行に長時間かかるタスクです。これらの遅延を短縮する対処法は、次のとおりです。

- `o l@prc <` で使用する action メソッドを軽量化する。これは、`o l@prc < から a ia` コールされる action メソッドで、DML の実行、外部サービスコール、およびリソースを大量に消費する他の操作を回避するためのベストプラクティスです。指定した間隔で `o l@prc < から繰り返し` コールされる action メソッドの影響を考慮します。特に、広範囲にわたって配布されたり、継続的に開かれたりするページで使用する場合には注意が必要です。
- Visualforce ページから Apex をコールする間隔を延長する。たとえば、`o l@prc < コンポーネントの使用時は、属性を 15 ではなく 30 秒に調整します。`
- Ajax を使用して不要なロジックを非同期コードブロックに移動する。

クエリおよびセキュリティ

Apex コントローラの作成時に `h` `q` キーワードを使用して 1 人のユーザのデータセットを表示するだけで、SOQL クエリを向上できる可能性があります。

ページに項目値をすべて表示する

ページに大きなテキストエリア項目など多くの項目があり、他のエンティティとの主従関係がある場合、Visualforce ページに返されるデータのサイズに関する制限およびバッチ制限のためデータをすべて表示できないことがあります。ページには、次の警告が表示されます。「You requested too many fields to display. Consider removing some to prevent field values from being dropped from the display. (要求した項目が多すぎて表示できません。項目値をすべて表示するには項目値の一部を削除することを検討してください。)」

ページに項目値をすべて表示するには、一部の項目を削除して返されるデータ量を削減します。または、コントローラ拡張を独自に作成して、関連リストに表示される子レコードをクエリします。

コンポーネント ID へのアクセスのベストプラクティス

JavaScript または他の Web 対応言語で Visualforce コンポーネントを参照するには、そのコンポーネントの 属性の値を指定する必要があります。DOM ID はコンポーネントの 属性とその要素を含むすべてのコンポーネントの 属性の組み合わせで構成されます。

グローバル変数を使用すると、Visualforce コンポーネント用に生成される DOM ID の参照が簡略化され、ページ構造全体での連動関係の一部が削減されます。特定の Visualforce コンポーネントの DOM ID を参照するには、ページのコンポーネント階層の各レベルを区切るドット表記を使用して、コンポーネントのパス指定子を `o` `p` `$` に追加します。たとえば、Visualforce コンポーネント階層の同じレベルにあるコンポーネントを参照するには `c` `p` `$itemID` を使用し、完全なコンポーネントパスを指定するには `c` `p` `$grandparentID.parentID.itemID` を使用します。

パス指定子は、コンポーネント階層と次のように照合されます。

が使用されているコンポーネント階層の現在のレベルでまず照合されます。

- 次に、一致が検出されるか、コンポーネント階層の最上位レベルに達するまで、コンポーネント階層の各上位レベルが照合されます。

バックトラッキングはないため、ID の照合で上に移動してから下に戻る必要がある場合は、一致しません。

次の例に、`c` `p` `$` のいくつかの使用方法を示します。

```

g p p < a a>
1 < s y>
e r bcrdr kld: apt runro o r : s 99 # ; s i ; i-a ;
e n r m ge n didwt e g e ptm t xi a e . emt lm à g } ; ri l c ; o i-a ;
1 < s / y>

ro d m p fro in < a i = F " >
B t d go pd p te k g k al oar wi g'ke g t e a n l o i o n = C a p $ s i
e tB l g a t p o d s n k aoh ac s i = s " >
e ex e tB e t g a p e t p o c t s m n k a o a c s I i = s " >
e A t e A t e l e t r t h re o t n n f o d m c a s p i s .
e e e e l i e h p e r f t b M e I n & > m D g D O b r s h s : s k i i i < s > /
e t e t d p ro e t e B t l g a t e n t e _ d h o n c k o h P a c m I s p s < > /
e ex e tB e t g a p p o c < c m k / a a s >
```

```

e   e   w e t e   t e t n   eh oht n a mo lom i c " p   s   " >-
e   e   x t t e   nou tu   pl   etqu bl< o   ed lc   lPa &r ya=b c " s   yk ss=a i   "
e   e   ne {tol   d rc t e n ot om t G i p h ac ! I   .   s   ;" >
e   e   t e e r l   r c c h   kE s i
e   ex   x t t e   nou u   b   p   <   /a   Pa>
e   ex   e   B   l g o p   p o c <c   k/a   a   s   >
e   ex   t t B   lBgo t p u < d B& t ik a h i t a   o b si o=" m   s"   i a=" " >
e   e   to b t te   cnouk h ! le-n $p e n t s a t i s   n p   P a h s   a a a a " (   s
e   ex   n   tn e t o o m m e n t h o h i c m I i   "   s   " >-
e   ex   nou tu   pl   etqu bl< o   ed lc   lPa &r ya=b c " s   yk ss=a i   "
e   ex   ne {tol   d rc t e n ot o k e t G i p h n ac (! d h c m .I   s   .   s
e   ex   nee d l   c r c c h   k s i
e   ex   nou u   b   p   <   /a   Pa>
e   ex   e   to b t te   cnouk h ! tl t-e s p d t n h s e t i s   ro   c P a o s   a n i   a   s   " (   F
e   ex   n   tn e t o o m m e n t h o h i c m I i   "   s   " >-
e   ex   nou tu   pl   etqu bl< o   ed lc   lPa &r ya=b c " s   yk ss=a i   "
e   ex   ne {tol   d rc t e n o o k m e t Bi p l a g e n t e a t ! e & h n c k.} oh Pac m .I   s   .   s
{   e   t e n o e d m e C   r o p e & t l g a t e n t e & h F c k.} oh Pac m .I   s   .   s
e   ex   x t t e   nou u   b   p   <   /a   Pa>
e   ex   t t B   lBgo p u p o c <   k/a   a   s >
ex   B   l g o p   p o c <   k/a   a   >
e   e   to b t te   cnouk h ! l -t n s p b l s a i g s   B   l g o P a s   s a i   " k   Pa   " ,
e   ne   ed e e t   c p t o f l   m t c t   a p r il o i   m p b l   h D q   a   s a i   >-
x t t e   nou tu   pl   etqu bl< o   ed lc   lPa &r ya=b c " s   yk ss=a i   "
e   ex   ne {tol   d rc t e n o o k m e t Bi p l a g e n t e a t ! e & h n c k.} oh Pac m .I   s   .   s
t   e e & u l   h c c h   F   i
x t t e   nou u   b   p   <   /a   Pa>
r   h   <   >   /
t   n w   e o   tro b t te   &cnouk h--nled 'o p tr so d i s   p   Pa h p s   v i   a a
t   t e n   l d h n   b l e   m o g n o   o m m a c   r o s   s a i   s   >-
x t t e   nou tu   pl   etqu bl< o   ed lc   lPa &r ya=b c " s   yk ss=a i   "
e   ex   ne {tol   d rc t e n o o k m e t Bi p l a g e n t e a t ! I   .   s   ;" >
t   new   e t o   r o   r   Wh k   ' i
x t t e   nou u   b   p   <   /a   Pa>
x   ro   m p f   <   /a   >
g   p   p   <   /a   a

```

一意の ID の使用

コンポーネント はページの各階層セグメント内で一意である必要があります。ただし Salesforce では、参照する必要があるすべてのコンポーネント、およびその参照に必要なコンポーネント階層の上位コンポーネントに対して、ページで一意の を使用することをお勧めします。

たとえば、1 つのページに 2 つのデータテーブルがあるとします。両方のデータテーブルが同じページブロック内に指定されている場合は、両方のデータテーブルの 属性が一意である必要があります。それそれが別のページブロックに指定されている場合は、同じコンポーネント を付与することができます。ただし、この場合、特定のデータテーブルを参照できる唯一の方法として、すべてのコンポーネントに を割り当ててから、*i* Visualforce で自動的に参照するのではなく、コンポーネント階層を使用してデータテーブルのコンポーネントを参照する必要があります。また、ページ階層に変更があると、プログラムが機能しません。

コンポーネント ID での反復

テーブル、リストなどの一部のコンポーネントでは、レコードのコレクションの反復をサポートしています。これらのタイプのコンポーネントに ID を割り当てると、コンポーネントの各反復に、最初の ID に基づいて一意の「複合 ID」が割り当てられます。

たとえば、次のページには `h` に設定された ID を含むデータテーブルがあります。

t	e	n	t	e	g	n	d	p	o	r	h	n	C	e	u	r	s	c	a	d	a	n	c	e	t	s	d	c	"	v	g	h	S	=	'	a	a	s	i	=	"	P	>
x	e	x	t	e	x	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	i	s	"	>	b	s	"	=	'	a	T	a	"	i	=	"	a	>		
e	x	x	t	e	x	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	i	s	"	>	a	.	"	a	>	/										
e	x	x	t	e	x	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d	m	p	r	n	l	o	u	f	C	a	i	=	'	s	"	>	!	a	.	"	a	>	/										
x	t	e	x	t	e	n	l	o	u	d</td																																	

静的リソースのベストプラクティス

`<apex:page>` の `action` 属性を使用した静的リソースのコンテンツの表示

コンポーネントの `action` 属性を使用して、Visualforce ページから静的リソースにリダイレクトできます。この機能を使用すれば、機能豊富なカスタムヘルプを Visualforce ページに追加できます。たとえば、ユーザを PDF にリダイレクトするとします。

1. PDF を `mc` という名前で静的リソースとしてアップロードします。
2. 次のページを作成します。

静的リソース参照は、関数でラップされています。そうしないと、ページは適切にリダイレクトされません。

このリダイレクトは PDF ファイルに限りません。静的リソースのコンテンツにページをリダイレクトすることもできます。たとえば、JavaScript、画像、およびその他のマルチメディアファイルを組み合わせた多数の HTML ファイルで構成されるヘルプシステム全体を含む静的リソースを作成できます。エントリポイントが 1 つある限り、リダイレクトは機能します。次に例を示します。

1. ヘルプコンテンツを含む zip ファイルを作成します。
2. zip ファイルを `mc` という名前で静的リソースとしてアップロードします。
3. 次のページを作成します。

ユーザがページにアクセスすると、静的リソースの `mc` ファイルが表示されます。

関連リンク

[静的リソースの使用](#)

コントローラおよびコントローラ拡張のためのベストプラクティス

コントローラでの共有ルールの適用

他の Apex クラスと同様に、すべてのコントローラおよびコントローラ拡張はシステムモードで実行されます。

通常、コントローラまたはコントローラ拡張ではユーザの組織の共有設定、ロール階層および共有ルールを遵守する必要があります。これは、クラス定義で `sh` キーワードを使用することにより実行

できます。詳細は、[Force.com Apex コード開発者ガイド](#) の「`String` キーワードの使用」を参照してください。

コントローラ拡張が標準コントローラを拡張する場合、標準コントローラのロジックはシステムモードでは実行されません。代わりに、ユーザモードで実行されます。このモードでは現在のユーザの権限、項目レベルのセキュリティ、共有ルールが適用されます。

コントローラのコンストラクタを setter メソッドの前に評価

コンストラクタの前に評価される `setter` メソッドには依存しないでください。たとえば、次のコンポーネントでは、コンポーネントのコントローラがコンストラクタメソッドの前にコールされる¹ の `setter` に依存しています。

```

t n n t e nom p c tr lptr c< u arinC C C p =" s " >
xtt e t e m b ue p mt u<d nr a da c pva "a si i ="
t t e ne r et r op n d r u y" is " i ="
{ a e e t g l d lt uc ss a =" !v a" >
xtt e t r b u p < /a ia >
//.
//.

t n n o om p < /a >

blit lt p a c rhC C C pi ss a s

t n te to r u roC do h // s
t t bi { u p rhC C C pi i s ( ) a =
ee e t e l d f u c lu i sv( ; )
te et d do Er M i = ;
}

e r B t e do ped do El M af a = s a
te t r do h // s
t e edline t re{ pg std t uc }ig is sv a ; s ;

```

コンストラクタが setter の前にコールされるため、コンストラクタがコールされると null になります。このため、`EM` が true に設定されることはありません。

メソッドは複数回評価される場合がある(副次的影響を使用しない)

controller、action 属性および式のメソッドを含むメソッドは複数回コールできます。コントローラまたはコントローラ拡張でカスタムメソッドを作成するときの評価の順序または副次的影響には依存しないでください。

コンポーネント facet の使用のためのベストプラクティス

facet は、コンポーネントに示されるデータに関するコンテキスト情報を提供する、Visualforce コンポーネント内の 1 つの領域のコンテンツで構成されます。たとえば、`bl_p` はテーブルのヘッダー、`a` フッター、キャプションの *facet* をサポートしますが、`u_mpc` は列のヘッダーまたはフッターの *facet* のみをサポートします。`p_f` コンポーネントを使用すると、Visualforce コンポーネントのデフォルトの *facet* を独自のコンテンツで上書きできます。*facet* の開始タグと終了タグ内で使用できるのは 1 つの子のみです。

メモ:

すべてのコンポーネントが facet をサポートしているわけではありません。facet をサポートしているコンポーネントは「[標準のコンポーネントの参照](#)」に記載されています。

`p_f` を定義すると、必ず他の Visualforce コンポーネントの子として使用されます。facet の属性では親コンポーネントが上書きされる領域を特定します。

例:<apex:dataTable>での facet の使用

次のマークアップでは、`bl p e ex` コンポーネントを `a a a>p_f` を使用して変更できる方法を示します。

```

t      n t e gnd prd por bl kn c   au   acc a a      " " >
x      B l go p p c < k a a >
e      ed{ bl plt un j< ou acc a ava a! a v =a" >a
e      ex te n t m fn o <c pa h ah =a ia<1 > i si s
e      ex te n n o e cex m t h ! ap f < la < > /a a>
e      ex te n e t p fm oont < f nro a mdf a =a < > > i a
e      et A { ru cco e} ex ft pas ap f n o < (<) > /a a>
e      x n lo u mp < a >
e      ex te ene e p f d r ex < m a a a f "a < a /a a>
e      xtet txe fu u ep l nup } <T ma v a! . a "a > /
e      x n lo u mp < /a >
e      x n lo u mp < a >
e      ex te e w m d r ex h t p f < a a >
e      xtet txe fu u ep l w a r p n <To j a m v a! . a . "a > /
e      x n lo u mp < /a >
ex     d bl p < /a a a a>
B l go p p c < k/a a >
g p p < /a a >

```

メモ: このページで取引先データを表示するには、有効な取引先レコードの ID をページの URL のクエリパラメータとして指定する必要があります。次に例を示します。

```
h e :pSalesforce_instance d/ p_f z o /ai / a 1# 00 000B0G
```

ページは次のように表示されます。

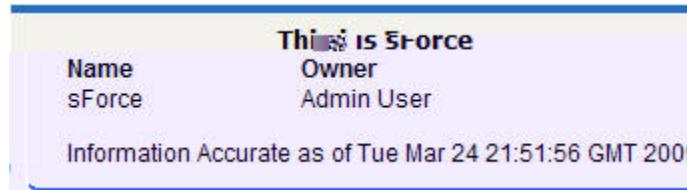


図 27: facet による <apex:dataTable> の拡張

<apex:actionStatus> での facet の使用

facet を使用できる他のコンポーネントは `apex:actionStatus` です。`ao i ap S sa> < コンポーネント S sa>` を拡張することにより、ページが更新されるたびにインジケーターを表示できます。たとえば、次のマークアップを使用して進行状況ホイールを定義できます。

```

t      n g e x p l o c n k m o a a > " a >
x      ro m p f < a a >

```

```

x tet txe ou ut ptl upe t&T n a dmu hn &Weot ahd c aoi s c ! " i =" " > /
x t t tn oe tp amdt ou<r cau ia S sa => S sa>
e ex te nt t p m r < a a => " a>
{ e ermg au} e < ci$ a esp! ! Ro d psid "<> Af g -- v i s y i i a
e ex t p f < /a a>
x t t tn o p a < /a ia S sa>
x t e n t {o l&pm e n &entren a &am rn Per &t idar=" il our c " ="
t t e t m t &utr n c u r ls sa" S sai v => a" > /
x ro m p f < /a >
g p p < /a >

```

関連付けられているコントローラは次のようにカウンタを更新します。

```

blex & pnc d mo p i ss a a = ; 0
n t n r g du c
e e e e b le e e gnc t ent f n c m o d r Ra i ( )
t t n ou c ++ ;
t n m ru lh ;
e et abli t t rpg I gou c i ( )
ntr nru ou c ;

```

ページは次のように表示されます。

Watch this counter: 1 

図 28 : facet による <apex:actionStatus> の拡張

関連リンク

[静的リソースの使用](#)

ページブロックコンポーネントのベストプラクティス

<apex:pageBlockSectionItem>への複数の子コンポーネントの追加

コンポーネントに含めることができる子コンポーネントは最大でも 2 つだけです。ただし、子コンポーネントを 3 つ以上追加する必要がある場合があります。たとえば、<apex:pageBlockSectionItem> の前にアスタリスクを追加し、さらに関連付けられている入力テキスト項目を表示する必要がある場合などです。これを実行するには、次のように <apex:pageBlockSectionItem> コンポーネントでアスタリスクおよび出カラベルをラップします。



メモ: このページで取引先データを表示するには、有効な取引先レコードの ID をページの URL の クエリパラメータとして指定する必要があります。次に例を示します。

http://[Salesforce_instance](#)/001/y/Email/000R000Pa>

```

g : < ! -- Pa >-
n t egndprd por h &n C au acc a a => " >
x ro m p f < a >
e ex te l goe pt l &n e e < o t kMadd m al i => y " = i " >
e ex tB legn p pe c let ntk a n M c os lodu nc y 2 s " s=" " >

```

```

e   ex   e   tB e l g a p   p o c   < m   k a   a   s   >
e   ex   x t  te   nou u b   p   <   a   Pa>
e   ex   x t  te   tx eou ux t e t   t p   o&gt; u   pa*   p   < x   /a   >
e   ex   x t  te   eou u b   p   tleAmp   < ou   actLmro   nvaouf a'cc   m   N   "a   ="
e   ex   x et n txe { u   pltaim n T ou   act redn   vn ou a' bc   m   .   "ai   =
e   ex   ex   e   tB   l g a p   p o c   <c   k/a   a   s   >
e   ex   B   l go p   p c   <   k/a   a   >
x   ro   m p f   <   /a   >
g   p   p   <   /a   >

```

PDF を表示するためのベストプラクティス

Visualforce ページを PDF として表示すると Salesforce 組織の情報を簡単に共有できます。『PDF としてのページの表示』に記載されているガイドラインの他に、次の概念を確認する必要があります。

PDF の表示パフォーマンス

Visualforce ページを PDF として表示する際のパフォーマンスを向上させるには、[グローバル変数](#)を使用して静的画像およびスタイルシートリソースを参照します。



警告: リモートサーバの静的リソースを参照すると、PDF として Visualforce ページを表示するのにさらに時間がかかります。リモートサーバは、[設定] の [セキュリティのコントロール] > [リモートサイトの設定] で、許可されたリモートサイトリストに追加する必要があります。Visualforce を使用して Apex トリガで PDF を表示する場合、リモートリソースを参照することはできません。参照すると例外が発生します。

PDF でのコンポーネントの動作

次のセクションに、PDF でいつでも使用できるコンポーネント、機能がある場合があるコンポーネント、使用できないコンポーネントのリストを示します。通常、次のコンポーネントを使用しません。

- アクションの実行を JavaScript に依存するコンポーネント
- Salesforce スタイルシートに依存するコンポーネント

Visualforce ページがこれらのカテゴリの1つに該当するかどうかを確認するには、ページの任意の場所を右クリックして HTML ソースを参照します。JavaScript () を参照する [タグ](#) またはスタイルシート () を参照する [タグ](#) がある場合、生成された PDF が期待どおりに表示されることをテストする必要があります。

PDF で安全に使用できるコンポーネント

```

t   • o   om   p   <(ページに PDF で安全に使用できるコンポーネントが含まれる場合に限る)
t   d   •   p   <   aL   ia sa   >
n   d   •   p   f   <   a   i   >
t   •   p   f   <   a   a>
e   n   • l du   p   c   (ページに PDF で安全に使用できるコンポーネントが含まれる場合に限る)
tn   • r   p   <   a   s   >
      • m g p   <   a   a
te   c u   u b   p   <   a   L   a>
t   n c u   u   p   <   ka   L   i   >
te   n c u   u   p   <   a   Pa>

```

```

e t t x c u u p p <T a >
  . g p p < a a >
n . l r d p p < a a i G >
n . l r o p p < p a a G >
  r m p p < a a >a
t r . p p < a a >a
ee t . l p h < URL が Salesforce スタイルシートを直接参照しない場合に限る)
e r bl p < va i a a>

```

PDF での使用に注意が必要なコンポーネント

```

e t . r b u p < a ia >
n lo u mp < a a >
t n n o om px p < a a >
t n n o Bm pdo p < a a >y
e d . bl p T< a a a a>

```

PDF で安全に使用できないコンポーネント

```

e n n o upc o c < a ia F i >
e n o ldp < a ia P >
t tn o o pog < a ia R i >
t tn o o p a < a ia S sa>
n t o o upro < pp a ia S >
t t o Bndmm px o < a a a >
n no dmm px < k L ia >
d o l p < a i a
e n n t o d ph c < a L a i s >
l o p f h < a s a
r o m p f < a >
r o m p f < a a >
e n t o l du rp c c < p a i s >
e t n x u h p o pc < k a >
e t n u ldp p < a a >
t e n u l p p < a a >
t e n n u dd p p < a H i >
e t e n t u r p pc < a a S >
t n t x u p p < a a >
t n e x u p p < a a >a a
t w l o p < V a i s i s >
  m o p < a ss a
  m o p < a ss sa>
e t c u u l p p < a a >
B o l go p p c < k a a >
t t B o l B go p u o p c < k a a s >
e t B o l g o p p o c < k a a s >
e t B e l g o p p o c < m k a a s >
e B o l go p b l p c < T k a a a >
e g p o M < a ass a
e g p o M < a ass sa>

```

```

n      B l r p p      <      a      a    >a
e t      B l r p p m I <      a      a      a >
t      t r l d p      <      a      L     a s >
t      n   • or lo pc      <      as      >
t e e n   •   o   φrc      <      as   φ   a>
e t e l x      p b o c k c <      k s      >s
t t l p c      <      as L   i s >
t t l n p o c      <      as      Oi   >
t t l n p o c      <      as      Oi   s>
t l d p o c      <      as      R i   a>
b p      <      a    >a
e n b l p      <      a      a Pa>
• d b r p      <      a      >a
• d b r r p o u      <      p      a      aG >

```

<apex:panelbar> のベストプラクティス

子<apex:panelBarItem>コンポーネントのコレクションの<apex:panelBar>コンポーネントへの追加

コンポーネントに含めることができるのは、<apex:panelBarItem>の子コンポーネントのみです。ただし、子コンポーネントのコレクションを追加する必要がある場合があります。たとえば、取引先に関連付けられている取引先責任者ごとに項目を追加する必要がある場合などです。これを実行するには、次のように<apex:panelBar>コンポーネントで<apex:panelBarItem>をラップします。



メモ: このページで取引先データを表示するには、有効な取引先レコードの ID をページの URL のクエリパラメータとして指定する必要があります。例:

http://Salesforce_instance.com/apex/ContactList?Id=000R00000000000000

```

x e t      n   t e g n d p r d p o r h   n C      a u      a c c      a a      =!      a " >
e ex t      n   B l r p p      <      a      a      a
e ex t      e { r   l p u n t p t n o a   j o c c a   c v r a c   a" !   a   .   s a " v   =! a" >
e ex t      e B l r p l p t l I   r e   a   c e x a f o   a t   B a l l   l r i p s   p m   ' l x   / a   a   a >
e ex n      B l r p p      <      / a      a      >a
ex g p p      <      / a      a      a

```

第 22 章

標準のコンポーネントの参照

標準の Visualforce コンポーネントの完全なリストにはこのガイドの目次からアクセスできます。

`analytics:reportChart`

このコンポーネントは、Salesforce レポートグラフを Visualforce ページに追加するために使用します。グラフデータを絞り込んで特定の結果を表示できます。このコンポーネントは、API バージョン 29.0 以降で使用できます。

レポートグラフを追加する前に、Salesforce アプリケーションのソースレポートにグラフがあることを確認します。

注: この機能は、制限されたパイロットプログラムとして使用できます。プレスリリースや公式声明で参照されている未リリースのサービスまたは機能は、現在利用できず、提供が遅れたり中止されたりする可能性があります。サービスのご購入をご検討中のお客様は、現在利用可能な機能に基づいて購入をご決定ください。

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
<code>filter</code>	String	<p>すでにレポートに含まれている項目の絞り込みに加え、項目でレポートグラフを絞り込んで特定のデータを取得します。レポートに設定できる項目の検索条件は最大 20 個です。条件には、JSON 形式の文字列で次の属性があります。</p> <ul style="list-style-type: none"><code>column</code>: 絞り込み対象にする項目の API 名。<code>operator</code>: 絞り込みに使用する条件の API 名。たとえば、「次の文字列と一致しない」で絞り込むには、API 名「<code>notEqual</code>」を使用します。<code>value</code>: 検索条件。 <p>次に例を示します。</p> <pre>{ "filter": { "operator": "notEqual", "value": "Red", "column": "Status" } }</pre>		29.0	global

属性名	属性型	説明	必須項目	API バージョン	アクセス
reporting_error	Boolean	<p>項目と演算子の API 名を取得するには、次の例のように分析 REST API または分析 Apex ライブラリを介して describe 要求を行います。</p> <p>分析 API</p> <pre>/services/data/v29.0/analytics/reports/00OD000001ZbNHMA0/describe</pre> <p>分析 Apex ライブラリ</p> <ol style="list-style-type: none"> 最初に、describe 要求からレポートメタデータを取得します。 <pre>e te t eo n ro p r g pM R s R a a . e e e dt r b roc p bA s i R M D (000% 00@00N 0)</pre> <ol style="list-style-type: none"> 続いて、次のメソッドを使用して、項目のデータ型に基づいて演算子を取得します。 <pre>e te t eo n ro p r g pM R s R a a . e t t et e teg t l r r pro D p M a @ O a a)</pre>		29.0	global
reporting_error	Boolean	<p>この属性は、エラーのあるグラフをユーザに表示するかどうかを制御するために使用します。エラーがあり、この属性が設定されていない場合、グラフにはエラー以外のデータは表示されません。</p> <p>エラーはさまざまな理由で発生します。たとえば、グラフで使用する項目へのアクセス権がユーザにない場合や、グラフがレポートから削除された場合などがあります。</p> <p>ページでグラフを非表示にするには、この属性を true に設定します。</p>		14.0	global
report_id	String	<p>コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。</p>		29.0	global
size	String	<p>レポートの一意の ID。レポートの ID は、Salesforce のレポート URL から取得したり、API で要求したりできます。</p> <p>次のいずれかの値でグラフのサイズを指定します。</p> <ul style="list-style-type: none"> tiny small medium large huge <p>指定されていない場合、グラフのサイズは medium になります。</p>		29.0	global

apex:actionFunction

AJAX 要求を使用したコントローラの action メソッドを JavaScript コードから直接呼び出すことをサポートするコンポーネントです。 apex:actionFunction はコンポーネントの子である必要があります。

コントローラの action メソッドの他の Visualforce コンポーネントからの呼び出しのみをサポートする apex:actionFunction とは異なり、 apex:script では JavaScript コードのブロック内からコールできる新しい JavaScript 関数を定義します。

注意: API バージョン 23 以降、 apex:actionFunction などの反復コンポーネント内に apex:actionFunction を配置できなくなりました。 apex:actionFunction は反復コンポーネントの後に配置し、反復内には、それをコールする通常の JavaScript 関数を置きます。

例

```

    g : <!-- Pa -->
    tene gerplapc n mo a p a " >
    x ro m pf < a >
    e etn t mhe n u o--ipfdl ov aiS i Hs ya >
    xt ne a n pco ctm ll ne ia ee ei e) ll neHs r yel r" dua = Hs ya "
    tt u m u s sa" y S sa > /
    x ro m pf < /a >
    xt te nou t b d pou < a Pa = " >
    x tet txe oue u p l up ll<To a v H a" " > /
    xt t tn teo teo r t < rn ua qit t g $ sas mau = si .!i = y S sa >
    ex te nt{ e pefm one qrex tpm a =as p E !s < a /a a>
    xt t tn o pa < /a ia S sa>
    xt te nou u b p < /a Pa>
    et e ll C ltho tJm ! n -a aoe H p e y a d r v n dia c m ip $ s a >
    c w e m te dtcoe p om < l$ it i r & i c Hs( pya ,< 00)i/ >
    x tet txe ou pe p l up <Td> }@ c e t d v k o a" i h ?s "ai ps s " > />
    et A ede t odnt h tncl< ! renko enli etv el p idne c c lW ha laqg i , a i
    x e n n doht th J r m n d u p o o v saia h m i a p i i a a >
    xt te nou ne pt el p n< nc doct a h J kr ma ic = t pl la vb saia Y s! " s) y ss=a " >
    e l C c M k i
    xt te nou u b p < /a Pa>
    x ro m pf < a >
    xt n tn{ upcnot e co< e c p o tae F h i nm n moat= !h J r AI " c =a p o v saia "
    r red r o h =" s s " a>
    x e n r t mp mp { r n mt a e a =ai $T u P a as$ a = ls " & " > /
    xt n n upc o c< /a ia F i >
    x ro m pf < /a a >
    g p p < /a a >
    n or ld r C *** / ***
    en blex l pnc d m o p i ss a a / 
    en nr ug m is s ;a
    t e bl et er { ip cg gr m i is U s a( )
    en rn ru u m
    } e e e ble ue en grf o fil co i U Ra Hs ya a;( )
    en n ru lu g I fm U a s . N a;( )
}

```

```

    e   blet t d p c n n r g i v i s S a(i S )
t     n      s     s     a ; S a( i S )
}
t e blet et tf p c g g i i s S a( )
et it r ru
}
e e e e ble e nt erc{ n fo c h Ra o ( )
n n ru lin ;
}
t r etnt r p ng o 立 a i s s a ;' '

```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
o c	ApexPages.Action	ページマークアップの他の場所で DOM イベントによって actionFunction がコールされるときに呼び出される action メソッド。このメソッドを参照するには、差し込み項目の構文を使用します。たとえば、action="[{save}]" ではコントローラの save メソッドを参照します。アクションが指定されていない場合、ページは単に更新されます。		12.0	global
o u f	String	AJAX 要求の完了後にフォーカスされるコンポーネントの ID。		12.0	global
d d mm	String i	ページの他のコンポーネントが actionFunction コンポーネントを参照できるようにする識別子。		12.0	global
m	Boolean i	ページの項目に関連付けられている入力規則を処理することなく、このコンポーネントに関連付けられているアクションをすぐに実行するかどうかを指定する boolean 値。true に設定すると、アクションがすぐに実行され、入力規則はスキップされます。指定されていない場合、この値はデフォルトの false に設定されます。		12.0	global
t lo r a l o d u m f	String a	ページマークアップの他の場所で呼び出された場合の JavaScript 関数の名前。action 属性で指定されているメソッドを実行します。action メソッドが完了すると、reRender 属性で指定されるコンポーネントが更新されます。	はい	12.0	global
pString	onbeforeupdate	イベントの発生時(AJAX 要求が処理されたとき、ただし、ブラウザの DOM が更新される前に呼び出される JavaScript)。		12.0	global
o ol m c p	String	AJAX 更新要求の結果がクライアントで完了したときに呼び出される JavaScript。		12.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
rendered	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		12.0	global
reRender	Object	R action メソッドの結果がクライアントに返されるときに再作成される 1 つ以上のコンポーネントの ID。この値には、単一の ID、ID のカンマ区切りのリスト、または ID のリストまたはコレクションの差し込み項目の式を使用できます。		12.0	global
update	String	s sa AJAX 更新要求の状況を表示する関連付けられているコンポーネントの ID。「actionStatus コンポーネント」を参照してください。		12.0	global
timeout	Integer	i AJAX 更新要求がタイムアウトするまでの時間(ミリ秒)。		12.0	global

apex:actionPoller

指定した間隔に従って AJAX 要求をサーバに送信するタイマーです。各要求により、ページの全体または一部を更新できます。

o lā prc < は作用するリージョン内にある必要があります。たとえば、n o p og < と a i a
 o lā prc e & を併用するには、e oPlāxte n < が n o a p og P & 内にある必要があ i >
 ります。

<apex:actionPoller> を使用するときの考慮事項

- o lā prc < で使用する action メソッドを軽量にする。これは、o lā prc < からコア i a ルされる action メソッドで、DML の実行、外部サービスコール、およびリソースを大量に消費する他の操作を回避するためのベストプラクティスです。指定した間隔で o lā prc < から繰り返しコールされる action メソッドの影響を考慮します。特に、広範囲にわたって配布されたり、長期間開いたままになったりするページで使用する場合には注意が必要です。
- o lā prc < では、ログインセッションをアクティブに保持しながら接続が定期的に更新されます。o lā prc < が存在するページは、非アクティブであるという理由でタイムアウトになることはありません。
- 他のアクションの結果として再表示される場合は、o lā prc < 自体がリセットされます。
- このコンポーネントは拡張リストとは併用しないでください。

例

```
g < ! -- Pa >-
  gexplā pc n k mo a p = " a " >
  ro m p f < a >
  txe ou ut p tl up e tkt n a d u m Weot ahdc aoi s c ! " i =
  xt et n t { o lā prc e n d e n t r a a a m rn PerCt ider=t il our c r l " R " i =
  ro m p f < /a >
```

e n or lō
blex t
n t n r
e e e e ble
t n ou
c n n r
}
e et abli t
c ntr nr
}

属

属性名	属性型	説明	必須項目	API バージョン	アクセス
uom	String	この値には、単一のID、IDのカンマ区切りのリスト、またはIDのリストまたはコレクションの差し込み項目の式を使用できます。	s	10.0	global
om	Integer	Ajax 更新要求の状況を表示する関連付けられているコンポーネントのID。「actionStatus コンポーネント」を参照してください。	i	10.0	global

apex:actionRegion

Ajax 要求の生成時に Force.com サーバが処理する必要のあるコンポーネントを区切る Visualforce ページの領域です。このコンポーネントのみがサーバによって処理されるため、ページのパフォーマンスが向上します。

コンポーネントは要求時にサーバが処理するコンポーネントのみを定義します。要求が完了したときに再表示されるページの領域は定義しません。動作を制御するには、`oncomplete`、`onsuccess`、`onfailure` 属性を使用します。

キーワードの使用も参照してください。

```

ex      e   tB e l gna p  po c  <c m k/a   a   s   >
x      et n   e { u ldp t wtf ne< m tuai l jnd v c  a!  !
tx      wn   } o ! ( ) i y s   a > /
ex      e   tB l gna p  po c  <c   k/a   a   s   >
ex      e   B   l go p  p   c   <   k/a   a   >
x      ro   m p f   <   /a   >
g      p   p   <   /a   >

```

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
refId	String	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		10.0	global
disabled	Boolean	ページの項目に関連付けられている入力規則を処理することなく、このコンポーネントに関連付けられているアクションをすぐに実行するかどうかを指定する boolean 値。true に設定すると、アクションがすぐに実行され、入力規則はスキップされます。指定されていない場合、この値はデフォルトの false に設定されます。		11.0	global
rendered	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		10.0	global
onAjax	Boolean	Ri actionRegion の処理時に actionRegion 外で AJAX が呼び出す動作を無効にするかどうかを指定する boolean 値。true に設定すると、actionRegion 外にあるコンポーネントは AJAX レスポンスには含まれません。false に設定すると、ページのすべてのコンポーネントがレスポンスに含まれます。指定されていない場合、この値はデフォルトの true に設定されます。		10.0	global

apex:actionStatus

AJAX 更新要求の状況を表示するコンポーネントです。AJAX 要求の状況は進行中または完了のいずれかです。

例

```

g      :   <   !   -- Pa   >-
te     ne   gexpla ipc n k mo a   a   " >
x      te   ro m p f   <   a   >
xt     txe ou ut p tl upe tKT n   a du in  Weot ahd c aoi s   c   !   " i   =
xt     tn   teto tpa ere n t   n ma i am c Sgas   a   " i (   i   ... )
te     tx   e on de p mdf ours c u   " ( " i )   "   S sa > /
xt     e n   t{o lpm e n d entere a   am rn Per idar=" il our   c   "   =
tt     et   at u eutr n c u   r ls   sa"   S sai v 1 =a"   s / "

```

```

x      ro m p f      < /a /a >
      g p p < /a /a >
n      or ld r C *** : / ***
      blex t pnc d m o p i ss a a = ; 0
n      t n r g du c
e e e eble e e gnt ent f fn cm o m r Ra i ( )
t      n ou c ++ ; ;
t      n m ru lu
}      et ablt t rpg Igou c i ( )
      ntr nru ou c ; ;
}

```

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
dir	String	生成された HTML コンポーネントの読み取り方向。使用可能な値には「RTL」(右から左) または「LTR」(左から右) があります。		10.0	global
region	String	状況インジケータが状況を表示している actionRegion コンポーネントの ID。		10.0	global
id	String	ページの他のコンポーネントが actionStatus コンポーネントを参照できるようにする識別子。		10.0	global
language	String	「en」または「en-US」など、生成された HTML 出力の基本言語。この属性についての詳細は、 W3C仕様 を参照してください。		10.0	global
label	String	actionStatus コンポーネントをページに表示する方法。使用可能な値には、div HTML 要素にコンポーネントを埋め込む「block」または span HTML 要素にコンポーネントを埋め込む「inline」があります。指定されていない場合、この値はデフォルトの「inline」に設定されます。		10.0	global
onClick	String	onclick イベントが発生した場合(このコンポーネントをクリックした場合)に呼び出される JavaScript。		10.0	global
dblClick	String	ondblclick イベントが発生した場合(このコンポーネントをダブルクリックした場合)に呼び出される JavaScript。		10.0	global
onKeyDown	String	onkeydown イベントが発生した場合(ユーザがキーボードのキーを押した場合)に呼び出される JavaScript。		10.0	global
onKeyPress	String	onkeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出される JavaScript。		10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
onkeyup	String	onkeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出される JavaScript。	10.0	global	
onmousedown	String	onmousedown イベントが発生した場合(ユーザがマウスボタンをクリックした場合)に呼び出される JavaScript。	10.0	global	
onmousemove	String	onmousemove イベントが発生した場合(ユーザがマウスピントを移動した場合)に呼び出される JavaScript。	10.0	global	
onmouseout	String	onmouseout イベントが発生した場合(ユーザがコンポーネントからマウスピントを移動した場合)に呼び出される JavaScript。	10.0	global	
onmouseover	String	onmouseover イベントが発生した場合(ユーザがマウスピントをコンポーネントに重ねた場合)に呼び出される JavaScript。	10.0	global	
onmouseup	String	onmouseup イベントが発生した場合(ユーザがマウスボタンを放した場合)に呼び出される JavaScript。	10.0	global	
r	String	AJAX 要求の開始時に呼び出される JavaScript。	10.0	global	
o	String	AJAX 要求の完了時に呼び出される JavaScript。	10.0	global	
r	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	10.0	global	
r	String	AJAX 要求の開始時の状況要素の表示に使用されるスタイル。主にインライン CSS スタイルを追加するために使用されます。	10.0	global	
r	String	AJAX 要求の開始時の状況要素の表示に使用されるスタイルクラス。外部 CSS スタイルシートを使用するときに適用される CSS スタイルの指定に主に使用されます。	10.0	global	
r	String	AJAX 要求の開始時に表示される状況テキスト。	10.0	global	
o	String	AJAX 要求の完了時の状況要素の表示に使用されるスタイル。インライン CSS スタイルの追加に主に使用されます。	10.0	global	
o	String	AJAX 要求の完了時の状況要素の表示に使用されるスタイルクラス。外部 CSS スタイルシートを使用するときに適用される CSS スタイルの指定に主に使用されます。	10.0	global	
o	String	AJAX 要求の完了時に表示される状況テキスト。	10.0	global	

属性名	属性型	説明	必須項目	API バージョン	アクセス
1	String	s y	AJAX 要求の状態を問わず、状況要素の表示に使用されるスタイル。インライン CSS スタイルの追加に主に使用されます。	10.0	global
1 1	C	String s y	AJAX 要求の状態を問わず、状況要素の表示に使用されるスタイルクラス。外部 CSS スタイルシートを使用するときに適用される CSS スタイルの指定に主に使用されます。	10.0	global
1	String	i	ユーザがコンポーネントにマウスポインタを重ねたときにツールチップとして表示されるテキスト。	10.0	global

Facet

facet 名	説明	API バージョン
r	AJAX 要求の開始時に表示されるコンポーネント。この facet は、startText 属性の代わりに使用します。actionStatus コンポーネントの外観は、属性 name="start" を含む facet によって要求の開始時に制御されるため、start facet が actionStatus コンポーネントの本文に表示される順序は重要ではありません。	10.0
o p	AJAX 要求の完了時に表示されるコンポーネント。この facet は、stopText 属性の代わりに使用します。actionStatus コンポーネントの外観は、属性 name="stop" を含む facet によって要求の完了時に制御されるため、stop facet が actionStatus コンポーネントの本文に表示される順序は重要ではありません。	10.0

apex:actionSupport

他のコンポーネントに AJAX サポートを追加するコンポーネントです。このコンポーネントでは、ボタンのクリック、マウスを重ねるなどの特定のイベントの発生時にサーバが非同期にコンポーネントを更新できます。

また、[apex:button](#) も参照してください。

例

```

g : < ! -- Pa >-
t e n e g e x p l a i c n k m o a p a = a " >
x t te r o m p f < a >
e x t e t n o u e a t b n d p o p k c a a = " " >
e x t e t t x e o u u e p l u p l < T n @ c o M : v k a = i ! ! " > /
e x t e t n e t e d u p r o n k p l a i c a c S v k = i "
e e e n e n t e n t c r n n o m r C i a = i "
x t te n o u u b p p k / a >
x t t t n o e t p m d t o u r c a i a S s a = S s a > /
x t t e t t x e e n t n n T m c g s a = i ( i . . ! )
x t t x e o n d o p T s = ( " ) / 

```

```

x      ro m p f < /a >
      g p p < /a a
e n   or lâ r C *** : / *** /
blex l pnc d m o p i ss a a = ; 0
n t nr g du c
e e e b l e a e gnt ent f fn c m o m r Ra i ( )
t n ou c ++ ; ;
t n m ru llh ; ;
e et abl t t rpg I gou c i ( )
c ntr nru ou c ;
}

```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
c	ApexPages.Action	サーバに対する AJAX 要求によって呼び出される action メソッド。このメソッドを参照するには、差し込み項目の構文を使用します。たとえば、action="{!incrementCounter}" ではコントローラの incrementCounter() メソッドを参照します。アクションが指定されていない場合、ページは単に更新されます。		10.0	global
isDisabled	Boolean	ユーザがコンポーネントを無効にできる boolean 値。「true」に設定すると、イベントの実行時にアクションが呼び出されません。		16.0	
isSkipable	Boolean	関連付けられているイベントのデフォルトのブラウザ処理をスキップするかどうかを指定する boolean 値。true に設定すると、この処理はスキップされます。指定されていない場合、この値はデフォルトの true に設定されます。		10.0	global
eventName	String	AJAX 要求を生成する DOM イベント。使用可能な値には、「onblur」、「onchange」、「onclick」、「ondblclick」、「onfocus」、「onkeydown」、「onkeypress」、「onkeyup」、「onmousedown」、「onmousemove」、「onmouseout」、「onmouseover」、「onmouseup」、「onselect」などがあります。		10.0	global
focusId	String	AJAX 要求の完了後にフォーカスされるコンポーネントの ID。		10.0	global
id	String	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		10.0	global
isManaged	Boolean	ページの項目に関連付けられている入力規則を処理することなく、このコンポーネントに関連付け		11.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
action	String	正在実行中のアクションを示す boolean 値。true に設定すると、アクションがすぐに実行され、入力規則はスキップされます。指定されていない場合、この値はデフォルトの false に設定されます。		11.0	global
beforeDomUpdate	String	onbeforeDomUpdate イベントの発生時(AJAX 要求が処理されたとき、ただし、ブラウザの DOM が更新される前)に呼び出される JavaScript。		10.0	global
complete	String	AJAX 更新要求の結果がクライアントで完了したときに呼び出される JavaScript。		10.0	global
onSubmit	String	AJAX 更新要求がサーバに送信される前に呼び出される JavaScript。		10.0	global
rendered	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		10.0	global
returnIds	Object	AJAX 更新要求の結果がクライアントに返されるときに再作成される 1 つ以上のコンポーネントの ID。この値には、単一の ID、ID のカンマ区切りのリスト、または ID のリストまたはコレクションの差し込み項目の式を使用できます。		10.0	global
status	String	AJAX 更新要求の状況を表示する関連付けられているコンポーネントの ID。「actionStatus コンポーネント」を参照してください。		10.0	global
timeout	Integer	AJAX 更新要求がタイムアウトするまでの時間(ミリ秒)。		10.0	global

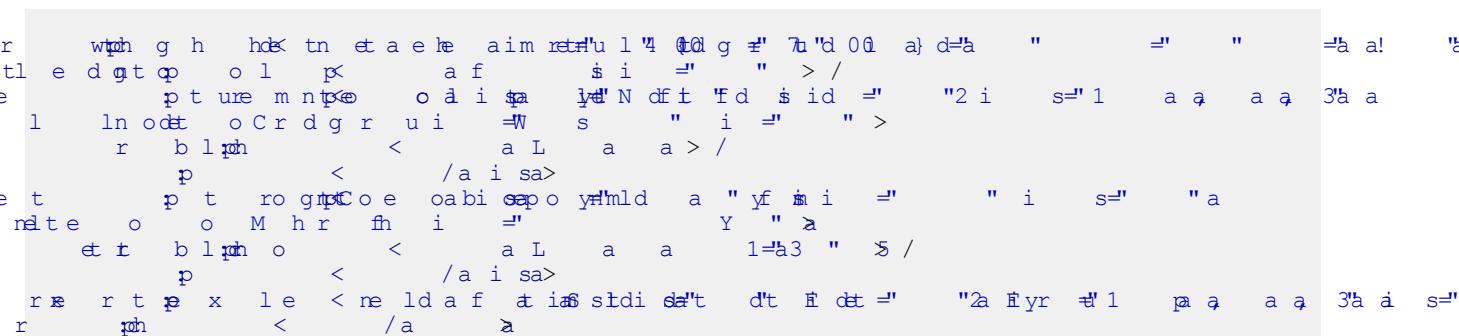
apex:areaSeries

Visualforce グラフで網掛け領域として表示されるデータ系列です。fill 属性を true に設定した折れ線グラフの系列に似ていますが、各 X 値に対する複数の Y 値がレベルとしてそれぞれの上に「積み上げ」られます。

少なくとも、各点が表す領域量を定義する線に沿って各点の X 値および Y 値として使用するデータコレクションの項目、および目盛りとして使用する X 軸および Y 軸を指定する必要があります。グラフにレベルを追加するには、複数の Y 値を追加します。各レベルには新しい色が使用されます。

注: このコンポーネントは apex:areaSeries コンポーネントで囲む必要があります。1 つのグラフに複数のコンポーネントを使用でき、 apex:areaSeries および apex:chart など、コンポーネントを追加できますが、判読しにくい結果になる可能性があります。

3つのY値をレベルとしてプロットする面グラフ



属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
axis	String	i sa このグラフの系列のバインド先の軸。グラフの4辺の境界の1つである必要があります。 この軸のバインド先は同階層のコンポーネントによって定義される必要があります。	はい	26.0	
color	String	c レベル領域の塗りつぶしの色として順に使用される一連の色の値。色は、HTMLスタイル(16進)の色をカンマ区切りで指定します。たとえば、# F 0 Q F 0 E 0 00		26.0	
highlight	Boolean	h マウスポインタを重ねたときに各レベルを強調表示するかどうかを指定するboolean値。指定されていない場合、この値はデフォルトのtrueに設定されます。		23.0	
lineWidth	Integer	h i W h i レベルが強調表示されるときにレベルを囲む線の太さをピクセル単位で指定する整数。		26.0	
opacity	String	c p String i i レベルが強調表示されるときにレベル上に重ねる色の不透明度を表す0 ~ 1までの小数値。		26.0	
style	String	c String i i レベルが強調表示されるときにレベルを囲む線のHTMLスタイルの色を指定する文字列。		26.0	
target	String	i ページの他のコンポーネントがグラフコンポーネントを参照できるようにする識別子。		26.0	global
transparency	String	c p String i a この系列のレベルの塗りつぶされた領域の不透明度を表す0 ~ 1までの小数値。		26.0	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
rendered	Boolean	グラフでグラフ系列を表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		26.0	
renderr	String	各データポイントが表示される方法を追加または上書きする JavaScript 関数の名前を指定する文字列。追加のスタイルを指定またはデータを追加するために実装します。		26.0	
no	display	このグラフ系列をグラフの凡例に追加するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		26.0	
1	points	マウスポインタを重ねたときに各データポイントマーカーのツールチップを表示するかどうかを指定する boolean 値。ツールチップの形式は : です。指定されていない場合、この値はデフォルトの true に設定されます。		26.0	宜
ld	String	このグラフ系列のタイトル。グラフの凡例に表示されます。		26.0	
ld	titles	複数のデータ系列を使用する積み上げグラフの場合、各系列のタイトルをカンマで区切ります。例: @ld Mcrd rd ci =D 。 a Wa, Pa, "			
ld	String	系列のデータポイントごとの X 軸の値の取得元で はい ある、グラフデータに指定された各レコードの項目。この項目はグラフデータのすべてのレコードに存在している必要があります。	はい	26.0	
ld	String	系列のデータポイントごとの Y 軸の値の取得元で はい ある、グラフデータに指定された各レコードの項目。この項目はグラフデータのすべてのレコードに存在している必要があります。	はい	26.0	

apex:attribute

カスタムコンポーネントの属性の定義です。属性タグはコンポーネントタグの子としてのみ指定できます。

名前が id、rendered などである属性は定義できません。これらの属性はすべてのカスタムコンポーネント定義で自動的に作成されます。

例

```

g : < ! -- Pa >-
    g p p < a a
e t n ne m om: Celm pt n re o oneM yc Vl p b r q r a llo ro yr d C w ' 'a = " " > /

```

```

g p p < /a >
tnn@omt n@n@p:m: < !p--> y >->

tnn o om p < a >
ext e t e m beu p m imd nra e tao e V the thy l 'a hshot nin e'famhi s v p a
e ne r et r op n d r u y" i S " i = " " > /
ext e t e m b u p ebmrdr< toro da ndia o c =ae tp lebro "r@ cs b rd f i si s
e ne r et r op n d r u y" i S " i = " " > /

e { le hrd r b rd n kro s C y" ! " >
xet txe du u p l up < Tlm u a v V a" ! y a" > /
tnn o om p < /a >

```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
cc	String	属性を同じ名前空間の任意のページ外で属性として使用できるかどうかを指定する。使用できる値は、「public」(デフォルト)および「global」です。属性を属性の名前空間外で使用できるように指定する場合は、global を使用します。親の apex:component のアクセス属性が global に設定されている場合は、このコンポーネントに対しても global に設定する必要があります。親の apex:component のアクセス属性が public に設定されている場合は、このコンポーネントに対して global に設定することはできません。注: この指定がある属性には、appexchange の管理パッケージの説明にある非推奨ポリシーが適用されます。		14.0	
g	T Object	関連付けられているカスタムコンポーネントコントローラでこの属性の値をクラス変数に割り当てる setter メソッド。この属性を使用する場合は、getter メソッドおよび setter メソッドまたは get 値および set 値を含むプロパティを定義する必要があります。		12.0	global
d lu	f	属性のデフォルト値。		13.0	global
d nr	o c p	属性のテキストによる説明。この説明はカスタムコンポーネントが保存されるとすぐにコンポーネントの参照に含められます。		12.0	global
do	c	これは、一部のパッケージインストールに影響する問題に対応する一時オプション。この属性は今後のリリースで廃止されます。Salesforce による指示がない限り使用しないでください。		15.0	
d	String i	カスタムコンポーネント定義で他のタグが属性を参照できるようにする識別子。		12.0	global

属性名	属性型	説明	必須項目	API バージョン	アクセス
	String	a 関連付けられているカスタムコンポーネントが属性の値を含む場合に Visualforce マークアップで使用されている属性の名前。コンポーネントの定義では、この名前はその他すべての属性とは異なり一意である必要があります。id、rendered、action という名前の属性は定義できません。これらの属性は、すべてのカスタムコンポーネントの定義で自動的に作成されるか、または使用できないかのいずれかです。	はい	12.0	global
	Boolean	i 関連付けられているカスタムコンポーネントが Visualforce ページに含まれる場合に属性の値を指定する必要があるかどうかを指定する boolean 値。 true に設定すると、値が必須になります。指定されていない場合、この値はデフォルトの false に設定されます。		12.0	global
p	String	y 属性の Apex データ型。assignTo 属性を使用してこの属性の値をコントローラクラス変数に割り当てる場合、データ型の値とクラス変数のデータ型が一致している必要があります。type 属性は次のデータ型のみを値として使用できます。 <ul style="list-style-type: none"> string、integer、または boolean などのプリミティブデータ型 Account などの sObject、My_Custom_Object__c、または汎用型の SObject String[]、Contact[] などの配列表記を使用して指定する一次元リスト type="map" を使用して指定する対応付け。対応付けの特定のデータ型を指定する必要はありません。 カスタム Apex データ型 (クラス) 	はい	12.0	global

apex:axis

グラフの軸を定義します。これを使用して軸の単位、目盛り、ラベル、およびその他の表示オプションを設定します。1つのグラフの各境界に1つ、最大4つの軸を定義できます。

e 注: このコンポーネントは **囲む** コンポーネントで囲む必要があります。

例

x	xet	e	p t		' 1	"a a
c		t	dt n ero u		'	3'a a
x	xet	e	p t		="	" a
c	ee e	ln	un			
x	xet	e t	p t			
c		t ndte o o				
x	te	et t	bl			
x	x		p			
e	e	te b r r t pnd		i	" ai sa" i "	
x	e	ne ld	t n			
xe	en	tel t	re p		="	" a f y
t		r	ph		="	1

属性

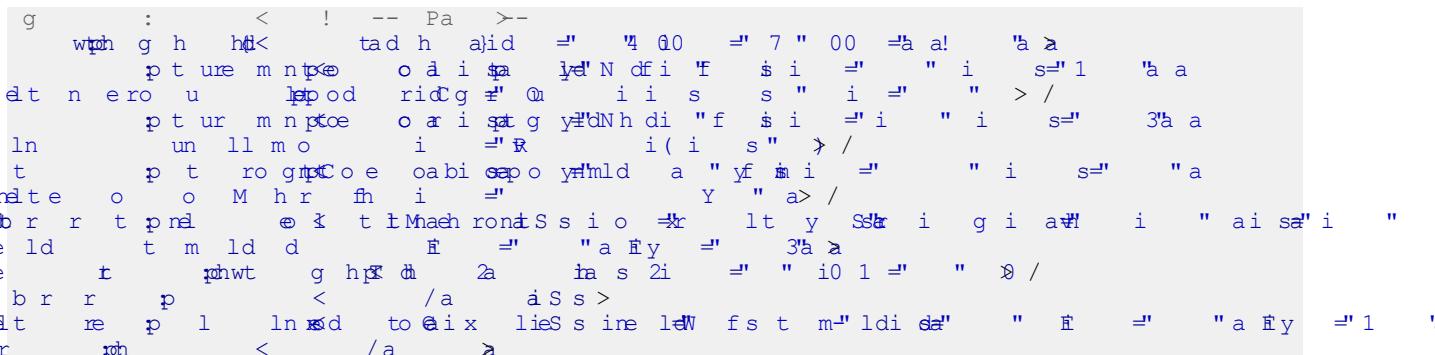
属性名	属性型	説明	必須項目	APIバージョン	アクセス
axisType	String	軸がバインドされているグラフの境界。有効なオプションは、次のとおりです。 <ul style="list-style-type: none"> left right top bottom gauge radial <p>最初の 4 つの位置は、標準線形グラフの各辺に対応します。<code>「gauge」は <code>ug rg p</code></code> <で使用される軸に固有のもので、「radial」は <code>「x e e r d r r p」</code> <で使用される軸に固有のものです。</p>	はい	23.0	
axisVisible	Boolean	軸要素をグラフに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		23.0	
axisMarks	Integer	軸に配置する刻みマークの数を指定する integer 値。設定すると、軸の刻みマークの自動計算が無効になります。軸のタイプが Numeric の場合にのみ有効です。		26.0	
axisLabel	String	軸のラベル。		23.0	
axisType	String	軸のタイプを指定する。これは、軸の間隔およびスペースの計算に使用されます。有効なオプションは、次のとおりです。 <ul style="list-style-type: none"> 「Category」は項目名、項目種別など数値以外の情報に使用する。 「Numeric」は定量的な値に使用する。 <p>「Gauge」は <code>ug rg p</code> <のみで使用される必須のオプション。</p> <p>「Radial」は <code>x e e r d r r p</code> <のみで使用される必須のオプション。</p>	はい	23.0	

apex:barSeries

Visualforce グラフで棒として表示するデータ系列です。少なくとも、各棒の X 値および Y 値として使用するデータコレクションの項目、および目盛りとして使用する X 軸および Y 軸を指定する必要があります。グループ化または積み上げ棒区分をグラフに追加するには、複数の Y 値を追加します。各区分には新しい色が使用されます。

注: このコンポーネントはコンポーネントで囲む必要があります。1つのグラフに複数のコンポーネントおよびコンポーネントとコンポーネントを含めることができます。コンポーネントとコンポーネントを追加できますが、判読しにくい結果になる可能性があります。

例



属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
axis	String	<p>このグラフの系列のバインド先の軸。グラフの4辺の境界の1つである必要があります。</p> <ul style="list-style-type: none"> left right top bottom <p>この軸のバインド先は同階層のコンポーネントによって定義される必要があります。</p>	はい	23.0	
color	String	<p>棒の塗りつぶしの色として順に使用される一連の色の値。色は、HTMLスタイル(16進)の色をカンマ区切りで指定します。たとえば、です。# F0F0F0</p>		26.0	
order	Boolean	<p>属性の色をどのような順序で使用していくかを指定するboolean値。</p> <p>trueに設定すると、cの最初の色が最初の棒が積み上げの場合は棒区分に使用され、2番目の色は2番目の棒というように順に使用されます。各の先頭で、再び最初の色に戻ります。</p>		26.0	

属性名	属性型	説明	必須項目	API バージョン	アクセス
colorOrder	String	e.g. "red,blue,green"	false	26.0	s>
gapPercentage	Integer	棒のグループの間隔を棒の幅のパーセントで指定する整数。		26.0	
gapSize	Integer	個々の棒の間隔を棒の幅のパーセントで指定する整数。		26.0	
highlighted	Boolean	マウスポインタを重ねたときに各棒を強調表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		23.0	
label	String	棒が強調表示されるときに棒の上に重ねる HTML スタイルの色を指定する文字列。		26.0	
strokeWidth	Integer	棒が強調表示されるときに棒を囲む線の太さをピクセル単位で指定する整数。		26.0	
translucent	String	棒が強調表示されるときに棒の上に重ねる色の不透明度を表す 0 ~ 1 までの小数値。		26.0	
strokeColor	String	棒が強調表示されるときに棒を囲む線の HTML スタイルの色を指定する文字列。		26.0	
componentId	String	ページの他のコンポーネントがグラフコンポーネントを参照できるようにする識別子。		23.0	global
direction	String	グラフの棒の方向。有効なオプションは、次のとおりです。 <ul style="list-style-type: none"> horizontal vertical 指定されていない場合、この値はデフォルトの「vertical」に設定されます。	isA	23.0	はい
drawGraph	Boolean	グラフでグラフ系列を表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		23.0	
drawLabels	String	各棒が表示される方法を追加または上書きする JavaScript 関数の名前を指定する文字列。追加のスタイルを指定またはデータを追加するために実装します。		26.0	
noDataLabel	Boolean	このグラフ系列をグラフの凡例に追加するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		23.0	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
d c	Boolean	棒の値をグループ化するか積み上げるかを指定する boolean 値。		26.0	
p	Boolean	マウスポインタを重ねたときに各棒のツールチップを表示するかどうかを指定する boolean 値。このツールチップの形式は <xField>: <yField> です。指定されていない場合、この値はデフォルトの true に設定されます。		23.0	
l i	String	このグラフ系列のタイトル。グラフの凡例に表示されます。		23.0	
e ld te t		複数のデータ系列を使用する積み上げグラフの場合、各系列のタイトルをカンマで区切ります。例: @ld M crd rd ci =D 。 a Wa, Pa, "			
ld	String	系列のデータポイントごとの X 軸の値の取得元であります、グラフデータに指定された各レコードの項目。この項目はグラフデータのすべてのレコードに存在している必要があります。	はい	23.0	
dd g	Integer	左軸および右軸とグラフの棒の間のパディングをピクセル単位で指定する整数。		26.0	
ld	String	系列のデータポイントごとの Y 軸の値の取得元であります、グラフデータに指定された各レコードの項目。この項目はグラフデータのすべてのレコードに存在している必要があります。	はい	23.0	
dd g	Integer	上軸および下軸とグラフの棒の間のパディングをピクセル単位で指定する整数。		26.0	

apex:canvasApp

指定された canvasApp 属性または apex:canvasApp 属性のペアで識別されるCanvas アプリケーションを表示します。 apex:canvasApp 属性が優先されます。

要件:

- 組織で Force.com Canvas が有効になっている必要があります。

<apex:canvasApp> コンポーネントを使用するときの考慮事項は、次のとおりです。

- 開発組織とは、Canvas アプリケーションを開発し、パッケージ化する組織です。
- インストール組織とは、パッケージ化された Canvas アプリケーションのインストール先の組織です。
- Visualforce ページでの apex:canvasApp コンポーネントの使用状況は、Canvas アプリケーションのアプリケーション名または開発者名が変更されても更新されません。

- キャンバスアプリケーションは `apex:canvasApp` を使用してそれを参照する Visualforce ページが存在する場合は削除できません。

次の例では、開発者名のみを使用してキャンバスアプリケーションを表示します。組織に名前空間プレフィックスがない場合は、`namespacePrefix` 属性を使用しないでください。

```
x e new eg op d ip h l < a s a' s 'a >
x n A t e p m a v a ca mia i pp aN a v a N "a > /
ex g p p < /a a
```

次の例では、アプリケーション名のみを使用してキャンバスアプリケーションを表示します。

```
x e new eg op d ip h l < a s a' s 'a >
x n A t e p m a v a ca mia i pp aN a v a N "a > /
ex g p p < /a a
```

次の例では、開発者名と、キャンバスアプリケーションが作成された組織の名前空間プレフィックスを使用して、キャンバスアプリケーションを表示します。

```
x e new eg op d ip h l < a s a' s 'a >
x n A t e p m a v a ca mia i pp aN a v a N "a > /
x en r e c p o e f g x msar a i P c = D f O N sa a i P " > /
ex g p p < /a a
```

次の例では、アプリケーション名と、キャンバスアプリケーションが作成された組織の名前空間プレフィックスを使用して、キャンバスアプリケーションを表示します。

```
x e new eg op d ip h l < a s a' s 'a >
x n A t e p m a v a ca mia i pp aN a v a N "a > /
x en r e c p o e f g x msar a i P c = D f O N sa a i P " > /
ex g p p < /a a
```

次の例では、特定の出力パネルでキャンバスアプリケーションを表示します。

```
x t te nou tu b l ou bl < a s a' s 'a >
x ne eA e d l or n pp m a ap v a w pp N a v a N "a
x en r e c p o e f g x msar n a i P c = D e f dc t N n s de l o ma i P c " o p f r a C = ! . y i a
x t te nou u b p < /a Pa>
```

属性

	属性名	属性型	説明	必須項目	APIバージョン	アクセス
n1	appName	String	キャンバスアプリケーションの名前。applicationName または developerName が必要です。		29.0	
b r d r	borderWidth	String	キャンバスアプリケーションの境界線の幅(ピクセル)。指定されていない場合、デフォルトの 0 ピクセルに設定されます。		29.0	
d	windowId	String	キャンバスアプリケーションウインドウの一意の ID。この属性を使用して、イベントをキャンバスアプリケーションの対象にします。		29.0	
n o r	renderId	String	キャンバスアプリケーションが表示される HTML 要素 ID。指定されていない場合、デフォルトの null に設定されます。この属性で指定されるコンテナを <apex:canvasApp> コンポーネントの後に配置することはできません。		29.0	
d l or	developerName	String	キャンバスアプリケーションの開発者名。この名前は、キャンバスアプリケーションが作成されたときに定義され、キャンバスアプリケーションのプレビューアで表示されます。developerName または applicationName が必要です。		29.0	
g h h	height	String	キャンバスアプリケーションウインドウの高さ(ピクセル)。指定されていない場合、デフォルトの 900 ピクセルに設定されます。		29.0	
d	componentId	String	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
m	maxHeight	String	キャンバスアプリケーションウインドウの高さの最大値(ピクセル)。デフォルトは 2000 ピクセルで、「infinite」も有効な値です。		29.0	
at	width	String	キャンバスアプリケーションウインドウの幅の最大値(ピクセル)。デフォルトは 1000 ピクセルで、「infinite」も有効な値です。		29.0	
x m r	componentPath	String	キャンバスアプリケーションが作成された Developer Edition 組織の名前空間の値。キャンバスアプリケーションが Developer Edition 組織で作成されていない場合は省略できます。指定されていない場合、デフォルトの null に設定されます。		29.0	
o A	errorFunction	String	キャンバスアプリケーションの表示に失敗した場合にコールされる JavaScript 関数の名前。		29.0	
o A	loadFunction	String	キャンバスアプリケーションの読み込み後にコールされる JavaScript 関数の名前。		29.0	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
param	String	キャンバスアプリケーションに渡されるパラメータのオブジェクト表現。JSON 形式またはJavaScript オブジェクトリテラルとして指定する必要があります。JavaScript オブジェクトリテラルとしてのパラメータの例: {param1:'value1',param2:'value2'}。指定されていない場合、デフォルトの null に設定されます。		29.0	
display	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global
center	String	キャンバスアプリケーションウィンドウでスクロールバーを使用するかどうかを指定します。有効な値は auto yes no です。指定されていない場合、デフォルトの no に設定されます。		29.0	
height	String	キャンバスアプリケーションウィンドウの幅(ピクセル)。指定されていない場合、デフォルトの 800 ピクセルに設定されます。		29.0	

apex:chart

Visualforce グラフです。サイズおよびデータバインドを含むグラフの一般的な特性を定義します。

例

```
graph TD; A((A)) --> B((B)); C((C)) --> D((D)); E((E)) --> F((F)); G((G)) --> H((H)); I((I)) --> J((J)); K((K)) --> L((L)); M((M)) --> N((N)); O((O)) --> P((P)); Q((Q)) --> R((R)); S((S)) --> T((T)); U((U)) --> V((V)); W((W)) --> X((X)); Y((Y)) --> Z((Z));
```

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
animate	Boolean	最初に表示するときにグラフをアニメーションさせるかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		23.0	
color	String	グラフの背景に使用する色を HTML スタイルの(16 進数)色として指定する文字列。指定されていない場合は、白の背景が使用されます。		26.0	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
colorSet	String	各子系列で使用される一連の色。色は、HTML スタイル (16 進) の色をカンマ区切りで指定します。たとえば、#0000FF です。これらの色は Visualforce グラフで使用されるデフォルトの色を上書きします。これらの色は同様に個々のデータ系列に指定した colorSet で上書きできます。		26.0	
dataBind	Object	a グラフのデータバインドを指定する。これには、式のコントローラメソッド参照、JavaScript 関数、または JavaScript オブジェクトを指定できます。どの場合でも、結果はレコードの配列である必要がありますが、すべてのレコードに子データ系列コンポーネントで参照されるすべての項目を含める必要があります。	はい	23.0	
floatable	Boolean	i CSS の絶対位置設定を使用して、通常の HTML ドキュメントフロー外にグラフをフロート表示するかどうかを指定する boolean 値。		23.0	
height	String	i グラフの長方形の高さ。整数で指定した場合はピクセル単位となり、パーセント記号が続く数値で指定した場合は HTML 要素を含む高さのパーセントとなります。ブラウザおよびデータセット全体で一貫性のある動作を実現するには、ピクセルを使用してください。極端に高いまたは低いグラフを生成される可能性がある可変のデータセットを操作する場合は、パーセントを使用します。多数の棒を含む横棒グラフに最も効果的です。 注意: パーセントの高さは Firefox では正しく表示されないことが判明しています。	はい	23.0	
hidden	Boolean	i 初期状態でグラフを表示するか非表示にするかを指定する boolean 値。ページが初めて表示されるときにグラフを非表示にする場合は、true に設定します。		23.0	
label	String	i ページの他のコンポーネントがグラフコンポーネントを参照できるようにする識別子。		23.0	global
legendEnabled	Boolean	e e ex n l d g p デフォルトのグラフの凡例を表示するかどうかを指定する boolean 値。グラフに詳細オプションのコンポーネントを追加します。指定されていない場合、この値はデフォルトの true に設定されます。		23.0	
name	String	a 追加設定の指定または動的操作の実行に使用する生成された JavaScript オブジェクトの名前。名前はすべてのグラフコンポーネントで一意である必要		23.0	

属性名	属性型	説明	必須項目	API バージョン	アクセス
e	ex	e	があります。含む側の最上位コンポーネント (<code>xge</code> <code>p</code> <code>tpn</code>) または <code><om</code> <code>xe</code> <code>pa</code>) の名前	a	>
r d r d	e	t m M r c p M C	空間が指定されている場合、グラフ名にプレフィックスとして名前空間が付けられます。たとえば、	a y a	となります。
r d r o	Boolean		コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの <code>true</code> に設定されます。	23.0	
r bl	T String		グラフを表示する DOM 要素の ID を指定する文字列。	23.0	
m h	Boolean	§	表示されたグラフのサイズを変更できるかどうかを指定する boolean 値。	23.0	
m h	String		使用するグラフのテーマの名前を指定する文字列。テーマには、定義済みの一連の色が用意されています。使用可能なテーマは、次のとおりです。	26.0	
			<ul style="list-style-type: none"> • Salesforce • 青 • 緑 • 赤 • 紫 • 黄 • 空 • カテゴリ1 • カテゴリ2 • カテゴリ3 • カテゴリ4 • カテゴリ5 • カテゴリ6 		
d	e t	lоро	デフォルトの「Salesforce」では、Salesforce レポートおよび分析のグラフと同じ色が指定されます。 <code>c</code> を使用すれば、グラフコンポーネントに独自の色を定義できます。	23.0	
	h	String i	グラフの長方形の幅。整数で指定した場合はピクセル単位となり、パーセント記号が続く数値で指定した場合は HTML 要素を含む幅のパーセントとなります。ブラウザおよびデータセット全体で一貫性のある動作を実現するには、ピクセルを使用してください。グラフでブラウザウィンドウの幅を広げる場合は、パーセントを使用します。	はい	23.0

apex:chartLabel

ラベルの表示方法を定義します。このコンポーネントは、ラップするコンポーネントに応じて、データ系列のラベル、円グラフの区分のラベル、および軸のラベルの表示に影響を与えるオプションを提示します。

注: このコンポーネントはデータ系列コンポーネントまたはコンポーネントで囲む必要があります。

例

```
<apex:chartLabel c="red" p="inside" f="label">
</apex:chartLabel>
```

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
c	String	HTMLスタイルの(16進)色として指定されるラベルテキストの色。指定されていない場合、この値はデフォルトの「#000」(黒)に設定されます。		23.0	
p	String	<p>ラベルの位置を指定するか、ラベルの表示を無効化する。有効なオプションは、次のとおりです。</p> <ul style="list-style-type: none"> rotate middle insideStart insideEnd outside over under none (ラベルは表示されない) <p>指定されていない場合、この値はデフォルトの「middle」に設定されます。</p>		23.0	
f	String	系列の各データポイントのラベルの取得元である、グラフデータに指定された各レコードの項目。この項目はグラフデータのすべてのレコードに存在している必要があります。指定されていない場合、この値はデフォルトの「name」に設定されます。		23.0	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
font	String	CSS スタイルのフォント定義として、ラベルテキストに使用するフォント。指定されていない場合、この値はデフォルトの「11px Helvetica, sans-serif」に設定されます。		23.0	
id	String	ページの他のコンポーネントがグラフコンポーネントを参照できるようにする識別子。		23.0	global
margin	Integer	視覚化の原点に対するラベルからの最小距離をピクセル単位で指定する。指定されていない場合、この値はデフォルトの 50 に設定されます。		23.0	
labelOrientation	String	ラベルテキストの文字を表示する(通常または縦書き)。有効なオプションは、次のとおりです。 <ul style="list-style-type: none">horizontalvertical 指定されていない場合、通常の左から右のテキストではこの値はデフォルトの、「horizontal」に設定されます。		23.0	
labelVisible	Boolean	グラフのラベルをグラフに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		23.0	
labelText	String	軸または系列のラベルとして表示されるラベルに追加または上書きする JavaScript 関数の名前を指定する文字列。		26.0	
labelAngle	Integer	ラベルテキストを回転する角度。指定されていない場合、この値はデフォルトの 0 に設定されます。		23.0	

apex:chartTips

データ系列の要素にマウスを重ねると表示されるツールチップを定義します。このコンポーネントでは、データ系列のコンポーネントの tips 属性を true に設定すると表示されるデフォルトのツールチップより多くの設定オプションを提供します。

注: このコンポーネントはデータ系列コンポーネントで囲む必要があります。

例

```

<apex:chartTips
    title="With a header and footer"
    header="Header content"
    footer="Footer content"
    type="text"
    font="14px Arial, sans-serif"
    color="black"
    background="white"
    border="1px solid black"
    padding="5px"
    width="300px"
    height="100px"
    tips="true"/>

```

x x t e ld t m ld d g h p a 2 a " a k y = 1 ' a a i o 1 " & /
 e e t e b r r p < /a a s s >
 r r p h

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
height	Integer	ツールチップの高さ(ピクセル単位)。		23.0	
label	String	ページの他のコンポーネントがグラフコンポーネントを参照できるようにする識別子。		23.0	global
labelField	String	系列のデータポイントごとのツールチップのラベルとして使用するグラフデータの各レコードの項目。ツールチップは <label>: <value> として表示されます。この項目はグラフデータのすべてのレコードに存在している必要があります。この値が指定されていない場合、円およびゲージグラフの系列では labelField、それ以外のデータ系列では xField にデフォルトで設定されます。		23.0	
renderData	Boolean	データ系列のツールチップをグラフに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		23.0	
renderHint	String	グラフのヒントとして表示されるツールチップに追加または上書きする JavaScript 関数の名前を指定する文字列。		26.0	
showOnMouseOver	Boolean	マウスポインタを置いたときにグラフのヒントを表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		23.0	
valueLabel	String	系列のデータポイントごとのツールチップの値として使用するグラフデータの各レコードの項目。ツールチップは <label>: <value> として表示されます。この項目はグラフデータのすべてのレコードに存在している必要があります。この値が指定されていない場合、円およびゲージグラフの系列では dataField、それ以外のデータ系列では yField にデフォルトで設定されます。		23.0	
width	Integer	ツールチップの幅(ピクセル単位)。		23.0	

apex:column

テーブルの単一の列です。 apex:column コンポーネントは必ず apex:repeat または apex:repeat の子である必要があります。

項目を `u` の `< u` 属性として指定すると、その項目に関連付けられている表示ラベルがデフォルトで列ヘッダーとして使用されます。この動作を上書きするには、`列の header facet` 属性か、または `列の header facet` を使用します。

このコンポーネントでは、「`html-`」プレフィックスを使用した **HTML パススルー属性**がサポートされています。パススルー属性は、テーブルのすべての行の列に対して生成された `<td>` タグに適用されます。

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
<code>isNew</code>	<code>Boolean</code>	列がテーブルで新しい行を開始するかどうかを指定する <code>boolean</code> 値。 <code>true</code> に設定されている場合、列が新しい行を開始します。指定されていない場合、この値はデフォルトの <code>false</code> に設定されます。		10.0	global
<code>label</code>	<code>String</code>	この列がテーブルでまたぐ列の数。この値はヘッダーセルおよびフッターセルには適用されません。		10.0	global
<code>labelDirection</code>	<code>String</code>	生成された列のテキストの読み取り方向。使用可能な値には「RTL」(右から左) または「LTR」(左から右) があります。この値はヘッダーセルおよびフッターセルには適用されません。		10.0	global
<code>listStyleClass</code>	<code>String</code>	列フッターの表示に使用されるスタイルクラス(定義されている場合)。この属性は、主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。		10.0	global
<code>listStyleType</code>	<code>String</code>	この属性は Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。		10.0	global
<code>order</code>	<code>String</code>	この属性は Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。		10.0	global
<code>rowspan</code>	<code>String</code>	この属性は Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。		10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
oor ol	f c c String k	この属性は Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。		10.0	global
oor abl	f c c String k	この属性は Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。		10.0	global
woom od	of String	この属性は Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。		10.0	global
oor or	f p String	この属性は Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。		10.0	global
oor o	f String	この属性は Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。		10.0	global
woom odmf	String	この属性は Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。		10.0	global
oor ou mf m	String	この属性は Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。		10.0	global
t oor o ou mf u	String	この属性は Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。		10.0	global
oor o ou mf	String	この属性は Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。		10.0	global
oor o ou mf u	String	この属性は Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。		10.0	global
oor l f	String	s この属性は Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。		10.0	global
oor l f	String	i この属性は Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。		10.0	global
oor l uf	String v	列フッターに表示されるテキスト。この属性の値を指定すると、列の footer facet を使用できません。		12.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
dr1_h	c	String	a テーブルヘッダーの表示に使用されるスタイルクラス(定義されている場合)。この属性は、主に、外部 CSS スタイルシートを使用するときに適用されるCSSスタイルを指定するために使用されます。	10.0	global
dm_loh	c	String	a このヘッダー列がテーブルでまたぐ列の数(定義されている場合)。この属性はVisualforce ページのバージョン 16.0 以降では使用できません。	10.0	global
drd_rh		String	a この属性は Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。	10.0	global
ndl_hg		String	a この属性は Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。	10.0	global
ndr_olh	c c	String k	a この属性は Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。	10.0	global
dr_dblh	c c	String k	a この属性は Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。	10.0	global
wdmo_cho		String	a この属性は Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。	10.0	global
dro_im	p	String	a この属性は Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。	10.0	global
ndro_hu		String	a この属性は Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。	10.0	global
wdrno_ohmo		String	a この属性は Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。	10.0	global
dro_outmo_m		String	a この属性は Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。	10.0	global
tdro_ohmou		String	a この属性は Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。	10.0	global
dro_outmo		String	a この属性は Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。	10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
dr_o_chmu	String	<p>この属性は Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。</p>		10.0	global
dr_lh	String	<p>この属性は Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。</p>		10.0	global
dr_lh	String	<p>この属性は Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。</p>		10.0	global
dr_lhu	String ^V	<p>列ヘッダーに表示されるテキスト。この属性の値を指定すると、列の headerFacet を使用できません。この属性の値を指定することで、列本文の inputField または outputField を使用する場合に表示されるデフォルトのヘッダーラベルが上書きされます。</p>		12.0	global
dg	String ⁱ	<p>ページの他のコンポーネントが列コンポーネントを参照できるようにする識別子。</p>		10.0	global
lg	String	<p>「en」または「en-US」など、生成された HTML 出力の基本言語。この属性についての詳細は、W3C仕様を参照してください。</p>		10.0	global
ol_cc	String ⁱ	<p>列で onclick イベントが発生した場合(列をクリックした場合)に呼び出される JavaScript。この値はヘッダーセルおよびフッターセルには適用されません。</p>		10.0	global
dbl_ll_cc	String ⁱ	<p>列で ondblclick イベントが発生した場合(列をダブルクリックした場合)に呼び出される JavaScript。この値はヘッダーセルおよびフッターセルには適用されません。</p>		10.0	global
node_o_k	String ^y	<p>列で onkeydown イベントが発生した場合(ユーザがキーボードのキーを押した場合)に呼び出される JavaScript。この値はヘッダーセルおよびフッターセルには適用されません。</p>		10.0	global
or_pk	String ^y	<p>列で onkeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出される JavaScript。この値はヘッダーセルおよびフッターセルには適用されません。</p>		10.0	global
ou_pk	String ^y	<p>列で onkeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出される JavaScript。この値はヘッダーセルおよびフッターセルには適用されません。</p>		10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
w onmousedown	String	s 列で onmousedown イベントが発生した場合(ユーザがマウスボタンをクリックした場合)に呼び出される JavaScript。この値はヘッダーセルおよびフッターセルには適用されません。		10.0	global
o o um onmousemove	String	s 列で onmousemove イベントが発生した場合(ユーザがマウスポインタを移動した場合)に呼び出される JavaScript。この値はヘッダーセルおよびフッターセルには適用されません。		10.0	global
o o um ou onmouseout	String	s 列で onmouseout イベントが発生した場合(ユーザが列からマウスポインタを移動した場合)に呼び出される JavaScript。この値はヘッダーセルおよびフッターセルには適用されません。		10.0	global
o o um or onmouseover	String	sv 列で onmouseover イベントが発生した場合(ユーザがマウスポインタを列に重ねた場合)に呼び出される JavaScript。この値はヘッダーセルおよびフッターセルには適用されません。		10.0	global
o o um u onmouseup	p String	s 列で onmouseup イベントが発生した場合(ユーザがマウスボタンを放した場合)に呼び出される JavaScript。この値はヘッダーセルおよびフッターセルには適用されません。		10.0	global
r d rd	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		10.0	global
nr o	p Integer	s この列の各セルがテーブルで占める行の数。		10.0	global
1	String	s y 列の表示に使用されるスタイル。主に、INLINE CSS スタイルを追加するために使用されます。この値はヘッダーセルおよびフッターセルには適用されません。		10.0	global
l l	c String	s y 列の表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。この値はヘッダーセルおよびフッターセルには適用されません。		10.0	global
l	String	i ユーザがコンポーネントにマウスポインタを重ねたときにツールチップとして表示されるテキスト。		10.0	global
l u	String	a ヘッダーセルおよびフッターセル以外ですべての列のセルに表示されるテキスト。この属性の値を指定すると、列の開始タグおよび終了タグの間にコンテンツを追加することはできません。		12.0	global

属性名	属性型	説明	必須項目	API バージョン	アクセス
h	String	列のピクセル(px)またはパーセント(%)単位の幅。 指定されていない場合、この値はデフォルトの100ピクセルに設定されます。	10.0	global	

Facet

facet名	説明	API バージョン
f	列のフッターセルに表示されるコンポーネント。列の最後のセルの表示は、name="footer"を含むfacetによって制御されるため、footer facetが列コンポーネントの本文に表示される順序は重要ではありません。footer facetを使用する場合、列のfooterValue属性の値を指定できません。	10.0
h	列のヘッダーセルに表示されるコンポーネント。列の最初のセルの表示は、name="header"を含むfacetによって制御されるため、header facetが列コンポーネントの本文に表示される順序は重要ではありません。header facetを使用する場合、列のheaderValue属性の値を指定できません。このfacetの値を指定することで、列本文のinputFieldまたはoutputFieldを使用する場合に表示されるデフォルトのヘッダー表示ラベルが上書きされます。	10.0

apex:commandButton

このタグの指定値に応じて、submit、reset、またはimageに設定されている型属性を持つHTML入力要素として表示されるボタンです。このボタンはコントローラで定義されているアクションを実行してから、現在のページを更新するか、またはアクションで返されるPageReference変数に基づいて他のページに移動します。

コンポーネントは必ずコンポーネントの子である必要があります。

このコンポーネントでは、「html-」プレフィックスを使用したHTMLパススルー属性がサポートされています。パススルー属性は、生成されたpタグに適用されます。

上述の例では次のHTMLを表示します。

```
<apex:commandButton type="button" value="Submit" />
```

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
cc	String	コマンドボタンにフォーカスを置くキーボードのアクセスキー。コマンドボタンにフォーカスがあるときにEnterキーを押す操作は、ボタンをクリックする操作と同じです。		10.0	global
c	ApexPages.Action	サーバに対するAJAX要求によって呼び出されるactionメソッド。このメソッドを参照するには、差し込み項目の構文を使用します。たとえば、action="{!save}"ではコントローラのsaveメソッドを参照します。アクションが指定されていない場合、ページは単に更新されます。標準コントローラでsave、edit、またはdeleteアクションに関連付けられているコマンドボタンは、ユーザに適切な権限がある場合にのみ表示されます。同様に、editアクションおよびdeleteアクションに関連付けられているコマンドボタンは、レコードがページに関連付けられている場合にのみ表示されます。		10.0	global
l	String	コマンドボタンの代替のテキストによる説明。		10.0	global
d r	String	生成されたHTMLコンポーネントの読み取り方向。使用可能な値には「RTL」(右から左)または「LTR」(左から右)があります。		10.0	global
d b l d	Boolean	このボタンを無効な状態で表示するかどうかを指定するboolean値。trueに設定されている場合、ボタンは無効な状態で表示されます。指定されていない場合、この値はデフォルトのfalseに設定されます。		10.0	global
d	String	ページの他のコンポーネントがcommandButtonコンポーネントを参照できるようにする識別子。		10.0	global
m g	String	このボタンとして表示される画像の絶対または相対URL。指定されている場合、生成されたHTML入力要素の型は「image」に設定されます。		10.0	global
d m m	Boolean	ページの項目に関連付けられている入力規則を処理することなく、このコンポーネントに関連付けられているアクションをすぐに実行するかどうかを指定するboolean値。trueに設定すると、アクションがすぐに実行され、入力規則はスキップされます。指定されていない場合、この値はデフォルトのfalseに設定されます。		11.0	global
l g	String	「en」または「en-US」など、生成されたHTML出力の基本言語。この属性についての詳細は、 W3C仕様 を参照してください。		10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
blur	String	onblur イベントが発生した場合(フォーカスがコマンドボタンから離れた場合)に呼び出される JavaScript。		10.0	global
click	String	onclick イベントが発生した場合(ユーザがコマンドボタンをクリックした場合)に呼び出される JavaScript。		10.0	global
complete	String	AJAX 更新要求の結果がクライアントで完了したときに呼び出される JavaScript。		10.0	global
dblclick	String	ondblclick イベントが発生した場合(ユーザがコマンドボタンをダブルクリックした場合)に呼び出される JavaScript。		10.0	global
focus	String	onfocus イベントが発生した場合(フォーカスがコマンドボタンにある場合)に呼び出される JavaScript。		10.0	global
keydown	String	onkeydown イベントが発生した場合(ユーザがキーボードのキーを押した場合)に呼び出される JavaScript。		10.0	global
keypress	String	onkeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出される JavaScript。		10.0	global
keyup	String	onkeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出される JavaScript。		10.0	global
mousedown	String	onmousedown イベントが発生した場合(ユーザがマウスボタンをクリックした場合)に呼び出される JavaScript。		10.0	global
mousemove	String	onmousemove イベントが発生した場合(ユーザがマウスポインタを移動した場合)に呼び出される JavaScript。		10.0	global
mouseout	String	onmouseout イベントが発生した場合(ユーザがコマンドボタンからマウスポインタを移動した場合)に呼び出される JavaScript。		10.0	global
mouseover	String	onmouseover イベントが発生した場合(ユーザがマウスポインタをコマンドボタンに重ねた場合)に呼び出される JavaScript。		10.0	global
mouseup	String	onmouseup イベントが発生した場合(ユーザがマウスボタンを放した場合)に呼び出される JavaScript。		10.0	global
rendered	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
dr	Object	R AJAX 更新要求の結果がクライアントに返されるときに再作成される1つ以上のコンポーネントのID。この値には、単一のID、IDのカンマ区切りのリスト、またはIDのリストまたはコレクションの差し込み項目の式を使用できます。	10.0	global	
u	String	s sa AJAX 更新要求の状況を表示する関連付けられているコンポーネントのID。「actionStatus コンポーネント」を参照してください。	10.0	global	
l	String	s y commandButton コンポーネントの表示に使用されるスタイル。主に、インライン CSS スタイルを追加するために使用されます。	10.0	global	
l l	C	String s y commandButton コンポーネントの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。	10.0	global	
b d	String	i a ユーザが Tab キーを繰り返し押したときに、他のページコンポーネントと比較して、このボタンが選択される順序。この値は、ユーザが Tab キーを押したときに選択される最初のコンポーネントを0として、0 ~ 32767 の数値である必要があります。	10.0	global	
om	Integer	i AJAX 更新要求がタイムアウトするまでの時間(ミリ秒)。	10.0	global	
l	String	i ユーザがコンポーネントにマウスポインタを重ねたときにツールチップとして表示されるテキスト。	10.0	global	
l u	Object	a commandButton にラベルとして表示されるテキスト。	10.0	global	

apex:commandLink

コントローラで定義されているアクションを実行してから、現在のページを更新するか、またはアクションで返される PageReference 变数に基づいて他のページに移動するリンクです。[dmm](#) [px](#) [<コンポーネント](#) [ia](#) [>](#) [m](#) [p](#) [f](#) コンポーネントの子である必要があります。

[n](#) [no](#) [dmm](#) [px](#) [<に要求パラメータを追加するには、ネストされた](#) [mp](#) [p](#) コンポーネントを使用 [a>a](#) します。

[t](#) [bindmm](#) [pxto](#) [t](#) [<now](#) [ua](#) [p](#) [p](#) も参照してください。 [>](#)

このコンポーネントでは、「html_」プレフィックスを使用した [HTML パススルー属性](#)がサポートされています。パススルー属性は、生成された [タグ](#)に適用されます。 [> a](#)

例

```
n    m{ dmm e } < e   k a l e h ia d i a =" n o s d m i t C a" v S á =" k L ia " > /
```

上述の例では次の HTML を表示します。

```
e td e t g h ro n : h k o d i n a " C r n e a o i n f k a r g d a l " J k "# i = a s "( k S a >
```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
cc	String	コマンドリンクにフォーカスを置くキーボードのアクセスキュー。コマンドリンクにフォーカスがあるときにEnterキーを押す操作は、リンクをクリックする操作と同じです。		10.0	global
c	ApexPages.Action	サーバに対する AJAX 要求によって呼び出される action メソッド。このメソッドを参照するには、差し込み項目の構文を使用します。たとえば、action="{!save}" ではコントローラの save() メソッドを参照します。アクションが指定されていない場合、ページは単に更新されます。標準コントローラで save、edit、または delete アクションに関連付けられているコマンドリンクは、ユーザに適切な権限がある場合にのみ表示されます。同様に、edit アクションおよび delete アクションに関連付けられているコマンドリンクは、レコードがページに関連付けられている場合にのみ表示されます。		10.0	global
ch	String	指定 URL の符号化に使用される文字セット。指定されていない場合、この値はデフォルトの「ISO-8859-1」に設定されます。		10.0	global
cd	String	コマンドリンクに使用される画面のホットスポットの位置と形状(クライアント側の画像マップ用)。カンマ区切り値の数および順序は定義される形状に依存します。たとえば、長方形を定義するには、coords="left-x, top-y, right-x, bottom-y" を使用します。円形を定義するには、coords="center-x, center-y, radius" を使用します。多角形を定義するには、coords="x1, y1, x2, y2, ..., xN, yN" を使用します。ここで、x1 = nN および y1 = yN です。座標はピクセルまたはパーセントで表すことができます。また、座標は対応付けられる画像の左上からの距離を表します。「shape 属性」も参照してください。		10.0	global
dr	String	生成された HTML コンポーネントの読み取り方向。使用可能な値には「RTL」(右から左) または「LTR」(左から右) があります。		10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
language	String	「en」、「en-US」など、このコマンドリンクで参照されるリソースの基本言語。この属性についての詳細は、W3C仕様を参照してください。	10.0	global	
languageId	String	ページの他のコンポーネントが commandLink コンポーネントを参照できるようにする識別子。	10.0	global	
skipInputValidation	Boolean	ページの項目に関連付けられている入力規則を処理することなく、このコンポーネントに関連付けられているアクションをすぐに実行するかどうかを指定する boolean 値。true に設定すると、アクションがすぐに実行され、入力規則はスキップされます。指定されていない場合、この値はデフォルトの false に設定されます。	11.0	global	
languageAttribute	String	「en」または「en-US」など、生成された HTML 出力の基本言語。この属性についての詳細は、W3C仕様を参照してください。	10.0	global	
onBlur	String	onblur イベントが発生した場合(フォーカスがコマンドリンクから離れた場合)に呼び出される JavaScript。	10.0	global	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
onmousemove	String	s onmousemove イベントが発生した場合(ユーザがマウス pointer を移動した場合)に呼び出される JavaScript。		10.0	global
onmouseout	String	s onmouseout イベントが発生した場合(ユーザがマウス click からマウス pointer を移動した場合)に呼び出される JavaScript。		10.0	global
onmouseover	String	s onmouseover イベントが発生した場合(ユーザがマウス pointer をマウス click に重ねた場合)に呼び出される JavaScript。		10.0	global
onmouseup	String	s onmouseup イベントが発生した場合(ユーザがマウス button を放した場合)に呼び出される JavaScript。		10.0	global
url	String	現在のドキュメントからこの commandLink で指定される URL へのリレーション。この属性の値は、リンクタイプのスペース区切りのリストです。この属性についての詳細は、 W3C仕様 を参照してください。		10.0	global
rendered	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		10.0	global
renderer	Object	R AJAX 更新要求の結果がクライアントに返されるときに再作成される 1 つ以上のコンポーネントの ID。この値には、単一の ID、ID のカンマ区切りのリスト、または ID のリストまたはコレクションの差し込み項目の式を使用できます。		10.0	global
reverse	String	v 現在のドキュメントへのこの commandLink で指定される URL からの逆リンク。この属性の値は、リンクタイプのスペース区切りのリストです。この属性についての詳細は、 W3C仕様 を参照してください。		10.0	global
shape	String	s a クライアント側の画像マップのホットスポットの形状。有効な値は、default、circle、rect、および poly です。「coords 属性」も参照してください。		10.0	global
status	String	s sa AJAX 更新要求の状況を表示する関連付けられているコンポーネントの ID。「actionStatus コンポーネント」を参照してください。		10.0	global
style	String	s y commandLink コンポーネントの表示に使用されるスタイル。主に、インライン CSS スタイルを追加するために使用されます。		10.0	global
styleClass	String	s y c commandLink コンポーネントの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシート		10.0	global

属性名	属性型	説明	必須項目	API バージョン	アクセス
style	String	使用するときに適用される CSS スタイルを指定するために使用されます。			
index	String	ユーザが Tab キーを繰り返し押したときに、他のページコンポーネントと比較して、このリンクが選択される順序。この値は、ユーザが Tab キーを押したときに選択される最初のコンポーネントを 0 として、0 ~ 32767 の整数である必要があります。		10.0	global
resource	String	このコマンドリンクによって取得されるリソースを表示するフレームの名前。この属性に使用できる値には、「_blank」、「_parent」、「_self」、「_top」があります。また、目的の移行先の name 属性に値を割り当てることにより、独自のターゲット名を指定することもできます。		10.0	global
timeout	Integer	AJAX 更新要求がタイムアウトするまでの時間(ミリ秒)。		10.0	global
label	String	ユーザがコンポーネントにマウスポインタを重ねたときにツールチップとして表示されるテキスト。		10.0	global
type	String	このコマンドリンクで指定されるリソースの MIME コンテンツタイプ。この属性の使用できる値には、「text/html」、「image/png」、「image/gif」、「video/mpeg」、「text/css」、および「audio/basic」があります。使用できる値の完全なリストなど、詳細は、W3C 仕様を参照してください。		10.0	global
value	Object	commandLink ラベルとして表示されるテキスト。 commandLink タグの本文にコンテンツを埋め込むことにより、コマンドリンクとして表示するテキストまたは画像を指定することもできます。 value 属性および埋め込みコンテンツの両方が指定されると、これらは一緒に表示されます。		10.0	global

apex:component

カスタム Visualforce コンポーネントです。すべてのカスタムコンポーネントの定義は 1 つの `apex:component` タグ内でラップされている必要があります。

このコンポーネントでは、「html-」プレフィックスを使用した [HTML パススルー属性](#) がサポートされています。`apex:component` 属性は、`apex:repeat` 属性に応じて、生成されたコンテナタグである `apex:repeat` または `apex:repeat` に適用されます。

g : < ! -- Pa >-

```

g p p < a a
e t n n e m o m: C e l m p t n n c o o n e M y c V l p b r g r a l l o r o y r d C v ' ' a
g p p < / a a

t n n @ o m t n G m p m: < ! p -- y >-
t n n o o m p p < a >
x t t e t e m b e u p m b e u d n r a e t a o e v p a e t h y l h o t n i n f a i n s i s v p a
t e n e r e t r g p n d r u y " i s " i = " " > /
x t t e t e m b u p e b m r d r < b o r o d a n r i a o c a e t p b o r o r o c s b r d f i s i s
t e n e r e t r g p n d r u y " i s " i = " " > /
e { l e h r d r b r d n b o r o s c y " ! " >
x t e t t x e f u u p l u p < T l m u a v v a " ! y a " > /
h < 1 >
t n n o o m p p < / a >

```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
cc	String	コンポーネントを同じ名前空間の任意のページ外でコンポーネントとして使用できるかどうかを示す。使用できる値は、「public」(デフォルト)および「global」です。コンポーネントをコンポーネントの名前空間外で使用できるように指定する場合は、globalを使用します。アクセス属性がglobalに設定されている場合は、すべての必須の子 apex:attributes のアクセス属性も global に設定する必要があります。アクセス属性が public に設定されている場合は、子 apex:attributes のアクセス属性を global に設定することはできません。注意: この指定があるコンポーネントには、管理パッケージの説明にある非推奨ポリシーが適用されます。		14.0	
dllo	Boolean	この属性が「true」に設定されている場合、コンポーネント内に DML を指定できる。デフォルトは「false」です。DML を使用することにより、コンシューマがコンポーネントを使用してページの部分更新を行ううえで問題となり得る副次的影响がある場合があります。コンポーネント内で DML を使用する場合は、コンシューマがページを適切に更新できるように rerender 属性を指定する必要があります。また、コンポーネントの説明で DML が処理するデータの種類の詳細を示して、発生す		13.0	global

属性名	属性型	説明	必須項目	API バージョン	アクセス
label	String	この可能性のある副次的影響をコンポーネントのコンシューマが認識できるようにする必要があります。		12.0	global
name	String	このカスタムコンポーネントに追加ロジックを追加する 1 つ以上のコントローラ拡張の名前。		12.0	global
id	String	コンポーネント定義で他のタグがコンポーネントを参照できるようにする識別子。		12.0	global
language	String	関連付けられている翻訳が Salesforce にあるラベルの表示に使用される言語。この値はコンポーネントを参照しているユーザの言語より優先されます。この属性に使用できる値には、「en」、「en-US」などの Salesforce でサポートされている言語の言語キーがあります。		12.0	global
layout	String	コンポーネントの HTML レイアウトスタイル。使用できる値には、「block」(HTML div タグでコンポーネントをラップする)、「inline」(HTML span タグでコンポーネントをラップする)、および「none」(生成される HTML タグでコンポーネントをラップしない)があります。指定されていない場合、この値はデフォルトの「inline」に設定されます。		25.0	
rendered	Boolean	カスタムコンポーネントを表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの「true」に設定されます。		12.0	global
isLast	Boolean	Visualforce エディタでこのコンポーネントを終了する方法を指定する boolean 値。この属性が「true」に設定されている場合、Visualforce エディタがコンポーネントを自己終了タグとしてオートコンプリートします。「true」に設定されていない場合、開始タグおよび終了タグでコンポーネントをオートコンプリートします。たとえば、この属性が myComponent という名前のコンポーネントで「true」に設定されている場合、エディタはこのコンポーネントを <c:myComponent/> のようにオートコンプリートします。「false」に設定されている場合、エディタはこのコンポーネントを <c:myComponent></c:myComponent> のようにオートコンプリートします。コンポーネントに componentBody が含まれている場合、この属性のデフォルトは「false」です。コンポーネントに		15.0	

属性名	属性型	説明	必須項目	API バージョン	アクセス
componentBody が含まれていない場合は、この属性のデフォルトは「true」です。					

apex:componentBody

このタグを使用して、カスタムコンポーネントの作成者は、ユーザがカスタムコンポーネントにコンテンツを挿入できる場所を定義できます。これは、特にカスタム反復コンポーネントを生成する場合に役立ちます。このコンポーネントは apex:componentBody タグ内でのみ有効であり、カスタムコンポーネントあたり 1 つの定義のみを使用できます。

単純な例

```

g : < ! -- Pa >-
g p p < a >
extet txe ou ue p l upe <T get a ip thro v ET    aT noh oe omsi s br p     s      " < / > /
ex b do d: m E p < y a >
extet txe ou ue p l upe <T greta nlp h v Te a"u ndm scipsmi sc br p     s      " < / > /
ex b do d: m E p < / y a >
extet txe ou ue p l up e<T gt a pt hr veT   tE hom( ocaom) sicsbr pa     s      " < / > /
g p p < /a >
tnn o em kdo p l n E! -- p           y a >-
tnn o om p p < a >
extet txe ou u p l tup r<T t n et mc om v cou a'p sbr sp     " < / > /
x e tnn o Bm p do p < a > y > /
extet txe ou e u p l ntp <T t aon mc omv cou a'up Sbr sp     " < / > /
tnn o om p p < /a >

```

高度な例

```

g : < ! -- Pa >-
g p p < a a>
tn n mouc: rcc < y a s = a >a
ex n l r d p l opu m c b rd r a2 aiG s= " " 1 = " " >
extet txe fu u ep l nup } <T ma v a! . a 'a > /
ex n l r o p p < p a a G >
extet txe fu u p l up tn &Tll jarg v a! i ai s " > /
ex n l r o p p < p a a G >
{ et b ll } g n kell oglia id p sc a! i ai s a " > /
ex n l r d p p < p /a a G >
ex n l r o p p < p /a a G >
tn n mouc: cc < / y a s >
g p p < /a >
tnn o o m p u: < cc ! -- y a s >-
tn n nom p or lptr & m ou occ C = y s " >
extet m b utp tmr e< n rat ied gnr eva de ae rep blis 'T rs i i = p i a a
e n t leat gen tout cc nr hsa a i i i a ". > /
x t re tripn & p {< ou pa ltcuwn a = aou cc "v a! a s " >
x e tnn o Bm p do p < a >
e x e { r bl p {} r & l t man i&ama &c= aul p vac a! " > /

```

```

ex x e t n n o Bm pdo p < /a /a >a y
      t r o om p p < /a /a >a

n or l@ r C *** / ***
  blt {An p c m ou occ iC ss a y s

t blt An [ptcoun ccou c@ i<s > a s
g

e due acl tn bndl stag b il s [eetgn b laj i r g m keyll i sogl alp i p sc
  te rem em oufn rec or kl net ur oa gr d f i = ] s i . s i ( ; )

n rt rm ou cc a s;
}

}

```

上述の例では次の HTML を表示します。

```

ble d < c g alpodd" c "Go nsrie a=d" b @o eblt a @" " 0 o@ " " i 0=" y "a ss
te e tr tn go bid< ! -- S a a a >-
  t b do er d d b do ll <tl <y<c> gi =" h y " ss=a R " >
  t te e tr tn go pc< ! -- S a a >-
  e n n m lwt e nt<k la #asig h <"hi i 1 @" " i 1 @" "
  t et et tnln te @ ic r lin=d H sap k" kss=asi i "
  t et tnln te @ ic r r =a g n H sad " sj=p dji/ "< :>/ >/sa i a=" i

t e bl b rd r < a 1 @" >
  t b do r < >
  t t e t d ro d c < >
  t te d bl < < > < a>
  t t b do b do r < >
  t t e t r dn ed enrtTt Mf < >
  t t r n d n r < >
  t t r b do bl < >
  t t r b do d < >
  t t e n t d t r < >
  t t bl b do r < >
  t t e e t ml u Ao < >
  t t e t n et t d < >
  t t r r < >
  t t r d t u Ao < >
  t t bl b do r < >
  t t d r < >
  t t b do bl < >

```

n	p < s / >a
d	< >
r	< >
b do	< / >y
bl	< / a>

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
i	String	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		13.0	global
isDisplayed	Boolean	コンポーネントをページに表示するかどうかを指定するboolean値。指定されていない場合、この値はデフォルトの true に設定されます。		13.0	global

apex:composition

2番目のテンプレートページのコンテンツを含むページの領域です。テンプレートページは、1つ以上のコンポーネントを含む Visualforce ページです。コンポーネントは関連付けられているテンプレートを指定し、テンプレートのコンポーネントの本文に一致するコンポーネントを提供します。コンポーネント外のコンテンツは表示されません。

pn d、< p はも参照してください。>

例

t g a om : o p ! -- Pa si >-	a . .
t ete thg e tt ep e<t l !th -ir\$et patGaes ea f w hab gh oia pi s , a . .	a . .
t et txe oueut p i up <d me aete p vho Ta" dfr h hr i si s "k > / > /	"k > / > /
tne ener p mdr <br á s =a "k > / > /	a a "k y > / > /
t et txe oueut p i up <d ext aweterpe h Ta" n(d r h dhab)si s br g p p < / a >	a a "k y > / > /
t re n r p bmdo < á s =a "y > /	a a "k y > / > /
e g g : p < ! -- Pa a >-	a a "k y > / > /
t et ete ne dm p lpm < a o a o si p =a si " >	a a "k y > / > /
e nede ep mf d r <te ga i p he =a et h e@h@e (m agif s d sp p af y a /a i >	a a "k y > / > /
e n d e p bmdo et g a i pe hb de=et en Xe@m(gaf i) si ps p fy y a /a i >	a a "k y > / > /
t a om p < / a > s i >	a a "k y > / > /
g p p < / a >	a a "k y > / > /

上述の例では次の HTML を表示します。

ete p hro T dfr hr ha i si s < a > /
ete g p le d T hoh (m bagif si sp a > /
et et weterpe h T n(d r h dhab)si ps a < a > /
et g pe hb do T dn m(gaf i) si ps y y a > /

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
	String	<p>このコンポーネントの表示には、この属性による影響はありません。</p> <p>基づいて表示する場合は、コンポーネント内にラップし、その属性に条件式を追加します。</p>		10.0	global
p	ApexPagesPageReference	このコンポーネントで使用するテンプレートページ。この値では、Visualforce ページの名前を指定するか、または差し込み項目の構文を使用して、ページまたは PageReference を参照します。	はい	10.0	global

apex: dataList

一連のデータを反復処理することで定義される、値の順序付きリストと順序なしリストです。

コンポーネントの本文では、リストでの1つの項目の表示方法を指定します。データセットには最大1,000個の項目を含めることができます。

例

g : < ! -- Pa >-
te n dor pl& ipcd n < o a ac = L iasa " >
x t t e d plt un } < ou t a&n iassou abc ! a s'v = a a " >
x tet txe fu u p lte upn < To uj a cc m v a! a N 'a > /
x t t d p < /aL iasa >
g p p < /a a

n b& or ld r C *** : / *** /
b& t{ d p n c o C i ss & iasa

t An t o un ccou c& i<s > a s;

t bl& te Ant p co t{ An ccg ou L ccis > s ()
t n ou f & cl in oue i cc (a s = t E & nC) & au & = L C [S INMI & R Q L 1 ; 0
n rt rm ou cc a s;

上述の例では次の HTML を表示します。

e laudt t g h :h< i = " Ba is " >
e letd t g h B :h< i inu rum Ba gi s " > 0 ss a a a< i >
e letd t g h B :h< l i ro =l C Ba pil s " > a< i >
e letd t e egh :h< liri 2 =b l Ba Wi s " > ss < . i >
lu < >

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
direction	String	i 生成された HTML コンポーネントの読み取り方向。使用可能な値には「RTL」(右から左) または「LTR」(左から右) があります。		10.0	global
first	Integer	i s リストに視覚的に表示される反復の最初の要素。ここで、0 は value 属性で指定されている一連のデータの最初の要素のインデックスです。たとえば、value 属性で指定されている一連のレコードの最初の 2 つの要素を表示しない場合は、first="2" と設定します。		10.0	global
dataId	String	i ページの他のコンポーネントが dataList コンポーネントを参照できるようにする識別子。		10.0	global
language	String	a 「en」または「en-US」など、生成された HTML 出力の基本言語。この属性についての詳細は、 W3C 仕様 を参照してください。		10.0	global
click	String	i onclick イベントが発生した場合(ユーザがリストをクリックした場合)に呼び出される JavaScript。		10.0	global
dblClick	String	i ondblclick イベントが発生した場合(ユーザがリストをダブルクリックした場合)に呼び出される JavaScript。		10.0	global
keydown	String	k yonkeydown イベントが発生した場合(ユーザがキーボードのキーを押した場合)に呼び出される JavaScript。		10.0	global
keypress	String	p k yonkeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出される JavaScript。		10.0	global
keyup	String	p k yonkeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出される JavaScript。		10.0	global
mousedown	String	s onmousedown イベントが発生した場合(ユーザがマウスボタンをクリックした場合)に呼び出される JavaScript。		10.0	global
mousemove	String	s onmousemove イベントが発生した場合(ユーザがマウスポインタを移動した場合)に呼び出される JavaScript。		10.0	global
mouseout	String	s onmouseout イベントが発生した場合(ユーザがリストからマウスポインタを移動した場合)に呼び出される JavaScript。		10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
onmouseover	String	イベントが発生した場合(ユーザがマウスポインタをリストに重ねた場合)に呼び出されるJavaScript。		10.0	global
onmouseup	String	イベントが発生した場合(ユーザがマウスボタンを放した場合)に呼び出されるJavaScript。		10.0	global
rendered	Boolean	コンポーネントをページに表示するかどうかを指定するboolean値。指定されていない場合、この値はデフォルトのtrueに設定されます。		10.0	global
size	Integer	リストに表示する項目の最大数。指定されていない場合、この値はデフォルトの0に設定され、表示可能なすべてのリストの項目を表示します。		10.0	global
style	String	dataListコンポーネントの表示に使用されるスタイル。主に、オンラインCSSスタイルを追加するために使用されます。		10.0	global
styleClass	String	dataListコンポーネントの表示に使用されるスタイルクラス。主に、外部CSSスタイルシートを使用するときに適用されるCSSスタイルを指定するために使用されます。		10.0	global
title	String	ユーザがコンポーネントにマウスポインタを重ねたときにツールチップとして表示されるテキスト。		10.0	global
type	String	表示するリストの種類。順序付きリストでは、使用できる値には「1」、「a」、「A」、「i」、または「I」があります。順序なしリストでは、使用できる値には「disc」、「square」、および「circle」があります。指定されていない場合、この値はデフォルトの「disc」に設定されます。		10.0	global
value	Object	リストに表示されるデータのコレクション。	はい	10.0	global
valueLabel	String	value属性で指定するデータのコレクションの1つの要素を表す変数の名前。この変数を使用して、dataListコンポーネントタグの本文に要素を表示できます。	はい	10.0	global

apex:dataTable

一連のデータを反復処理することで定義されるHTMLテーブルです。1行あたり1つのデータ項目に関する情報を表示します。この本文には、データの各項目で表示する情報の種類を指定する1つ以上の列コンポーネントが含まれます。データセットには最大1,000個の項目を含めることができます。

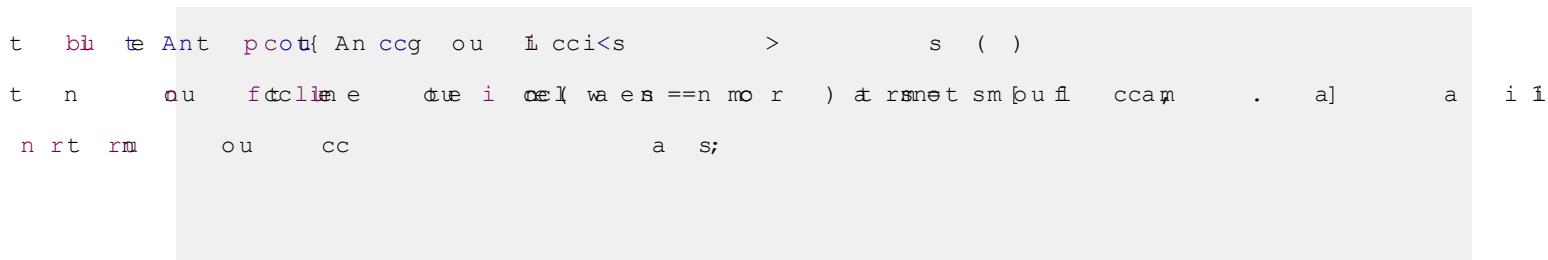
Salesforce.com APIバージョン20.0以降が稼動しているVisualforceページでは、このコンポーネントにpタグを含めて列を生成できます。

l r d p p も参照してください。 [\[G\]](#)

このコンポーネントでは、「`html-`」プレフィックスを使用した [HTML パスルー属性](#)がサポートされています。パスルー属性は、生成されたテーブルの [タグ](#)に適用されます。

ex ro te e l m o d r p fil i se optan o e ec in pro y s sca V pa is a
et l d h eturn ad i cv a ha a i U L R .
ro l m t ept f F t rete iqulit dd 0000003 g ddB h , s: U L R s
e he n erap xe fc sc m 06a i s a /a I y EalP 00 000R003 5
e e el hro e l fo V t Sdt ips d secur o l sn t@soTm info mo a f i a i a
g : < ! -- Pa >-
te n go epld pcd nble oad a T g# a a a "i =" "Pa>
e ed{ bl plt un T< ou tacn e at ave u datc !ve ba sihel ="a T addi =" "a ss a="" v
et te l l bl cl C s y ss=a a ss" a>
te n t tem fm to <bl n px to a c =a p i a " p f a i<a /a a>
te en e te p fm e r ebl ex dat ah =a "p>f a < a /a a>
te ne t te p fm eot ebl f ex oot a f =a "p>f a < /a a>
n lo u mp < a >
ex te en e e p fm d r ex kt m a a p f "a N< a /a a>
te ne t p fm eor n do fu ex eot a f =a " p f < /a a>
tet txe ph u u p lt apn n<T o q acc m v a! ! a . "a > /
n lo u mp < /a >
n lo u mp < a >
te en e e p fm e r ex kt a a a =p f "a >0< /a a>
te ne t p fm eor n do fu ex eot a f =a " p f < /a a>
tet txe ph u u p lt apn w cfil o un a q c v m a! ! a . . "a > /
n lo u mp < /a >
e d bl p t /a a a a>
g p p < /a a>
n or ld r C *** : / *** /

bh e l { d p cnc bl o G ss a a a a



上述の例では次の HTML を表示します。

```


|                                           |                                                                         |                    |
|-------------------------------------------|-------------------------------------------------------------------------|--------------------|
| blu te Ant pco{ An ccg ou L cci<s > s ( ) | n ou fcline due i cel wa es ==n mo r ) & rsnet sm[ou fl ccam . a] a i 1 | n rt rm ou cc a s; |
|                                           |                                                                         |                    |


```

```

nA dt lB    ll d      <      >      & a &gt;
r           <      &gt;
r l       dd c      <      ss=a " >
d t l r o d   C W<      > ss     < . &gt;
e d   blJ l r d   < W &gt;     ss<      &gt;
r           <      &gt;
b do       <      / &gt;
bl         <      / a>

```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
l g	String	i a 表示される HTML テーブルのページに対する位置。使用できる値には、「left」、「center」、または「right」があります。指定されていない状態になると、この値はデフォルトの「left」に設定されます。		10.0	global
b logo	c	String 表示される HTML テーブルの背景色。		10.0	global
b rd r		String 表示される HTML テーブルの周囲のフレームの幅(ピクセル単位)。		10.0	global
n ol	c p C	String i a caption facet が指定されている場合、表示される HTML テーブルのキャプションの表示に使用されるスタイルクラス。この属性は、主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。		10.0	global
n o	lc p	String i a caption facet が指定されている場合、表示される HTML テーブルのキャプションの表示に使用されるスタイル。この属性は、主に、インライン CSS スタイルを追加するために使用されます。		10.0	global
hl dd	c g p	String 各テーブルセルの境界線とそのコンテンツ間のスペース。この属性の値がピクセル単位の長さである場合、4辺の余白のすべてでコンテンツからの距離がこの値に設定されます。属性の値がパーセント単位の長さである場合は、上下の余白は利用可能な縦方向のスペースのパーセントに基づいてコンテンツからの距離が上下均等の長さに分割され、左右の余白は利用可能な横方向のスペースのパーセントに基づいてコンテンツからの距離が左右均等の長さに分割されます。		10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス		
border	c g c p	String	s	各テーブルセルの境界線とそのセルを囲む他のセルの境界線および/またはテーブルの境界の間のスペース。この値はピクセルまたはパーセント単位で指定する必要があります。	10.0	global	
columnClasses	c	String		テーブルの列に関連付けられている 1 つ以上のクラスのカンマ区切りのリスト。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。複数のクラスが指定されている場合、クラスはすべての列に繰り返し適用されます。たとえば、columnClasses="classA, classB" と指定すると、最初の列は classA でスタイル設定され、2 番目の列は classB でスタイル設定され、3 番目の列は classA でスタイル設定され、4 番目の列は classB でスタイル設定されます (以下同様)。	10.0	global	
columns	m c	Integer	s	このテーブルの列数。	10.0	global	
columnsWidth	d c	h	String	s	各テーブル列に適用される幅のカンマ区切りのリスト。値はピクセルで表すことができます (columnsWidth="100px, 100px" など)。	10.0	global
direction			String	i	生成された HTML コンポーネントの読み取り方向。使用可能な値には「RTL」(右から左) または「LTR」(左から右) があります。	10.0	global
first	f		Integer	i s	テーブルに視覚的に表示される反復処理の最初の要素。ここで、0 は value 属性で指定されている一連のデータの最初の要素のインデックスです。たとえば、value 属性で指定されている一連のレコードの最初の 2 つの要素を表示しない場合は、first="2" と設定します。	10.0	global
footerLabel	f C	String		footerfacet が指定されている場合、表示される HTML テーブルのフッター (一番下の行) を表示するために使用されるスタイルクラス。この属性は、主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。	10.0	global	
frame	f	String	a	このテーブルに引かれる境界線。使用できる値には、「none」、「above」、「below」、「hsides」、「vsides」、「lhs」、「rhs」、「box」、および「border」があります。指定されていない場合、この値はデフォルトの「border」に設定されます。	10.0	global	
headerLabel	h C	String	a	headerfacet が指定されている場合、表示される HTML テーブルのヘッダーの表示に使用されるスタイルクラス。この属性は、主に、外部 CSS スタ	10.0	global	

	属性名	属性型	説明	必須項目	API バージョン	アクセス
d	g	String	i イルシートを使用するときに適用される CSS スタイルを指定するために使用されます。			
l	g	String	i ページの他のコンポーネントが dataTable コンポーネントを参照できるようにする識別子。	10.0	global	
ol	c c	String	a i 「en」または「en-US」など、生成された HTML 出力の基本言語。この属性についての詳細は、 W3C 仕様 を参照してください。	10.0	global	
dbl l	c c	String	i onclick イベントが発生した場合(ユーザがデータテーブルをクリックした場合)に呼び出される JavaScript。	10.0	global	
no d o	k	String	i ondblclick イベントが発生した場合(ユーザがデータテーブルをダブルクリックした場合)に呼び出される JavaScript。	10.0	global	
o r	p k	String	y onkeydown イベントが発生した場合(ユーザがキーボードのキーを押した場合)に呼び出される JavaScript。	10.0	global	
o u	p k	String	y onkeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出される JavaScript。	10.0	global	
w on o und o	p k	String	y onkeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出される JavaScript。	10.0	global	
o o um o m	String	s onmousedown イベントが発生した場合(ユーザがマウスボタンをクリックした場合)に呼び出される JavaScript。	10.0	global		
o o um o u	String	s onmousemove イベントが発生した場合(ユーザがマウスポインタを移動した場合)に呼び出される JavaScript。	10.0	global		
o o um o r	String	s onmouseout イベントが発生した場合(ユーザがデータテーブルからマウスポインタを移動した場合)に呼び出される JavaScript。	10.0	global		
o o um u	String	s onmouseover イベントが発生した場合(ユーザがマウスポインタをデータテーブルに重ねた場合)に呼び出される JavaScript。	10.0	global		
o ol	p	String	s onmouseup イベントが発生した場合(ユーザがマウスボタンを放した場合)に呼び出される JavaScript。	10.0	global	
	c c	String	R i onRowClick イベントが発生した場合(ユーザがデータテーブルの行をクリックした場合)に呼び出される JavaScript。	10.0	global	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
onRowDblClick	String	onRowDblClick イベントが発生した場合(ユーザがデータテーブルの行をダブルクリックした場合)に呼び出される JavaScript。	10.0	global	
onRowMouseDown	String	onRowMouseDown イベントが発生した場合(ユーザがデータテーブルの行でマウスボタンをクリックした場合)に呼び出される JavaScript。	10.0	global	
onRowMouseMove	String	onRowMouseMove イベントが発生した場合(ユーザがマウスポインタをデータテーブルの行に重ねた場合)に呼び出される JavaScript。	10.0	global	
onRowMouseOut	String	onRowMouseOut イベントが発生した場合(ユーザがデータテーブルの行からマウスポインタを移動した場合)に呼び出される JavaScript。	10.0	global	
onRowMouseOver	String	onRowMouseOver イベントが発生した場合(ユーザがマウスポインタをデータテーブルの行に重ねた場合)に呼び出される JavaScript。	10.0	global	
onRowMouseUp	String	onRowMouseUp イベントが発生した場合(ユーザがデータテーブルの行の上でマウスボタンを放した場合)に呼び出される JavaScript。	10.0	global	
rowClasses	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	10.0	global	
rowClassNames	String	テーブルの行に関連付けられている 1 つ以上のクラスのカンマ区切りのリスト。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。複数のクラスが指定されている場合、それらのクラスがすべての行に繰り返し適用されます。たとえば、columnRows="classA, classB" と指定すると、最初の行は classA でスタイル設定され、2 番目の行は classB でスタイル設定され、3 番目の行は classA でスタイル設定され、4 番目の行は classB でスタイル設定されます(以下同様)。	10.0	global	
rowCount	Integer	このテーブルの行数。	10.0	global	
rowSeparator	String	このテーブルのセルの間に引かれる境界線。使用できる値には、「none」、「groups」、「rows」、「cols」、および「all」があります。指定されていない場合、この値はデフォルトの「none」に設定されます。	10.0	global	
style	String	dataTable コンポーネントの表示に使用されるスタイル。主に、INLINE CSS スタイルを追加するために使用されます。	10.0	global	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
c	String	コンポーネントの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。	10.0	global	
s	String	セクション 508 の準拠に関するテーブルの目的と構造の概要。	10.0	global	
i	String	ユーザがコンポーネントにマウスポインタを重ねたときにツールチップとして表示されるテキスト。	10.0	global	
a	Object	テーブルに表示されるデータのコレクション。	はい	10.0	global
v	String	value 属性で指定するデータのコレクションの 1 つの要素を表す変数の名前。この変数を使用して、dataTable コンポーネントタグの本文に要素自体を表示できます。	はい	10.0	global
h	String	テーブル全体の幅。利用可能な横方向の合計スペースに対する相対パーセント (width="80%" など)、またはピクセル数 (width="800px" など) のいずれかとして表されます。	10.0	global	

Facet

facet名	説明	APIバージョン
c	テーブルのキャプションに表示されるコンポーネント。テーブルのキャプションの表示は、name="caption" を含む facet によって制御されるため、caption facet が dataTable コンポーネントの本文に表示される順序は重要ではありません。	10.0
f	テーブルのフッター行に表示されるコンポーネント。テーブルの最後の行の表示は、name="footer" を含む facet によって制御されるため、footer facet が dataTable コンポーネントの本文に表示される順序は重要ではありません。	10.0
h	テーブルのヘッダー行に表示されるコンポーネント。テーブルの最初の行の表示は、name="header" を含む facet によって制御されるため、header facet が dataTable コンポーネントの本文に表示される順序は重要ではありません。	10.0

apex:define

Visualforce テンプレートページで定義されているコンポーネントです。

コンポーネントのコンテンツを提供するテン

t e a orex po tn p < r p \$ **も参照してください。** >

例

```
t g a om : o p ! -- Pa si >
t ete thg e tT ep e<t l !tm -ir$et patGae sea f w hab g oia pi s ,
a . >
x tet txe oueut p i up <@ ne aete p o vro T@' dfr h hr i si s 'k > / > /
x tne ener r p m dr <hr a i s =a 'k > / > /
x tet txe oueut p i up <@ est awetephe vh T@' n(d r h ddbi)si s br a a "k y > /
x tne n r p bmdo < a i s =a "y > /
g p p < / a >
```

```
e g g : p < ! -- Pa a >
x t e t ae dm p lpm < a oga o $ ip =a si " >
e xe ene ep md r <ente ga i p ke =a et hehne (m agif sds sp p af y a /a i
e xe en d e p bmdo et g a ipe hb do=t en kym( gaf ips p fy y a /a i
x t a om p < /a >
g p p < /a >
```

上述の例では次の HTML を表示します。

```
d ne ete p o vro T dfr hr ha i si s < a > /
ete g p ke d T hoh (m bagif si sp a y a > /
d est wetephe h T n(d r h ddbi)si s a a < y > /
et g pe hb do T dn m( gaf i) si ps y y a
```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
m	String	a この定義コンポーネントのコンテンツを挿入する はい 挿入コンポーネントの名前。	はい	10.0	global

apex:detail

[設定] でオブジェクトの関連付けられているページレイアウトで定義する、特定のオブジェクトの標準詳細ページです。このコンポーネントには、標準の Salesforce アプリケーションインターフェースに表示する、関連付けられている関連リスト、関連リストのフロート表示リンク、およびタイトルバーの包含または除外に関する属性が含まれます。

例

```
e ex ro te e lhm o d r p Btl i seoptpam o e ec li byoy s sc V p i s a
e et l d h eturn cd i cv a ha a i U L R .
ro l m t ept f F t rete iqulit 00 00000035 g daDb h , s: U L R s
e he n erap xe fc sc m 05pa i s a /a I /y Ra1 00 000R003 5
e e eli hro e l fr V tSdt ips a secur o l statm info m a f i a i
t n tegndprd por h a n C au acc a a " " >
t e at l bptejnw neouetacd i r d te't ! lea . l f "1 L a s =" s a i =" s a >
```

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
detailId	String	ページの他のコンポーネントが詳細コンポーネントを参照できるようにする識別子。		10.0	global
inlineEdit	Boolean	コンポーネントでオンライン編集をサポートするかどうかを制御する。 editMode と inlineEdit の値が両方とも true の場合、editMode の値が優先されます。		20.0	
oncomplete	String	oncomplete イベントが発生した場合(タブが選択されており、ページにそのコンテンツが表示されている場合)に呼び出される JavaScript。 この属性は、inlineEdit または showChatter が true に設定されている場合にのみ機能します。		20.0	
relatedList	Boolean	表示されるコンポーネントに関連リストを含めるかどうかを指定する boolean 値。true の場合、関連リストが表示されます。指定されていない場合、この値はデフォルトの true に設定されます。		10.0	global
relatedListHiLight	Boolean	表示されるコンポーネントに関連リストのフロート表示リンクを含めるかどうかを指定する boolean 値。true の場合、関連リストのフロート表示リンクが表示されます。指定されていない場合、この値はデフォルトの true に設定されます。relatedList 属性が false であるか、または [設定] [カスタマイズ] [デザイナーアクション] [関連リストのフロート表示リンクを有効化] オプションが選択されていない場合、この属性は無視されます。		10.0	global
rendered	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		10.0	global
renderedFor	Object	AJAX 更新要求の結果がクライアントに返されるときに再作成される 1 つ以上のコンポーネントの ID。 この値には、単一の ID、ID のカンマ区切りのリスト、または ID のリストまたはコレクションの差し込み項目の式を使用できます。 この属性は、inlineEdit または showChatter が true に設定されている場合にのみ機能します。		20.0	
showChatter	Boolean	Chatter 情報とレコードのコントロールを表示するかどうかを指定する boolean 値。		20.0	

属性名	属性型	説明	必須項目	API バージョン	アクセス
e	Boolean	これが true で、 <code>g p p</code> の <code>showHeader</code> が false である場合は、 <code>headerFor h < w a i f s ></code> が使用されている場合とまったく同様のレイアウトになります。			
e	Boolean	これが true で、 <code>g p p</code> の <code>showHeader</code> が true である場合は、通常の Chatter UI のようなレイアウトになります。			
bu	String	このコンポーネントのデータを提供するレコードの ID。		10.0	global
l	Boolean	表示されるコンポーネントにタイトルバーを含めるかどうかを指定する boolean 値。true の場合、タイトルバーが表示されます。指定されていない場合、この値はデフォルトの true に設定されます。		10.0	global

apex:dynamicComponent

このタグは動的 Apex コンポーネントのプレースホルダとして機能します。動的コンポーネントを返す Apex メソッドの名前を受け入れる 1 つの必須パラメータ (`name`) が含まれます。

次の Visualforce コンポーネントには、動的な Apex 表現は含まれません。

```
<apex:dynamicComponent name="myComponent">
</apex:dynamicComponent>
```

例

```
<apex:dynamicComponent name="myComponent">
</apex:dynamicComponent>
```

c	e	tn	r	ru	d	l	i	;	a												
t	t	et	mJr	tru	t	rAen	h	exuf	/sec	l	mfrm	os									
t	bIAnt	oup	c	cc	cc	i	a			p	p	,									
t	e	ng	r	true	e	@	RE	EC	■	rTdt	AnL	ISou	M	Nic	a,O	IMI	FR	©	L	1	;

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
lCu	UIComponent	動的 Visualforce コンポーネントを返す Apex メソッドの名前を受け入れる。	はい	22.0	
i	String	カスタムコンポーネント定義で他のタグが属性を参照できるようにする識別子。		22.0	global
isDisplayed	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		22.0	

apex:emailPublisher

apex:emailPublisher (メールパブリッシャー) では、ケースフィールドを使用するサポートエージェントは、メールメッセージを作成し、顧客に送信できます。メールテンプレートと添付ファイルをサポートできるように、このコンポーネントをカスタマイズできます。このコンポーネントは、ケースフィールドとメール-to-ケースが有効化された組織でのみ使用できます。このコンポーネントを使用するページには、バージョン 3 より前の Ext JS を含めることはできません。

次の例はメールパブリッシャーを表示します。

e	e	ex	t	n	te	gn	pr	de	po	rl	de	es	a	"	s	"	a	"	"	>	
e	e	x	e	l	mbh	p	r	d	h<	ml	mbia	Eir	a	iPsh	i	"	yi	a	iPs	"	
e	t	t	h	e	d	}	I	d	c	i	i	y	!	s	ia	"					
e	w	wt	x	d	h	p	i	="	60												
e	t	e	nl	n	d	l	m	E	i	=	S	ai	"	a							
e	x	e	d	ob	ol	d	rp	l	f	Ha	a	a	s	'	s	'	a				
e	t	e	u	o	le	B	dc	l	p	f	a	s	a	=	y	s	'	s	'	a	
w	t	at	add	ehol	ld	l	s	f	i	i	Ha	s	s	=	s	'	s	'	a		
t	tre	intel	f	l	bl	V	c	i	s	i	i	y	s	s	'	s	'	s	'	a	
t	te	o	tbol	d	bl	V	i	s	i	i	=	y	i	"	a						
t	tb	e	b	l	ca	dd	hV	i	s	i	i	=	y	i	"						
t	te	e	b	l	ca	dd	hv	i	s	i	i	=	y	i	"						
e	e	B	l	mdo				i	a	=	y										
e	e	t	bu	j	c			s	=	"											
t	e	A	addr																		
nt	e	o	edu	tml	eu	l	r	l	d	h	Fi	a	=	a	(i	'	a	";)		
e	e	@	oddrm	@	fn	r	o	mp	m@	no	rss	pseh	1	so	p	m	y	o	mapc	,	s
e	ex																				

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
underline	do C p Boolean a	メール本文が空の場合に本文を折りたたんで上下のサイズを小さくするかどうかを指定する boolean 値。		25.0	
bcc	b l cc	String i \$ [BCC] 項目の表示に関する設定。「editable」、「editableWithLookup」、「readOnly」、または「hidden」のいずれかです。		25.0	
bcc	b l cc	String i \$ i [CC] 項目の表示に関する設定。「editable」、「editableWithLookup」、「readOnly」、または「hidden」のいずれかです。		25.0	
body		String i a メール本文のデフォルトのテキスト値。		25.0	
textBody	String i a メール本文の形式。「text」、「HTML」、または「textAndHTML」のいずれかです。			25.0	
height	g h String i H a	メール本文の高さ(em 単位)。		25.0	
compact	c Boolean Q	クイックテキストのオートコンプリート機能をメールパブリッシャーで使用可能にするかどうかを指定します。		25.0	
id	I id	メールパブリッシャーを表示するレコードのエンティティ ID。現在のバージョンでは、ケースレコード ID のみがサポートされています。	はい	25.0	
draggable	d r p Boolean	ヘッダーが展開可能か固定であるかを指定する boolean 値。		25.0	
fromEmail	f String	送信元アドレスの制限されたセット。		25.0	
selectable	f String i \$ [送信者] 項目の表示に関する設定。「selectable」または「hidden」のいずれかです。			25.0	
refId	String i	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
onFailure	onFailure String	メールの送信に失敗した場合に呼び出される JavaScript。		25.0	
onSuccess	onSuccess String	メールが正しく送信された場合に呼び出される JavaScript。		25.0	
rendered	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global
rendered	Object R	メールが正しく送信されたときに再作成される 1 つ以上のコンポーネントの ID。この値には、単一の ID、ID のカンマ区切りのリスト、または ID の		25.0	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
リストまたはコレクションの差し込み項目の式を使用できます。					
BindedName	String	メールパブリッシャーの送信ボタンの名前。		25.0	
AdditionalField	Boolean	パブリッシャーのレイアウトに定義された追加項目を表示するかどうかを示す boolean 値。		25.0	
Attachment	Boolean	添付ファイルセレクタを表示するかどうかを指定する boolean 値。		25.0	
SendButton	Boolean	送信ボタンを表示するかどうかを指定する boolean 値。		25.0	
Template	Boolean	テンプレートセレクタを表示するかどうかを指定する boolean 値。		25.0	
buttonCaption	String	[件名] のデフォルト値。		25.0	
buttonCaptionSetting	String	[件名] 項目の表示に関する設定。「editable」、「readOnly」、または「hidden」のいずれかです。		25.0	
buttonFunction	String	メールを送信するために JavaScript からコールできる関数の名前。		25.0	
headerLabel	String	メールパブリッシャーヘッダーに表示されるタイトル。		25.0	
defaultRecipient	String	[宛先] 項目のデフォルト値。		25.0	
recipientSetting	String	[宛先] 項目の表示に関する設定。「editable」、「editableWithLookup」、「readOnly」、または「hidden」のいずれかです。		25.0	
width	String	メールパブリッシャーの幅(ピクセル(px) またはパーセント(%) 単位)。		25.0	

apex:enhancedList

現在選択されているビューのレコードの関連付けられたリストを含む、オブジェクトのリストビュー選択リストです。標準の Salesforce アプリケーションでは、このコンポーネントは特定のオブジェクトのメインタブに表示されます。[このコンポーネント](#)には、[p](#) と比較した、[ページあたりの高さおよび行数など、指定可能な追加属性があります。](#)

注: `<apex:enhancedList>` を他のコンポーネントの `<apex:repeat>` に設定されている場合は「block」属性を持つ `<apex:repeat>` コンポーネント内にあります。コンポーネントは `block` 属性が `false` に設定されないページでは使用できません。1 ページには、5 つの `<apex:repeat>` コンポーネントのみ含めることができます。このコンポーネントを使用するページには、バージョン 3 より前の Ext JS を含めることはできません。

w l

p

も参照してください。 >

例

```

g p p < a à
x e n n t et d p hte àn t< ou pæcw Lé ñhi sh o y=" g ò' èt è=" òu" 00cs P1 Pa " i 0=" L i
x e n n t et ed p h c t d e ñw e ñh ñs r Ly" g " ai è=3 " 00 s P Pa " 5
t d t d e z a i o bñc il =" L i à" s i è' F s 'a > /
g p p < /a à

```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
allowEdit	Boolean	s i 現在のユーザがリストをカスタマイズできるかどうかを指定する boolean 値。指定されていない場合、デフォルト値は true です。この属性が false に設定されている場合、現在のユーザはリスト定義を編集したり、リスト名、検索条件、表示される列、列の順序、または表示を変更したりすることはできません。ただし、この場合でも列幅または並び替え順序などの現在のユーザの個人設定を設定することはできます。		14.0	
height	Integer	i リストの高さをピクセル単位で指定する integer 値。 はい この値は必須です。	はい	14.0	
listId	String	i 目的のリストビューのデータベース ID。リストビューの定義を編集するときは、この ID はブラウザのアドレスバーの「fcf=」に続く 15 文字の文字列になります。 type が指定されていない場合、この値は必須です。		14.0	global
recordId	I	String i s ビューを表示する Salesforce オブジェクト。 type が指定されていない場合、この値は必須です。		14.0	
refreshOnSave	String	c p ブラウザでページが更新された後に実行される JavaScript。ページの更新によりこの JavaScript は自動的にコールされますが、インライン編集および後続の保存では自動的にコールされません。		14.0	
rerender	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	
rerender	Object	R AJAX 更新要求の結果がクライアントに返されるときに再作成される 1 つ以上のコンポーネントの ID。この値には、単一の ID、ID のカンマ区切りのリスト、または ID のリストまたはコレクションの差し込み項目の式を使用できます。注:他のコンポーネントの rerender 属性を使用して enhancedList を再表示する場合、enhanceList は「block」に設定されて		14.0	

属性名	属性型	説明	必須項目	API バージョン	アクセス
rowCount	Integer	ページあたりの行数を指定する integer 値。デフォルト値は現在のユーザの設定です。使用できる値は、10、25、50、100、200 です。注: この値に 100 を超える値が設定されている場合は、パフォーマンスが低下する可能性があるという警告メッセージが自動的に表示されます。		14.0	
type	String	type="Account" または type="My_Custom_Object__c" など、ビューを表示する Salesforce オブジェクト。		14.0	
width	Integer	リストの幅をピクセル単位で指定する integer 値。デフォルト値は利用可能なページ幅です。ビューポートの最初に表示可能な領域にリストが表示されていない場合は、デフォルト値はブラウザの幅になります。		14.0	

apex:facet

blip のヘッダーまたはフッターなどの、親コンポーネントの特定の部分で表示されるコンテンツのプレースホルダです。

コンポーネントは、親が facet をサポートしている場合にのみ、親コンポーネントの本文に含めることができます。facet コンポーネントの名前は、親コンポーネントの事前定義された facet 名の 1 つと一致する必要があります。この名前により facet コンポーネントのコンテンツが表示される場所が特定されます。facet コンポーネントが親コンポーネントの本文内で定義される順序は、親コンポーネントの表示には影響しません。

facet の例については、blip を参照してください。

注意: Apex で直接 blip を表すことはできませんが、facet を持つ動的コンポーネントで指定できます。次に例を示します。

```

<apex:form>
    <apex:pageBlock title="Customer Information">
        <apex:pageBlockSection title="Customer Details">
            <apex:inputText value="John Doe" />
            <apex:inputText value="123 Main Street" />
            <apex:inputText value="Anytown, USA" />
        </apex:pageBlockSection>
        <apex:pageBlockSection title="Order Information">
            <apex:inputText value="Product A" />
            <apex:inputText value="Product B" />
            <apex:inputText value="Product C" />
        </apex:pageBlockSection>
        <apex:facet name="Footer">
            <p>Copyright © 2023 Your Company. All rights reserved.</p>
        </apex:facet>
    </apex:pageBlock>
</apex:form>

```

```

e   { t tne o } c mc ! h a
e   x n lo u mpx < /a >
e   x n lo u mpx < a >
e   ex te ene e p fne dnr ex ah a =ap f "a > < p / a a>
e   { t tne on } co ch ! a p
e   x n lo u mpx < /a >
e   xte d bl p k /a a a a>
ex B l go p p c < k/a a >
g p p < /a a >

```

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
name	String	表示する facet の名前。この名前は親コンポーネントの事前定義されている facet 名の 1 つと一致する必要があります。facet コンポーネントのコンテンツが表示される場所を特定します。たとえば、dataTable コンポーネントには「header」、「footer」および「caption」という名前の facet があります。	はい	10.0	global

apex:flash

HTML オブジェクトおよび埋め込みタグを使用して表示される Flash ムービーです。

```

e gl b r ep l & o f d a h h s = af s "a s a" s "a >
t fl rwwap fh ecehd ba np ad em ac w l n/dnr ae.t pug / apd s a / as a1 a f _ / _ i
wt g h hd h i = 3 " 0 1 = " 0 0 > /
g p p < /a a >

```

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
flashvars	String	flashvars 属性を使用すると、ムービーにルートレベルの変数をインポートできる。すべての変数は SWF の最初のフレームが再生される前に作成されます。値はアンパサンド区切りの名前-値のペアのリストで構成されます。		14.0	
height	String	このムービーが表示される高さ。利用可能な縦方向の合計スペースに対する相対パーセント(50%など)、またはピクセル数(100など)のいずれかとして表されます。	はい	14.0	
id	String	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
p	Boolean	Flash ムービーを繰り返し再生するか、1 回のみ再生するかどうかを指定する boolean 値。true に設定されている場合、Flash ムービーが繰り返し再生されます。指定されていない場合、この値はデフォルトの false に設定されます。		14.0	
p	Boolean	Flash ムービーが表示された場合に自動的に再生を開始するかどうかを指定する boolean 値。true に設定されている場合、Flash ムービーが自動的に再生を開始します。指定されていない場合、この値はデフォルトの false に設定されます。		14.0	
	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global
c	String	表示されるムービーへのパス。URL として表されます。Salesforce の静的リソースとして Flash ムービーを保存できます。	はい	14.0	
h	String	このムービーが表示される幅。利用可能な横方向の合計スペースに対する相対パーセント (50% など)、またはピクセル数 (100 など) のいずれかとして表されます。	はい	14.0	

apex: form

ユーザが入力を行ってから、または、
Visualforce ページのセクションです。フォームの本文で、表示されるデータおよびその処理方法を特定します。
1つのページまたはカスタムコンポーネントで1つの
タグのみを使用するためのベストプラクティスです。

API バージョン^{v18.0} では、このタグは `<div>` の子コンポーネントにすることはできません。

このコンポーネントでは、「`html-`」 префиксを使用した **HTML パスルー属性**がサポートされています。パスルー属性は、生成された `<div>` タグに適用されます。

例

```

ex   x    ttB  lBgo pu p c < k a a s >
      ttB  eBndmme w o lt u < a o } a v a'v Sa ia ='' k 'a > /
ex   x    ttB  lBgo pu p c < k/a a s >
ex   e    e B el{go pbl p lc u < T k a rca va c a' ! s sa'v ='' a' >
e   x    e n lo ue hpu n & b c ma v a' ! . s a " > /
e   x    e n lo u hpte n n < ou c ac m v a' ! . a . 'a > /
e   x    e n lo u hpte n n < ou c ac m v a' ! . a . 'a > /
e   x    e n lo ue hpte }bu< jc a c v a' ! s " > /
e   x    e en do u mtr l hu < u a V a a' S sa' >
e   x    e t n e { u ldpt k p & u ac E v a' ! . S sa" > /
e   x    n lo u mp < /a >
ex   e B l go pbl p c < T k/a a a>
ex   ex   B l go p p c < k/a a a >
x    ro m p f < /a >
      g p p < /a >

```

上述の例では次の HTML を表示します。

```

w  t l l eo etau & t g dh e thaoas auy f a s sa y s sa >-
  roed m tndj e g:ma ro a' ='' mndj_ 0e atg Shsaro "t m ab i o h 0 a ss " " ='' s "
e  n  x o en xc et db o p t i showW pas ap a'mo rlim y'dd ia ic a / - - - " >
e  n n to d g p g < ! -- i vi sa >-
e  bl b rd r bl dd <c g a' ='' g0 cp ia ='' 0 s i a ='' >0
r  < >
t  edit l tb cl dp < T ss=a i "< >" >
  ded tndj g:du ro a' ='' dij_ttlQ BnBs sa o li p9i _" ss=a " >
  t at t bu p m p < i y's i "
e  n e mndj g:du ro a' ='' dij_@j a2 ss a 1i _9i _ i 3_ 0
  e  e l u t ln b c v a'v Sa ss=a " > /
t  d
r  < >
bl  < / a>
d  l B b do p < vi ss=a " y
  blt le l ec b rd r < al a id ss=a g:pl a' ='' g0 cp ia ='' 0 s i a ='' >0
  lorn og c p p < s ='' a" > /
et  t  e e r l w dcr oh < ss=a R " >
  t  e e l w ke dcr oh e & e lo ctss=pc br a " s # " > s N < a >
  t  e e l w eh dcr oh o< teAn c tsse a c m " s ='' h " > N< a >
  t  e e l w ke dcr oh e & t lo c ss=pc m R " h s ='' " N< a >
  t  e e l w ke dcr oh e & t to c bs=pc j R c" sh ='' " > S< a >
  t  e e l w ke dcr oh t ts tlo c ss=pc m R " s ='' " > S sa >
et  r
et  d h < / >a
t  b do < >y
t  tr ele w d a to r < ss=a Ra a i s " >
  t  etd l d c l h C < tn pss=a da a' > ps la 0000 000< / >a >
  t  etd l d e l h C dtm go mla tho a a' > s d > a p i i < as s </ >a >
  t  etd l d e l h e Gz o z o pss=a a a' > s ps G < a s </ >a >
  t  etd l d c tln t ee nc < tn e tps to a a' > $ r> af Si a c c a a i
lmu  f d p i < a s </ >a >
  t  etd l d c ll C < ss=a a a' >
  e  t  l c < s >
  t  en o @ l un pt o n < o oi v p a' " > N--< -- i/ >
  t  en e o e w l w u tp n o < o i v p a' N " N i/ >
  t  en o o d eme pto e e egt< d i l w c dnta ro d n " g o ='' s w p " > i<
  t  en o e tl u pelt d Et d l dn E o o a' s p'a > s < a a i/ >
  t  en oe o leu lp odt l Qdn oic o a' p s " > s i/ >
  e  t  l c < s / >
  t  d < >
  t  r < >
t  tr l w d tc o dd l < ss=a Ra a s 'a >
  t  etd l d c l h C < tn pss=a 2d a' > ps la 000070 s </ >a >
  t  etd l d ce l h C < B opss=a g a' > s k > ps y i < s s </ >a >
  t  etd l d c l h n A C nd ta opss=g al a' > sp > a y < s </ >a >
  t  etd l d ec l h e nc < pgs tn ja a' > ps d > a ip a < a < a </ >a >

```

t	e	e	t	etd	l	d	c	ll	C<	ss=a	a a" >
					l		c		< s	>	
			t	e	n	o	⊕	l un pt	o n <o	oi v	p a" " > N--<
			t	e	n e	o	eω	l wu tp	n o<	o i v	p a" N " x i/ >
			t	e	n	o	o	dem e p o e e	egt< d i l w c	dnt a" ro d n " g o	=" s x p " > ik
			t	e	n	o	⊕	tl u pelt d Ets l	dn w eo	oa" s pa" a >s <a a i/ >	
			t	e	n	œ	o	leu lp odt l	o dn oic v	a" p s " > s i/ >	
			t	e	e	t	l	c	< s / >		
			t			d			< / >		
			t		r				< / >		
			t	b	do				< / a>		
			d	bl					< / a>		
			d	n	tl	o	d	g	< ! -- \$. >	s a >-	
			ro	m	f				< / >		

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス	
cc	p	String	a	このフォームを処理するサーバが扱えるコンテンツタイプのカンマ区切りのリスト。この属性の使用できる値には、「text/html」、「image/png」、「image/gif」、「video/mpeg」、「text/css」、および「audio/basic」があります。使用できる値の完全なリストなど、詳細は、 W3C仕様 を参照してください。	10.0	global
cc_dhp	p	String	a	このフォームを処理するサーバが扱える文字符号化のカンマ区切りのリスト。指定されていない場合、この値はデフォルトの「UNKNOWN」に設定されます。	10.0	global
		String	i	生成された HTML コンポーネントの読み取り方向。使用可能な値には「RTL」(右から左) または「LTR」(左から右) があります。	10.0	global
c	p	String	y	サーバへのフォームの送信に使用されるコンテンツタイプ。指定されていない場合、この値はデフォルトの「application/x-www-form-urlencoded」に設定されます。	10.0	global
fc		Boolean	L	ページ自体が SSL を使用して提供されたかどうかには関係なく、SSL を使用してフォームが送信される。デフォルトは false です。値が false である場合、フォームは同じプロトコルを使用してページとして送信されます。forceSSL が true に設定されている場合、フォームの送信時には返されるページは SSL を使用します。	14.0	
		String	i	ページの他のコンポーネントがフォームコンポーネントを参照できるようにする識別子。	10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
language	String	「en」または「en-US」など、生成された HTML 出力の基本言語。この属性についての詳細は、 W3C仕様を参照してください。		10.0	global
onclick	String	onclick イベントが発生した場合(ユーザがフォームをクリックした場合)に呼び出される JavaScript。		10.0	global
ondblclick	String	ondblclick イベントが発生した場合(ユーザがフォームをダブルクリックした場合)に呼び出される JavaScript。		10.0	global
onkeydown	String	onkeydown イベントが発生した場合(ユーザがキーボードのキーを押した場合)に呼び出される JavaScript。		10.0	global
onkeypress	String	onkeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出される JavaScript。		10.0	global
onkeyup	String	onkeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出される JavaScript。		10.0	global
onmousedown	String	onmousedown イベントが発生した場合(ユーザがマウスボタンをクリックした場合)に呼び出される JavaScript。		10.0	global
onmousemove	String	onmousemove イベントが発生した場合(ユーザがマウスポインタを移動した場合)に呼び出される JavaScript。		10.0	global
onmouseout	String	onmouseout イベントが発生した場合(ユーザがフォームからマウスポインタを移動した場合)に呼び出される JavaScript。		10.0	global
onmouseover	String	onmouseover イベントが発生した場合(ユーザがマウスポインタをフォームに重ねた場合)に呼び出される JavaScript。		10.0	global
onmouseup	String	onmouseup イベントが発生した場合(ユーザがマウスボタンを放した場合)に呼び出される JavaScript。		10.0	global
onreset	String	onreset イベントが発生した場合(ユーザがフォームの[リセット]ボタンをクリックした場合)に呼び出される JavaScript。		10.0	global
onsubmit	String	onsubmit イベントが発生した場合(ユーザがフォームの[送信]ボタンをクリックした場合)に呼び出される JavaScript。		10.0	global
clientId	Boolean	clientid 生成プロセス時にこのフォームがその子コンポーネントの ID の前にフォームの ID を追加するかどうかを指定する boolean 値。指定されていない		10.0	global

属性名	属性型	説明	必須項目	API バージョン	アクセス
rd	Boolean	い場合、この値はデフォルトの true に設定されます。		10.0	global
style	String	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	s	10.0	global
class	String	フォームコンポーネントの表示に使用されるスタイル。主に、インライン CSS スタイルを追加するために使用されます。	s	10.0	global
target	String	フォームコンポーネントの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。	a	10.0	global
label	String	フォームの送信後にレスポンスを表示するフレームの名前。この属性に使用できる値には、「_blank」、「_parent」、「_self」、「_top」があります。また、目的の移行先の name 属性に値を割り当てることにより、独自のターゲット名を指定することもできます。	i	10.0	global
title	String	ユーザがコンポーネントにマウスポインタを重ねたときにツールチップとして表示されるテキスト。		10.0	global

apex:gaugeSeries

特定の基準に沿って、測定値の変化を示すデータ系列です。少なくとも、表示するゲージレベルのラベルおよび値のペアとして使用するデータコレクションの項目を指定する必要があります。ゲージグラフは、関連付けられた `p` (type が「gauge」である必要があります) に沿って適切な最小値と最大値を指定することで、読みやすくなります。

注: このコンポーネントはコンポーネントで囲む必要があります。グラフには `rg` を 1 つだけ使用します。

例

```
<apex:gaugeSeries value="50" min="0" max="100">
    <apex:range>0-25</apex:range>
    <apex:range>25-50</apex:range>
    <apex:range>50-75</apex:range>
    <apex:range>75-100</apex:range>
</apex:gaugeSeries>
```

属性

	属性名	属性型	説明	必須項目	APIバージョン	アクセス
lоро	c	String	ゲージレベルの塗りつぶしの色として使用される一連の色の値。色は、HTMLスタイル(16進)の色をカンマ区切りで指定します。たとえば、 です。 # F 0 Q F 0 0		26.0	
d	ld	String	ゲージレベルのデータ値の取得元である、グラフデータに指定された各レコードの項目。最初のレコードのみが使用されます。	はい	26.0	
d o u		Integer	ゲージグラフの中心に配置する穴の半径を、ゲージの半径のパーセントとして表す整数。デフォルトの0を使用すると、穴のない、つまり半円のゲージグラフが作成されます。		26.0	
l g	hg h h	Boolean i i	マウスポインタを重ねたときに各ゲージレベルを強調表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		26.0	
d		String i	ページの他のコンポーネントがグラフコンポーネントを参照できるようにする識別子。		26.0	global
l b l	ld	String	ゲージレベルのラベルの取得元である、グラフデータに指定された各レコードの項目。最初のレコードのみが使用されます。指定されていない場合、この値はデフォルトの「name」に設定されます。		23.0	
dl		Boolean	ゲージの針を表示するかどうかを指定する boolean 値。デフォルトは false で、針は表示されません。		26.0	
r d r d		Boolean	グラフでグラフ系列を表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		26.0	
r nd r r		String	ゲージ要素が表示される方法を追加または上書きする JavaScript 関数の名前を指定する文字列。追加のスタイルを指定またはデータを追加するために実装します。		26.0	
p	s	Boolean i s	マウスポインタを重ねたときにゲージレベルのツールチップを表示するかどうかを指定する boolean 値。ツールチップの形式は <labelField>: <dataField> です。指定されていない場合、この値はデフォルトの true に設定されます。		26.0	

apex:iframe

Visualforce ページ内のインラインフレームを作成するコンポーネントです。このフレームを使用すると、他の情報をスクロールまたは他の情報に置き換えたときに一部の情報を表示したままにすることができます。

このコンポーネントでは、「html-」プレフィックスを使用した [HTML パススルー属性](#)がサポートされています。パススルー属性は、生成された `<div>` タグに適用されます。

例

```
tt    rnew f e c n l : pto oen blc et age s dsa r " shm li f =" " i =" " a > /
```

上述の例では次の HTML を表示します。

```
tr    m efg he h de < aip e hanI f# tt'@i hewf e " a h #a: pro " as sc c // s sa
d e    h r m fi l = <" @i / a
```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
border	Boolean	境界線がインラインフレームを囲むかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの <code>false</code> に設定されます。		10.0	global
height	String	インラインフレームの高さ。利用可能な縦方向の合計スペースのパーセント (<code>height="50%"</code> など)、またはピクセル数 (<code>height="300px"</code> など) のいずれかとして表されます。指定されていない場合、この値はデフォルトの <code>600px</code> に設定されます。		10.0	global
refId	String	ページの他のコンポーネントがインラインフレームコンポーネントを参照できるようにする識別子。		10.0	global
rendered	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの <code>true</code> に設定されます。		10.0	global
scrollable	Boolean	インラインフレームをスクロールできるかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの <code>true</code> に設定されます。		10.0	global
url	String	インラインフレームの最初のコンテンツを指定する URL。この URL は、外部 Web サイトまたはアプリケーション内の他のページのいずれかです。		10.0	global
title	String	ユーザがコンポーネントにマウスポインタを重ねたときにツールチップとして表示されるテキスト。		10.0	global
width	String	インラインフレームの幅。利用可能な横方向の合計スペースのパーセント (<code>width="80%"</code> など)、また		10.0	global

属性名	属性型	説明	必須項目	API バージョン	アクセス
はピクセル数 (width="600px" など) のいずれかとして表されます。					

apex:image

HTML `img` タグで表示される画像です。

このコンポーネントでは、「html-」プレフィックスを使用した **HTML パススルー属性**がサポートされています。パススルー属性は、生成された `img` タグに適用されます。 >

例

```
et e mge p m gke Iu á mgwtm =m ge g d "væ fæ" ih / 2dg/yh á " i =" " @ =" " 55 /
```

上述の例では次の HTML を表示します。

```
e dmgi mgi I < mgwtmi m=ge g d "as f=" ih / 2dg/yh á " i =" " @ =" " 55 /
```

リソースの例

```
et e mge p e m gke Iue á e oiae=" m owtm a mgcdRts .Rys2g h h "a 2 =" " 0@
```

上述の例では次の HTML を表示します。

```
e dmgi e e emtñ I e <rgied de=" m ñ" as <m gd It a > h R/yg h h "a 2 =" " 0@
```

Zip リソースの例

```
erig pe e t < oue á e æ UñLB! Rñg Sp Rñst) g id t ip ' sa h / ig hs .h " i =" " 5@
```

上述の例では次の HTML を表示します。

```
d ee emtñ I e rg cd dl =" B mñ]" austl=" e [j gl ta i p/ sa gih sh. " i =" " 5@
```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
src	String	セクション 508 に準拠するために使用される画像の代替テキストの説明。	1	10.0	global
dir	String	生成された HTML コンポーネントの読み取り方向。使用可能な値には「RTL」(右から左) または「LTR」(左から右) があります。	1	10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
g h h	String i	この画像が表示される高さ。利用可能な縦方向の合計スペースに対する相対パーセント (height="50%" など)、またはピクセル数 (height="100px" など) のいずれかとして表されます。指定されていない場合、この値はデフォルトのソース画像ファイルのサイズに設定されます。		10.0	global
d m	String i	ページの他のコンポーネントが画像コンポーネントを参照できるようにする識別子。		10.0	global
m p	Boolean i s	この画像を画像マップとして使用するかどうかを指定する boolean 値。true に設定されている場合、画像コンポーネントは commandLink コンポーネントの子である必要があります。指定されていない場合、この値はデフォルトの false に設定されます。		10.0	global
l g	String a	「en」または「en-US」など、生成された HTML 出力の基本言語。この属性についての詳細は、 W3C仕様 を参照してください。		10.0	global
l o d g	c String	s 画像の詳細説明へのリンクの URL。		10.0	global
o l	c c String i	onclick イベントが発生した場合(ユーザが画像をクリックした場合)に呼び出される JavaScript。		10.0	global
dbl l	c c String i	ondblclick イベントが発生した場合(ユーザが画像をダブルクリックした場合)に呼び出される JavaScript。		10.0	global
no d o	k String	keydown イベントが発生した場合(ユーザがキーボードのキーを押した場合)に呼び出される JavaScript。		10.0	global
o r	p k String	onkeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出される JavaScript。		10.0	global
o u	p k String	onkeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出される JavaScript。		10.0	global
w on o und o	String s	onmousedown イベントが発生した場合(ユーザがマウスポンタをクリックした場合)に呼び出される JavaScript。		10.0	global
o o um o m	String s	onmousemove イベントが発生した場合(ユーザがマウスポインタを移動した場合)に呼び出される JavaScript。		10.0	global
o o um ou	String s	onmouseout イベントが発生した場合(ユーザが画像からマウスポインタを移動した場合)に呼び出される JavaScript。		10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
onmouseover	String	イベントが発生した場合(ユーザが画像にマウスポインタを重ねた場合)に呼び出されるJavaScript。	sv	10.0	global
onmouseup	String	イベントが発生した場合(ユーザがマウスボタンを放した場合)に呼び出されるJavaScript。	sp	10.0	global
visible	Boolean	コンポーネントをページに表示するかどうかを指定するboolean値。指定されていない場合、この値はデフォルトの true に設定されます。		10.0	global
style	String	画像コンポーネントの表示に使用されるスタイル。主に、インライン CSS スタイルを追加するために使用されます。	s	10.0	global
styleClass	String	画像コンポーネントの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。	sc	10.0	global
title	String	ユーザがコンポーネントにマウスポインタを重ねたときにツールチップとして表示されるテキスト。	i	10.0	global
url	String	表示されている画像へのパス。URL または静的リソースかドキュメントの差し込み項目のいずれかとして表されます。		10.0	global
imageMapName	String	この要素が画像を提供するクライアント側の画像マップ(HTML マップ要素)の名前。	sm	10.0	global
urlObject	Object	表示されている画像へのパス。URL または静的リソースかドキュメントの差し込み項目のいずれかとして表されます。	ao	10.0	global
width	String	この画像が表示される幅。利用可能な横方向の合計スペースに対する相対パーセント(width="50%"など)、またはピクセル数(width="100px"など)のいずれかとして表されます。指定されていない場合、この値はデフォルトのソース画像ファイルのサイズに設定されます。	si	10.0	global

apex:include

現在のページに2番目のVisualforceページを挿入するコンポーネントです。ページのサブツリー全体が、参照ポイントおよび含まれるページが保持される範囲で Visualforce DOM に挿入されます。

含まれるページからコンテンツを除外する必要がある場合は、代わりに `om` `sp` `ao` `<コンポーネント $ i >` を使用します。

例

```

g : < ! -- Pa >
et e dg p p gh< a ia ="'Pa>
et txe ou ue pl up et<T eg a p h v T gh ba" p a i si s "> / > /
ne e l du e p nc g m l pd u ác Na =á " > /
g p p < /a >
e n g l du : c < ! -- Pa >
et e rge p pl dudh<I c g a ia ="'Pa>
et txe ou u ep m up &Tdu txa c et hmer @ a" mo(r f n hysi s p a "a > /
g p p < /a >

```

上述の例では次の HTML を表示します。

```

et e eg p h Tgh p a i si s < a > /
c e de n p eghndu & esdu txa c et hmer @ " mo(r f n hysi s p p a < a s / >a

```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
i	String	ページの他のコンポーネントが挿入されたページを参照できるようにする識別子。		10.0	global
m	ApexPagesPageReference	コンテンツを現在のページに挿入する Visualforce ページ。この値では、Visualforce ページの名前を指定するか、または差し込み項目の構文を使用して、ページまたは PageReference を参照します。	はい	10.0	global
r	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		10.0	global

apex:includeScript

Visualforce ページで使用できる JavaScript ライブラリへのリンクです。指定されている場合、このコンポーネントは、生成された HTML ページの head 要素にスクリプト参照を挿入します。

パフォーマンス上の理由から、`g p p` の終了タグの前には、このコンポーネントではなく JavaScript タグを使用することをお勧めします。

このコンポーネントでは、「html-」プレフィックスを使用した HTML パススルー属性がサポートされています。パススルー属性は、生成された `c` タグに適用されます。 >

例

```

n t el du rep c elem < exp ca > j i $ j a"p ! R s . a s " > /

```

上述の例では次の HTML を表示します。

c t p ej pr ↗ s ic r grace ' vex / sa ia 2 s '‡ m s/ jl 1 β‡ 6 46000/ a s v '

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
	String	i ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		13.0	global
	Boolean	o スクriptトリソースをすぐに読み込むのか、ドキュメントモデルの作成後に読み込むのか を指定します。デフォルト値「false」では、スクリプトがすぐに読み込まれます。「true」に設定すると、コンポーネントが参照する JavaScript は、ページが「ready(準備完了)」になるまで待機してから読み込まれます。 n この方法で読み込まれたスクリプトは、すぐにではなく、イベントがトリガされた後に DOM に追加されます。これは、DOM が作成された後に行われますが、子フレームまたは外部リソース(画像など)の読み込みが終了する前に行われる場合があります。		29.0	
	Object	a JavaScript ファイルへの URL 。これは静的リソース はい への参照にすることができます。		13.0	global

apex:inlineEditSupport

このコンポーネントにより `u` `u_l` `p` `m_p_f` およびさまざまなコンテナコンポーネントのオンライン編集がサポートされます。オンライン編集をサポートするには、このコンポーネントも `m_p_f` タグ内に含める必要があります。

コンポーネントは次のタグの子孫としてのみ使用できます。

< aL ia sa >

$\mathbf{E} \leq \mathbf{S} \leq \mathbf{S}_1 \mathbf{S}_2 \mathbf{S}_3$

IN α α

< a >

\leftarrow a \rightarrow

< k a a >

方正

G K α α

ed n l m p l d の < E a 属性も参照してください。

属性名	属性型	説明	必須項目	API バージョン	アクセス
dblclick	C String	項目のコントラクトが変更されたときに使用される CSS スタイルクラスの名前。		21.0	
dblclicks	Boolean	オンライン編集が有効化されているかどうかを示す boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		21.0	
doubleclick	String	項目でのオンライン編集をトリガする ondblclick、 onmouseover などの標準 DOM イベントの名前。		21.0	
disabled	E Object	ボタン ID のカンマ区切りのリスト。これらのボタンはオンライン編集が有効化されているときには非表示になります。		21.0	
draggable	String	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		10.0	global
displayed	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		21.0	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
onValueChange	String	値がリセットされた場合にコールされる JavaScript 関数の名前。		21.0	
buttonIDs	Object	ボタンIDのカンマ区切りのリスト。これらのボタンはインライン編集が有効化されているときに表示されます。		21.0	

apex:input

フォーム項目で期待されるデータに適応する、HTML5に適した一般的な入力コンポーネントです。HTML 属性を使用すると、クライアントブラウザが、日付ピッカーや範囲スライダなど、型に適したユーザ入力フィードバックを表示したり、クライアント側で数値範囲や電話番号などの書式設定または検証を実行したりできます。このコンポーネントは、Salesforce オブジェクトの項目に対応しないコントローラプロパティまたはメソッドのユーザ入力を取得するために使用します。

このコンポーネントでは、Salesforce のスタイル設定を使用しません。また、Salesforce 項目、またはオブジェクトのその他のどのデータにも対応していないため、ユーザが入力した値を使用するにはカスタムコードが必要です。

このコンポーネントでは、「html-」プレフィックスを使用した [HTML パススルー属性](#)がサポートされています。パススルーアクションは、生成された [p](#) タグに適用されます。 >

例

```
<apex:input type="text" value="Initial Value" />
```

上述の例では次の HTML を表示します。

```
<input type="text" value="Initial Value" />
```

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
cc	String	項目にフォーカスを置くキーボードのアクセスキー。テキストボックスにフォーカスが置かれている場合は、ユーザが項目値を選択または選択解除できます。		29.0	
label	String	項目の代替テキストの説明。		29.0	
direction	String	生成された HTML コンポーネントの読み取り方向。使用可能な値には「RTL」(右から左) または「LTR」(左から右) があります。		29.0	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
disabled	Boolean	このテキストボックスを無効な状態で表示するかどうかを指定する boolean 値。true に設定されている場合、テキストボックスは無効な状態で表示されます。指定されていない場合、この値はデフォルトの false に設定されます。	29.0		
label	String	ページの他のコンポーネントが項目コンポーネントを参照できるようにする識別子。	29.0	global	
label	String	コントロールの横に表示ラベルを表示し、エラーメッセージ内のコントロールを参照できるようにするテキスト値。	29.0		
language	String	「en」または「en-US」など、生成された HTML 出力の基本言語。この属性についての詳細は、 W3C仕様 を参照してください。	29.0		
list	Object	入力項目に関連付けられた HTML ブロックに追加するオートコンプリート値のリスト。	29.0	< 29.0 ia sa >	
listLabel	String	属性は、カンマ区切りの静的文字列または Visualforce 式として指定されます。式は、カンマ区切り文字列またはオブジェクトのリストに解決できます。リスト要素には任意のデータ型を設定できますが、その型は、Apex 言語機能として、またはメソッドを介して、文字列に変換できる必要があります。			
onblur	String	onblur イベントが発生した場合(フォーカスが項目から離れた場合)に呼び出される JavaScript。	29.0		
onChange	String	onchange イベントが発生した場合(ユーザが項目のコンテンツを変更した場合)に呼び出される JavaScript。	29.0		
onClick	String	onclick イベントが発生した場合(ユーザが項目をクリックした場合)に呼び出される JavaScript。	29.0		
onDoubleClick	String	ondblclick イベントが発生した場合(ユーザが項目をダブルクリックした場合)に呼び出される JavaScript。	29.0		
onFocus	String	onfocus イベントが発生した場合(フォーカスが項目にある場合)に呼び出される JavaScript。	29.0		
onKeyDown	String	onkeydown イベントが発生した場合(ユーザがキーボードのキーを押した場合)に呼び出される JavaScript。	29.0		
onKeyPress	String	onkeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出される JavaScript。	29.0		

属性名	属性型	説明	必須項目	APIバージョン	アクセス
onkeyup	String	onkeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出される JavaScript。		29.0	
onmousedown	String	onmousedown イベントが発生した場合(ユーザがマウスボタンをクリックした場合)に呼び出される JavaScript。		29.0	
onmousemove	String	onmousemove イベントが発生した場合(ユーザがマウスポインタを移動した場合)に呼び出される JavaScript。		29.0	
onmouseout	String	onmouseout イベントが発生した場合(ユーザが項目からマウスポインタを移動した場合)に呼び出される JavaScript。		29.0	
onmouseover	String	onmouseover イベントが発生した場合(ユーザがマウスポインタを項目に重ねた場合)に呼び出される JavaScript。		29.0	
onmouseup	String	onmouseup イベントが発生した場合(ユーザがマウスボタンを放した場合)に呼び出される JavaScript。		29.0	
rendered	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		29.0	
required	Boolean	この項目が必須項目であるかどうかを指定する boolean 値。true に設定されている場合、この項目の値を指定する必要があります。選択されていない場合、この値はデフォルトの false に設定されます。		29.0	
size	Integer	入力項目の幅。一度に表示可能な文字数で表されます。		29.0	
style	String	input コンポーネントの表示に使用されるスタイル。主に、インライン CSS スタイルを追加するために使用されます。		29.0	
styleClass	String	input コンポーネントの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。		29.0	
tabIndex	String	ユーザが Tab キーを繰り返し押したときに、他のページコンポーネントと比較して、この項目が選択される順序。この値は、ユーザが Tab キーを押したときに選択される最初のコンポーネントを 0 として、0 ~ 32767 の整数である必要があります。		29.0	
title	String	ユーザがコンポーネントにマウスポインタを重ねたときにツールチップとして表示されるテキスト。		29.0	

属性名	属性型	説明	必須項目	API バージョン	アクセス
p	String	<p><code>t n Y</code></p> <p>生成された属性。有効な値は、次のとおりです。</p> <ul style="list-style-type: none">autodatedatetimedatetime-localmonthweektimeemailnumberrangesearchteltexturl	<p><code>p</code>要素に追加される HTML5</p> <p>値は、次のとおりです。</p>	29.0	
Object	a	<p>この項目に関連付けられているコントローラクラス変数を参照する式。たとえば、コントローラスの関連付けられている変数の名前が <code>myTextField</code> である場合、この変数を参照するには <code>value="[{!myTextField}]"</code> を使用します。</p>		29.0	

apex:inputCheckbox

checkbox 型の HTML 入力要素です。このコンポーネントを使用して、Salesforce オブジェクトの項目に対応しないコントローラメソッドのユーザ入力を取得します。

このコンポーネントでは、「`html-`」プレフィックスを使用した **HTML パスルー属性**がサポートされています。パスルート属性は、生成された `<p>` タグに適用されます。>

例

```

ex      e B el{go pbl p lc aut< n kroa ju a ppr va o a! ! i i s" v =" a" >
x      e lo ue hpu } o < m a v a! ! . 'a > /
x      e lo u hptue on <o> sc m v a! ! . a . 'a > /
x      e en do u mert l hu r< a V a a! ! . 'a >?
e      x et n e u p o leat < o ikai pv a! ! i s' a > /
x      n lo u mpx < /a >
ex      e B l go pbl p c < T k/a a a>
ex      B l go p p c < k/a a a >
x      ro m p f < /a >
g p p < /a >

```

上述の例では次の HTML を表示します。

```

w t llo e tau o g dr ! rh es qyto taur oaf uj pi y i y >-
e ro ed m ndj er g:dn ro ienc=" mndj 0 e avtgPduFyo "t m ab i o_h 0 a ss=" " =s"
e n x o en xc et db o p t i shorw pas ppa 'mo rlim y'dd ia ic a / - - - " >
e n n to d g p g < ! -- i v sa >-
e bld b r d r bl dd <c g ml =c" g0 cp ia = " 0 s i a= " >0
r < >
t edit l tb cl dp < T ss=a i "< > >
t ded ndj rg:dn <ro idj" idj_ttl: a n2bP cDay o li p9i _" ss=a " >
t t t t bup m p < i y's i "
e n e mndj rg:dn ro idj" idj_0dja 2B Fay li _9i _ i 3" 0
e e l u t ln b c v a'v S a ss=a " > /
t r < >
bl < / a>
d l B b do p < v ss=a " y
d blt le l ec b rd r < bl ss=a igs "l =c" g0 cp ia = " 0 s i a= " >0
et lorn og c p p < s =a" > /
d h < >a
t e e r l w dcr oh < ss=a R " >
t e e l w ke dcr oh < to a ss=a tu R pp m = " " >0 i Ny< a >
t e e l w ke dcr oh < o<teAn c tss a c a m " s =h " > N< a >
t r < >
et d h < / >a
t b do < >y
t t re le w d a to r < ss=a Ra a i s " >
r tnro t etd l d c lm ten C etix e ss=a l n a a" n> s Tl ga i Wi s v i a P
t etd l d c lm ten C etix ss=a l a A s T i gni pf ci s p i< s
e mndj t etd l d to tell e a< x ss=a phac a k y= " " < > / >
t r g:dn ro idj" idj_e 0: dej eP :Fay d lith c 3_dh dk0 3" 5k = " " < > / >
e mndj t etd l d to tell e a< x ss=a phac a k y= " " < > / >
t r < >
t bl b do < > / >y
d n tl o d g g < ! -- $ v sa >-
ro m f < / >

```

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
cc	String	チェックボックスにフォーカスを置くキー。チェックボックスにフォーカスが置かれている場合は、ユーザがチェックボックスの値を選択または選択解除できます。		10.0	global
dir	String	生成された HTML コンポーネントの読み取り方向。使用可能な値には「RTL」(右から左) または「LTR」(左から右) があります。		10.0	global
disabled	Boolean	このチェックボックスを無効な状態で表示するかどうかを指定する boolean 値。true に設定されている場合、チェックボックスは無効な状態で表示されます。指定されていない場合、この値はデフォルトの false に設定されます。		10.0	global
label	String	ページの他のコンポーネントがチェックボックスコンポーネントを参照できるようにする識別子。		10.0	global
labelMode	Boolean	ページの項目に関連付けられている入力規則を処理することなく、このコンポーネントに関連付けられているアクションをすぐに実行するかどうかを指定する boolean 値。true に設定すると、アクションがすぐに実行され、入力規則はスキップされます。指定されていない場合、この値はデフォルトの false に設定されます。		11.0	global
labelText	String	コントロールの横に表示ラベルを表示し、エラーメッセージ内のコントロールを参照できるようにするテキスト値。		23.0	
language	String	「en」または「en-US」など、生成された HTML 出力の基本言語。この属性についての詳細は、 W3C仕様 を参照してください。		10.0	global
onblur	String	onblur イベントが発生した場合(フォーカスがチェックボックスから離れた場合)に呼び出される JavaScript。		10.0	global
onChange	String	onchange イベントが発生した場合(ユーザがチェックボックス項目のコンテンツを変更した場合)に呼び出される JavaScript。		10.0	global
onClick	String	onclick イベントが発生した場合(ユーザがチェックボックスをクリックした場合)に呼び出される JavaScript。		10.0	global
ondblclick	String	ondblclick イベントが発生した場合(ユーザがチェックボックスをダブルクリックした場合)に呼び出される JavaScript。		10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
onfocus	String	s onfocus イベントが発生した場合(フォーカスがチェックボックスにある場合)に呼び出されるJavaScript。		10.0	global
onkeydown	String	y onkeydown イベントが発生した場合(ユーザがキーボードのキーを押した場合)に呼び出されるJavaScript。		10.0	global
onkeypress	String	y onkeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出されるJavaScript。		10.0	global
onkeyup	String	y onkeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出されるJavaScript。		10.0	global
onmousedown	String	s onmousedown イベントが発生した場合(ユーザがマウスポンタをクリックした場合)に呼び出されるJavaScript。		10.0	global
onmousemove	String	s onmousemove イベントが発生した場合(ユーザがマウスポンタを移動した場合)に呼び出されるJavaScript。		10.0	global
onmouseout	String	s onmouseout イベントが発生した場合(ユーザがチェックボックスからマウスポンタを移動した場合)に呼び出されるJavaScript。		10.0	global
onmouseover	String	s onmouseover イベントが発生した場合(ユーザがマウスポンタをチェックボックスに重ねた場合)に呼び出されるJavaScript。		10.0	global
onmouseup	String	s onmouseup イベントが発生した場合(ユーザがマウスポンタを放した場合)に呼び出されるJavaScript。		10.0	global
onselect	String	s onselect イベントが発生した場合(ユーザがチェックボックスを選択した場合)に呼び出されるJavaScript。		10.0	global
required	Boolean	コンポーネントをページに表示するかどうかを指定するboolean値。指定されていない場合、この値はデフォルトの true に設定されます。		10.0	global
required	Boolean	i このチェックボックスが必須項目であるかどうかを指定する boolean 値。true に設定されている場合、このチェックボックスの値を指定する必要があります。選択されていない場合、この値はデフォルトの false に設定されます。		10.0	global
selected	Boolean	s このチェックボックスを「オン」の状態で表示するかどうかを指定する boolean 値。選択されていない場合、この値はデフォルトの false に設定されます。		10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス	
l	String	s y	inputCheckbox コンポーネントの表示に使用されるスタイル。主に、インライン CSS スタイルを追加するために使用されます。	10.0	global	
l l	C	String	s y	inputCheckbox コンポーネントの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。	10.0	global
b d	String	i a	ユーザが Tab キーを繰り返し押したときに、他のページコンポーネントと比較して、このチェックボックスが選択される順序。この値は、ユーザが Tab キーを押したときに選択される最初のコンポーネントを 0 として、0 ~ 32767 の整数である必要があります。	10.0	global	
l	String	i	ユーザがコンポーネントにマウスポインタを重ねたときにツールチップとして表示されるテキスト。	10.0	global	
l u	Object	a	このチェックボックスに関連付けられているコントローラクラス変数を参照する差し込み項目。たとえば、コントローラクラスの関連付けられている変数の名前が myCheckbox である場合、この変数を参照するには value="={!myCheckbox}" を使用します。	10.0	global	

apex:inputField

Salesforce オブジェクトの項目に対応する値の HTML 入力要素です。コンポーネントは、項目が必須であるかまたは一意であるかどうかなどの関連項目の属性、およびユーザからの入力を取得するため表示するず遂に ヴ翻垂のする。 ほ 0 ウ 畏ま 曾翻ほ - ト連ノ ぼ曳しです蜻垂たとえばを指填ナ 邀

API バージョン 20.0 から、デフォルト値を持つ項目に一致する `inputField` には Visualforce ページでデフォルト値があらかじめ入力されています。

注: 参照のみの項目、および `EntireTeam`、`EntireSubTeam`、`EntireObject` などの複雑な自動動作が含まれる特定の Salesforce オブジェクトの項目は `inputField` の使用時には編集可能として表示されません。代わりに、`inputText` など他の入力コンポーネントを使用してください。

このコンポーネントでは、「`html-`」プレフィックスを使用した **HTML パススルー属性**がサポートされています。パススルー属性は、生成された `p` タグに適用されます。 >

属性名	属性型	説明	必須項目	API バージョン	アクセス
<code>label</code>	<code>String</code>	ページの他のコンポーネントが <code>inputField</code> コンポーネントを参照できるようにする識別子。		10.0	global
<code>labelText</code>	<code>String</code>	項目に表示されるデフォルトのラベルを上書きできるテキスト値。ラベルはフォームにラベルを表示しない空の文字列に設定することができます。 <code>null</code> に設定すると、エラーになります。		23.0	
<code>list</code>	<code>Object</code>	入力項目に関連付けられた HTML ロックに追加するオートコンプリート値のリスト。		< 29.0 >	<code>isAccessible</code>
<code>transform</code>	<code>String</code>	属性は、カンマ区切りの静的文字列または Visualforce 式として指定されます。式は、カンマ区切り文字列またはオブジェクトのリストに解決できます。リスト要素には任意のデータ型を設定できますが、その型は、Apex 言語機能として、または <code>String</code> メソッドを介して、文字列に変換できる必要があります。			

属性名	属性型	説明	必須項目	APIバージョン	アクセス
onblur	String	onblur イベントが発生した場合(フォーカスが項目から離れた場合)に呼び出される JavaScript。		12.0	global
onchange	String	onchange イベントが発生した場合(ユーザが項目のコンテンツを変更した場合)に呼び出される JavaScript。		12.0	global
onclick	String	onclick イベントが発生した場合(ユーザが項目をクリックした場合)に呼び出される JavaScript。	i	12.0	global
ondblclick	String	ondblclick イベントが発生した場合(ユーザが項目をダブルクリックした場合)に呼び出される JavaScript。	i	12.0	global
onfocus	String	onfocus イベントが発生した場合(フォーカスが項目にある場合)に呼び出される JavaScript。	s	12.0	global
onkeydown	String	onkeydown イベントが発生した場合(ユーザがキーボードのキーを押した場合)に呼び出される JavaScript。	y	12.0	global
onkeypress	String	onkeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出される JavaScript。	y	12.0	global
onkeyup	String	onkeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出される JavaScript。	y	12.0	global
onmousedown	String	onmousedown イベントが発生した場合(ユーザがマウスボタンをクリックした場合)に呼び出される JavaScript。	s	12.0	global
onmousemove	String	onmousemove イベントが発生した場合(ユーザがマウスポインタを移動した場合)に呼び出される JavaScript。	s	12.0	global
onmouseout	String	onmouseout イベントが発生した場合(ユーザが項目からマウスポインタを移動した場合)に呼び出される JavaScript。	s	12.0	global
onmouseover	String	onmouseover イベントが発生した場合(ユーザがマウスポインタを項目に重ねた場合)に呼び出される JavaScript。	s	12.0	global
onmouseup	String	onmouseup イベントが発生した場合(ユーザがマウスボタンを放した場合)に呼び出される JavaScript。	s	12.0	global
onselect	String	onselect イベントが発生した場合(ユーザがこの項目に関連付けられているチェックボックスを選択した場合)に呼び出される JavaScript。	s	12.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
rendered	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		10.0	global
required	Boolean	この inputField が必須項目であるかどうかを指定する boolean 値。true に設定されている場合、この項目の値を指定する必要があります。選択されていない場合、この値はデフォルトの false に設定されます。この入力項目でカスタムオブジェクト名を表示する場合、この属性が true に設定されていない限り、その値は nil に設定され、必須にはなりません。これは標準オブジェクト名には適用されません。標準オブジェクト名はこの属性に関係なく常に必須です。		10.0	global
showDatePicker	Boolean	<p>この項目に Visualforce の日付ピッカーを使用するのか、抑制してブラウザベースの日付ピッckerを使用するのかを指定します。</p> <p>この属性は、日付項目と日時項目にのみ影響します。データ型に適したブラウザベースの選択ウィジェットを有効にするには、日付または時刻と互換性のある次のいずれかのデータ型に 属性を設定する必要があります。</p> <ul style="list-style-type: none"> date datetime datetime-local month week time 		29.0	
style	String	<p>inputField コンポーネントの表示に使用される CSS スタイル。この属性を設定できない値もあります。テキストにクラス名が必要な場合は、ラップ用 span タグを使用します。</p>	s	12.0	global
styleClass	String	<p>inputField コンポーネントの表示に使用される CSS スタイルクラス。この属性を設定できない値もあります。テキストにクラス名が必要な場合は、ラップ用 span タグを使用します。</p>	s	12.0	global
tabOrder	Integer	ユーザが Tab キーを繰り返し押したときに、他のページコンポーネントと比較して、この項目が選択される相対的な順序を示すヒント。この値は、ユーザが Tab キーを押したときに選択される最初のコンポーネントを 1 として、1 ~ 3276 の整数である必要があります。	a	23.0	

属性名	属性型	説明	必須項目	API バージョン	アクセス
pelt	String	<p>生成された属性。 有効な属性は、次のとおりです。</p> <ul style="list-style-type: none"> auto date datetime datetime-local month week time email number range search tel text url 		29.0	
l u	Object	<p>a この inputField に関連付けられている Salesforce 項目を参照する差し込み項目。たとえば、取引先の名前項目の入力項目を表示する必要がある場合は、value="={!account.name}" を使用します。組織が期間指定換算レートを使用している場合は、この inputField を currency 型の数式の差し込み項目に関連付けることはできません。</p>		10.0	global

apex:inputFile

ファイルをアップロードする入力項目を作成するコンポーネントです。

注意: Visualforce を介してアップロードできるファイルの最大サイズは 10 MB です。

例

```

e l d n t p d n u f < U! a e p r t a i n a o p a m i l d r y f s a s >-
t n t e g n d p r d p o r l e s e n t x o a n m a c a o a n d o n = m c " E $ s = " " >
x e t r o d m p f r o m < a s s a > / F >
ex et B l g o p p c < k a a >
ex e t B l g a p p o c < k a a > s >
e x t e { u l e p l t u p n d & u b i n a i o n f l v t e a d f n o u } m c n y i = a !
e x t t n e B n d m m e o l t u < a o > a v a' v S a i a = a > /
e ex e t B l g o p p c < c k/a a > s >
x ex ro m p f < / a >
g p p < / a >
n or l d r C *** / ***

```

```

    t{ tnd po m c E i ss a
e blu tnd o e ux p n t c nE te ndiprd toe m& r C dr ld rse . S a a
t n oe u m a t e me met dr l& r c gD&d ( c ) . R ;( )
e ld red net fr o t gif d ft. t i UI e ns h U ne psr tolM )oi/us m sic i y SPD a
}

```

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
cc	String	a コンテンツタイプのカンマ区切りのセット。ブラウザでこのリストを使用して、選択可能な一連のファイルオプションを制限できます。指定されていない場合、コンテンツタイプリストが送信されず、すべてのファイルの種類にアクセスできます。		14.0	
cc	String	ss コンポーネントにフォーカスを置くキーボードのアクセスキー。		14.0	
c	String	a コンポーネントの代替テキストの説明。		14.0	
c	String	アップロードされたファイルのコンテンツタイプを保存する string 型のプロパティ。		14.0	
dir	String	i 生成された HTML コンポーネントの読み取り方向。使用可能な値には「RTL」(右から左) または「LTR」(左から右) があります。		14.0	
dbl d	Boolean	s is このコンポーネントを無効な状態で表示するかどうかを指定する boolean 値。true に設定されている場合、コンポーネントは無効な状態で表示されます。指定されていない場合、この値はデフォルトの false に設定されます。		14.0	
mf	String	i N アップロードされたファイルの名前を保存する string 型のプロパティ。		14.0	
f	Integer	i i アップロードされたファイルのサイズを保存する integer 型のプロパティ。		14.0	
g	String	i ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
gl	String	a 「en」または「en-US」など、生成された HTML 出力の基本言語。詳細は、 http://www.w3.org/TR/REC-html40/struct/dirlang.html で、この属性に関する W3C の仕様を参照してください。		14.0	
ru	String	onblur イベントが発生した場合(フォーカスがコンポーネントから離れた場合)に呼び出される JavaScript。		14.0	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
onchange	String	onchange イベントが発生した場合(ユーザがコンポーネント項目のコンテンツを変更した場合)に呼び出される JavaScript。		14.0	
onclick	String	onclick イベントが発生した場合(ユーザがコンポーネントをクリックした場合)に呼び出される JavaScript。		14.0	
ondblclick	String	ondblclick イベントが発生した場合(ユーザがコンポーネントをダブルクリックした場合)に呼び出される JavaScript。		14.0	
onfocus	String	onfocus イベントが発生した場合(フォーカスがコンポーネントにある場合)に呼び出される JavaScript。		14.0	
onkeydown	String	onkeydown イベントが発生した場合(ユーザがキーボードのキーを押した場合)に呼び出される JavaScript。		14.0	
onkeypress	String	onkeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出される JavaScript。		14.0	
onkeyup	String	onkeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出される JavaScript。		14.0	
onmousedown	String	onmousedown イベントが発生した場合(ユーザがマウスボタンをクリックした場合)に呼び出される JavaScript。		14.0	
onmousemove	String	onmousemove イベントが発生した場合(ユーザがマウスポインタを移動した場合)に呼び出される JavaScript。		14.0	
onmouseout	String	onmouseout イベントが発生した場合(ユーザがコンポーネントからマウスポインタを移動した場合)に呼び出される JavaScript。		14.0	
onmouseover	String	onmouseover イベントが発生した場合(ユーザがマウスポインタをコンポーネントに重ねた場合)に呼び出される JavaScript。		14.0	
rendered	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global
required	Boolean	このコンポーネントが必須項目であるかどうかを指定する boolean 値。true に設定されている場合、このコンポーネントの値を指定する必要があります。選択されていない場合、この値はデフォルトの false に設定されます。		14.0	

	属性名	属性型	説明	必須項目	API バージョン	アクセス
1	size	Integer	表示されるファイル選択ボックスのサイズ。		14.0	
1 1	style	String	コンポーネントの表示に使用されるスタイル。主に、インライン CSS スタイルを追加するために使用されます。		14.0	
b d	cssClass	String	コンポーネントの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。		14.0	
1	tabIndex	Integer	ユーザが Tab キーを繰り返し押したときに、他のページコンポーネントと比較して、このコンポーネントが選択される順序。この値は、ユーザが Tab キーを押したときに選択される最初のコンポーネントを 0 として、0 ~ 32767 の整数である必要があります。		14.0	
1 u	label	String	コンポーネントにマウスを置いたときにコンポーネントの横に表示されるテキスト。		14.0	
	value	Blob	このコンポーネントに関連付けられているコントローラクラス変数を参照する差し込み項目。たとえば、コントローラクラスの関連付けられている変数の名前が myInputFile である場合、この変数を参照するには value="#{myInputFile}" を使用します。	はい	14.0	

apex:inputHidden

hidden 型の HTML 入力要素(ユーザに表示されない入力要素)です。このコンポーネントを使用して、ページ間で変数を渡します。

このコンポーネントでは、「html-」プレフィックスを使用した [HTML パススルー属性](#)がサポートされています。パススルー属性は、生成された [p](#) タグに適用されます。 >

例

```
    e n u dd p tlm pu < e}t l aH di p n dd a" N ilu I a"pi =H i " > /
```

上述の例では次の HTML を表示します。

```
t e u d n tdp eh u < e n dgi e="H pi m n dd " h y=" i I " p =H i " > /
```

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
d	String	i ページの他のコンポーネントが inputHidden コンポーネントを参照できるようにする識別子。		10.0	global
d mm	Boolean	i ページの項目に関連付けられている入力規則を処理することなく、このコンポーネントに関連付けられているアクションをすぐに実行するかどうかを指定する boolean 値。true に設定すると、アクションがすぐに実行され、入力規則はスキップされます。指定されていない場合、この値はデフォルトの false に設定されます。		11.0	global
r d r d	Boolean	d コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		10.0	global
r n dg	Boolean	i この inputHidden 項目が必須項目であるかどうかを指定する boolean 値。true に設定されている場合、この項目に値が指定されている必要があります。選択されていない場合、この値はデフォルトの false に設定されます。		10.0	global
l u	Object	a 非表示入力項目に関連付けられているコントローラクラス変数を参照する差し込み項目。たとえば、コントローラクラスの関連付けられている変数の名前が myHiddenVariable である場合、この変数を参照するには value="={!myHiddenVariable}" を使用します。		10.0	global

apex:inputSecret

password 型の HTML 入力要素です。このコンポーネントを使用して、ユーザが入力した値がマスクされる、Salesforce オブジェクトの項目に対応しないコントローラメソッドのユーザ入力を取得します。

このコンポーネントでは、「html-」プレフィックスを使用した [HTML パススルー属性](#)がサポートされています。パススルー属性は、生成された [p](#) タグに適用されます。 >

例

```
te e { u r p t&lt;puc < e}tel a u td p s n r a" N ilu c I a"pi =" s " > /
```

上述の例では次の HTML を表示します。

```
te e utd t net ip h u ok< ei epte re&lt; p t pS m n er" hy="u cs$ alu " p =a s " v a" " > /
```

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス	
cc	String	項目にフォーカスを置くキーボードのアクセスキー。項目にフォーカスが置かれている場合は、値を入力できます。		10.0	global	
l d d d d l l l g l ru o ol	String	項目の代替テキストの説明。		10.0	global	
dr bl d d mm b l g m h ru d c c	String Boolean String Boolean String String Integer String String String	生成された HTML コンポーネントの読み取り方向。使用可能な値には「RTL」(右から左) または「LTR」(左から右) があります。 この項目を無効な状態で表示するかどうかを指定する boolean 値。true に設定されている場合、この項目は無効な状態で表示されます。指定されていない場合、この値はデフォルトの false に設定されます。 ページの他のコンポーネントがチェックボックスコンポーネントを参照できるようにする識別子。 ページの項目に関連付けられている入力規則を処理することなく、このコンポーネントに関連付けられているアクションをすぐに実行するかどうかを指定する boolean 値。true に設定すると、アクションがすぐに実行され、入力規則はスキップされます。指定されていない場合、この値はデフォルトの false に設定されます。 コントロールの横に表示ラベルを表示し、エラーメッセージ内のコントロールを参照できるようにするテキスト値。 「en」または「en-US」など、生成された HTML 出力の基本言語。この属性についての詳細は、 W3C仕様 を参照してください。 この項目に入力できる最大文字数。整数として表されます。 onblur イベントが発生した場合(フォーカスが項目から離れた場合)に呼び出される JavaScript。 onchange イベントが発生した場合(ユーザが項目のコンテンツを変更した場合)に呼び出される JavaScript。 onclick イベントが発生した場合(ユーザが項目をクリックした場合)に呼び出される JavaScript。	i s i s i a a i a a a i i		10.0 10.0 10.0 10.0 10.0 23.0 10.0 10.0 10.0 10.0 10.0 10.0	global global global global global global global global global global global global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
dblclick	String	i ondblclick イベントが発生した場合(ユーザが項目をダブルクリックした場合)に呼び出される JavaScript。		10.0	global
focus	String	s onfocus イベントが発生した場合(フォーカスが項目にある場合)に呼び出される JavaScript。		10.0	global
keydown	String	y onkeydown イベントが発生した場合(ユーザがキーボードのキーを押した場合)に呼び出される JavaScript。		10.0	global
keypress	String	y onkeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出される JavaScript。		10.0	global
keyup	String	y onkeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出される JavaScript。		10.0	global
mousedown	String	s onmousedown イベントが発生した場合(ユーザがマウスボタンをクリックした場合)に呼び出される JavaScript。		10.0	global
mousemove	String	s onmousemove イベントが発生した場合(ユーザがマウスポインタを移動した場合)に呼び出される JavaScript。		10.0	global
mouseout	String	s onmouseout イベントが発生した場合(ユーザが項目からマウスポインタを移動した場合)に呼び出される JavaScript。		10.0	global
mouseover	String	s onmouseover イベントが発生した場合(ユーザがマウスポインタを項目に重ねた場合)に呼び出される JavaScript。		10.0	global
mouseup	String	s onmouseup イベントが発生した場合(ユーザがマウスボタンを放した場合)に呼び出される JavaScript。		10.0	global
select	String	s onselect イベントが発生した場合(ユーザがこの項目のテキストを選択した場合)に呼び出される JavaScript。		10.0	global
disabled	Boolean	a この項目が参照のみとして表示されるかどうかを指定する boolean 値。true に設定されている場合、項目の値は変更できません。選択されていない場合、この値はデフォルトの false に設定されます。		10.0	global
readOnly	Boolean	i s 前回入力したパスワードをこのフォームに表示するかどうかを指定する boolean 値。true に設定されている場合、前回入力した値がマスクされて表示されます。指定されていない場合、この値はデフォルトの false に設定されます。		10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
rendered	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		10.0	global
required	Boolean	この項目が必須項目であるかどうかを指定する boolean 値。 true に設定されている場合、この項目の値を指定する必要があります。選択されていない場合、この値はデフォルトの false に設定されます。	i	10.0	global
label	Integer	項目の幅。一度に表示可能な文字数で表されます。	s	10.0	global
labelClass	String	inputSecret コンポーネントの表示に使用されるスタイル。主に、オンライン CSS スタイルを追加するために使用されます。	s	10.0	global
labelStyleClass	String	inputSecret コンポーネントの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。	s	10.0	global
TabIndex	String	ユーザが Tab キーを繰り返し押したときに、他のページコンポーネントと比較して、この項目が選択される順序。この値は、ユーザが Tab キーを押したときに選択される最初のコンポーネントを 0 として、0 ~ 32767 の整数である必要があります。	i	10.0	global
toolTip	String	ユーザがコンポーネントにマウスポインタを重ねたときにツールチップとして表示されるテキスト。	i	10.0	global
value	Object	この項目に関連付けられているコントローラクラス変数を参照する差し込み項目。たとえば、コントローラクラスの関連付けられている変数の名前が myPasswordField である場合、この変数を参照するには value="={!myPasswordField}" を使用します。	a	10.0	global

apex:inputText

text 型の HTML 入力要素です。このコンポーネントを使用して、Salesforce オブジェクトの項目に対応しないコントローラメソッドのユーザ入力を取得します。

このコンポーネントでは、Salesforce のスタイル設定を使用しません。また、オブジェクトの項目、その他のすべてのデータにも対応していないため、ユーザが入力した値を使用するにはカスタムコードが必要です。

このコンポーネントでは、「html-」プレフィックスを使用した [HTML パススルー属性](#)がサポートされています。パススルー属性は、生成された [p](#) タグに適用されます。 >

例

```
c n txe { u ptln up Te)te l u t x d tp n v a!Mu! I T p" i =" " > /
```

上述の例では次の HTML を表示します。

```
te tux d t net ep t hux e < h u p t x="pt m "hu y="I T "p =a " > /
```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
cc	String	項目にフォーカスを置くキーボードのアクセスキー。テキストボックスにフォーカスが置かれている場合は、ユーザが項目値を選択または選択解除できます。		10.0	global
cl	String	a 項目の代替テキストの説明。		10.0	global
dr	String	i 生成された HTML コンポーネントの読み取り方向。使用可能な値には「RTL」(右から左) または「LTR」(左から右) があります。		10.0	global
dbl d	Boolean	s i このテキストボックスを無効な状態で表示するかどうかを指定する boolean 値。true に設定されている場合、テキストボックスは無効な状態で表示されます。指定されていない場合、この値はデフォルトの false に設定されます。		10.0	global
d	String	i ページの他のコンポーネントが項目コンポーネントを参照できるようにする識別子。		10.0	global
l b l	String	a コントロールの横に表示ラベルを表示し、エラーメッセージ内にコントロールを参照できるようにするテキスト値。		23.0	
1 g	String	a 「en」または「en-US」など、生成された HTML 出力の基本言語。この属性についての詳細は、 W3C 仕様 を参照してください。		10.0	global
1	Object	s i s 入力項目に関連付けられた HTML ブロックに追加するオートコンプリート値のリスト。	<	29.0	ia sa >
t t n or	l	属性は、カンマ区切りの静的文列または Visualforce 式として指定されます。式は、カンマ区切り文列またはオブジェクトのリストに解決できます。リスト要素には任意のデータ型を設定できますが、その型は、Apex 言語機能として、または <code>g</code> メソッドを介して、文列に変換できる必要があります。			
		この項目に入力？ナ'顕バム棊-蜻垂メC鉤「吳女騎焰遂	10.0		global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
onblur	String	onblur イベントが発生した場合(フォーカスが項目から離れた場合)に呼び出される JavaScript。		10.0	global
onchange	String	onchange イベントが発生した場合(ユーザが項目のコンテンツを変更した場合)に呼び出される JavaScript。		10.0	global
onclick	String	onclick イベントが発生した場合(ユーザが項目をクリックした場合)に呼び出される JavaScript。	i	10.0	global
ondblclick	String	ondblclick イベントが発生した場合(ユーザが項目をダブルクリックした場合)に呼び出される JavaScript。	i	10.0	global
onfocus	String	onfocus イベントが発生した場合(フォーカスが項目にある場合)に呼び出される JavaScript。	s	10.0	global
onkeydown	String	onkeydown イベントが発生した場合(ユーザがキーボードのキーを押した場合)に呼び出される JavaScript。	y	10.0	global
onkeypress	String	onkeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出される JavaScript。	y	10.0	global
onkeyup	String	onkeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出される JavaScript。	y	10.0	global
onmousedown	String	onmousedown イベントが発生した場合(ユーザがマウスボタンをクリックした場合)に呼び出される JavaScript。	s	10.0	global
onmousemove	String	onmousemove イベントが発生した場合(ユーザがマウスポインタを移動した場合)に呼び出される JavaScript。	s	10.0	global
onmouseout	String	onmouseout イベントが発生した場合(ユーザが項目からマウスポインタを移動した場合)に呼び出される JavaScript。	s	10.0	global
onmouseover	String	onmouseover イベントが発生した場合(ユーザがマウスポインタを項目に重ねた場合)に呼び出される JavaScript。	s	10.0	global
onmouseup	String	onmouseup イベントが発生した場合(ユーザがマウスボタンを放した場合)に呼び出される JavaScript。	s	10.0	global
rendered	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		10.0	global
required	Boolean	この項目が必須項目であるかどうかを指定する boolean 値。true に設定されている場合、この項目	i	10.0	global

属性名	属性型	説明	必須項目	API バージョン	アクセス
width					
	Integer	この値を指定する必要があります。選択されていない場合、この値はデフォルトの false に設定されます。		10.0	global
style	String	inputText コンポーネントの表示に使用されるスタイル。主に、INLINE CSS スタイルを追加するために使用されます。		10.0	global
cssClass	String	inputText コンポーネントの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。		10.0	global
tabIndex	String	ユーザが Tab キーを繰り返し押したときに、他のページコンポーネントと比較して、この項目が選択される順序。この値は、ユーザが Tab キーを押したときに選択される最初のコンポーネントを 0 として、0 ~ 32767 の整数である必要があります。		10.0	global
label	String	ユーザがコンポーネントにマウスポインタを重ねたときにツールチップとして表示されるテキスト。		10.0	global
value	Object	この項目に関連付けられているコントローラクラス変数を参照する差し込み項目。たとえば、コントローラクラスの関連付けられている変数の名前が myTextField である場合、この変数を参照するには value="{!!myTextField}" を使用します。		10.0	global

apex:inputTextarea

テキストエリア入力要素です。このコンポーネントを使用して、値にテキストエリアが必要な、Salesforce オブジェクトの項目に対応しないコントローラメソッドのユーザ入力を取得します。

このコンポーネントでは、「html-」プレフィックスを使用した [HTML パススルー属性](#)がサポートされています。パススルー属性は、生成された [タグ](#)に適用されます。

例

```

<apex:inputTextarea value="<!>{!myText}<!>"/>

```

```
e et n rn{ d rnp t eto c <otp>d cnr} cs o á p! p a s i i < >/  
e e n t dgncp o co < p >: apsi i < >/  
x et n ex eue m dep{ t laetu c ot ad dr} cs 'b c a" p! p a s i i < >/  
x tit @Bndmme pc o lt u < en a o } a v a" v Sa ia = a > /  
ex B lgo p p c < k/a a >  
x ro m p f < /a >  
g p p < /a >
```

上述の例では次の HTML を表示します。

```
<div>{ g dh l who < ot -f cnr ncv o coa p! a s i i < >-  
ro edem ndj t e grdn o de e" indj tD ea sgrdi "btc shi o hD a s ip i " s"  
x oen x ce db o pt i &oww pas pa 'no rdm y=d ia ic a / - - - > /  
e ue n nddp epe ndj t egrih "b cle-@ipndj a s grdi "vo c a" ip _ D a s i  
n nto d g p g < ! -- i v sa >-  
e et n rn d rnp eto c ettp > b fT rsdi li i ia p a < sa >/  
e e n t dgncp o co < p >: apsi i < >/  
et ex r ede ndj t <gerde wa adja ip : D: asá i li D s" > /  
ee mndj t gerde wa cdja ip : D: asá i li D s" > /  
ln b c ss=a " > /  
n tl o d g g < ! -- $ v sa >-  
ro m f < / >
```

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
blur	String	onblur イベントが発生した場合(フォーカスがテキストエリアから離れた場合)に呼び出される JavaScript。		10.0	global
change	String	onchange イベントが発生した場合(ユーザがテキストエリアのコンテンツを変更した場合)に呼び出される JavaScript。		10.0	global
click	String	onclick イベントが発生した場合(ユーザがテキストエリアをクリックした場合)に呼び出される JavaScript。	i	10.0	global
dblclick	String	ondblclick イベントが発生した場合(ユーザがテキストエリアをダブルクリックした場合)に呼び出される JavaScript。	i	10.0	global
focus	String	onfocus イベントが発生した場合(フォーカスがテキストエリアにある場合)に呼び出される JavaScript。	s	10.0	global
keydown	String	onkeydown イベントが発生した場合(ユーザがキーボードのキーを押した場合)に呼び出される JavaScript。	y	10.0	global
keypress	String	onkeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出される JavaScript。	y	10.0	global
keyup	String	onkeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出される JavaScript。	y	10.0	global
mousedown	String	onmousedown イベントが発生した場合(ユーザがマウスボタンをクリックした場合)に呼び出される JavaScript。	s	10.0	global
mousemove	String	onmousemove イベントが発生した場合(ユーザがマウスポインタを移動した場合)に呼び出される JavaScript。	s	10.0	global
mouseout	String	onmouseout イベントが発生した場合(ユーザがテキストエリアからマウスポインタを移動した場合)に呼び出される JavaScript。	s	10.0	global
mouseover	String	onmouseover イベントが発生した場合(ユーザがマウスポインタをテキストエリアに重ねた場合)に呼び出される JavaScript。	sv	10.0	global
mouseup	String	onmouseup イベントが発生した場合(ユーザがマウスボタンを放した場合)に呼び出される JavaScript。	s	10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
onselect	String	onselect イベントが発生した場合(ユーザがテキストエリアのテキストを選択した場合)に呼び出される JavaScript。		10.0	global
readOnly	Boolean	このテキストエリアを参照のみとして表示するかどうかを指定する boolean 値。true に設定されている場合、テキストエリアの値を変更することはできません。選択されていない場合、この値はデフォルトの false に設定されます。		10.0	global
rendered	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		10.0	global
required	Boolean	このテキストエリアが必須項目であるかどうかを指定する boolean 値。true に設定されている場合、このテキストエリアの値を指定する必要があります。選択されていない場合、この値はデフォルトの false に設定されます。		10.0	global
value	Boolean	このテキストエリアをリッチテキストまたはプレーンテキストのどちらで保存するかを指定する boolean 値。true に設定されている場合、この値はリッチテキストとして保存されます。選択されていない場合、この値はデフォルトの false に設定されます。		10.0	global
rows	Integer	テキストエリアの高さ。一度に表示可能な行数で表されます。		10.0	global
style	String	テキストエリアコンポーネントの表示に使用されるスタイル。主に、インライン CSS スタイルを追加するために使用されます。		10.0	global
styleClass	String	テキストエリアコンポーネントの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。		10.0	global
tabIndex	String	ユーザが Tab キーを繰り返し押したときに、他のページコンポーネントと比較して、このテキストエリアが選択される順序。この値は、ユーザが Tab キーを押したときに選択される最初のコンポーネントを 0 として、0 ~ 32767 の整数である必要があります。		10.0	global
title	String	ユーザがコンポーネントにマウスポインタを重ねたときにツールチップとして表示されるテキスト。		10.0	global
valueObject	Object	このテキストエリアに関連付けられているコントローラクラス変数を参照する差し込み項目。たとえば、コントローラクラスの関連付けられている		10.0	global

属性名	属性型	説明	必須項目	API バージョン	アクセス
変数の名前が myLongDescription である場合、この変数を参照するには value="={!myLongDescription}" を使用します。					

apex:insert

- 他の Visualforce ページで `apex:component` コンポーネントで定義されている必要のある名前指定の領域を宣言するテンプレートコンポーネントです。複数のページ間でデータを共有するには、このコンポーネントを `apex:repeat` および `apex:forEach` コンポーネントと併用します。

例

```

<apex:insert name="myComponent">
    <apex:repeat value="someList" index="index">
        <apex:insert name="myComponent">
            <apex:repeat value="innerList" index="innerIndex">
                <apex:outputText value="Outer Item: " />
                <apex:outputText value="Inner Item: " />
            </apex:repeat>
        </apex:insert>
    </apex:repeat>
</apex:insert>

```

上述の例では次の HTML を表示します。

```

<div>
    <div>
        <div>
            <div>
                Outer Item: Outer Item: Outer Item: Outer Item:
            </div>
        </div>
    </div>
</div>

```

属性

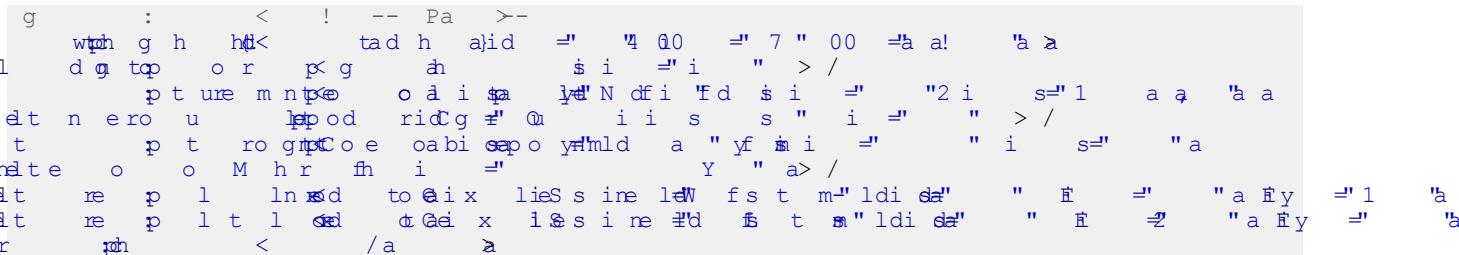
属性名	属性型	説明	必須項目	API バージョン	アクセス
name	String	この Visualforce ページに挿入するコンテンツを提供する、一致する定義タグの名前。	はい	10.0	global

apex:legend

グラフの凡例を定義します。このコンポーネントでは、コンポーネントの legend 属性で使用されるデフォルト以外の追加設定オプションを提供します。

注: このコンポーネントはコンポーネントで囲む必要があります。

例



属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
f	String	CSS スタイルのフォント定義として、凡例テキストに使用するフォント。指定されていない場合、この値はデフォルトの「12px Helvetica」に設定されます。		23.0	
i	String	ページの他のコンポーネントがグラフコンポーネントを参照できるようにする識別子。		23.0	global
p	Integer	凡例の境界線と凡例のコンテンツ間のスペース(ピクセル単位)。		23.0	
p	String	グラフに対する凡例の位置。有効なオプションは、次のとおりです。 <ul style="list-style-type: none"> left right top bottom 	はい	23.0	
Boolean		グラフの凡例をグラフに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		23.0	
s	Integer	凡例項目間のスペース(ピクセル単位)。		23.0	

apex:lineSeries

Visualforce 線形グラフで線で結ばれた点として表示されるデータ系列です。少なくとも、各点のX値およびY値として使用するデータコレクションの項目、および目盛りとして使用するX軸およびY軸を指定する必要があります。

注: このコンポーネントは [コンポーネントで囲む必要があります](#)。1つのグラフに複数のコンポーネントおよび [コンポーネントを含めることができます](#)。コンポーネントと [コンポーネントを追加できますが、判読しにくい結果になる可能性があります](#)。

例

```
g : < ! -- Pa >-
t r wth g h hd< tad h ajid = 4 00 = 7 " 00 =a a! "a a
x xet e pt ure m npe o a i sp yd N dfi fd si = "2 i s="1 a a "a a
t t elt n ero u lpod ridg @ i i s s " i = " > /
x xet e t p t ro gr coe oabi esp o yhml d a "yf bi i = " i s=" " a
c t nlate o o M hr fh i = " Y " a> /
xe en telt re p l lnkd to @ix lies sine lew fst m" ldi sa" " a = " a fly
et ell er u fr me e o izr m " a myl ss" aik s" " 4 aE = "#FF
xe en tel t re p l t l sel t@el x lies sine d t s" ldi sa" " a = " a fly
e er m e er l zpemc ar y" hil " aik E" E 4 aE = "# 38 F" 5 > /
t r ph < /a a
```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
axis	String	<p>このグラフの系列のバインド先の軸。グラフの4辺の境界の1つである必要があります。</p> <ul style="list-style-type: none"> left right top bottom <p>この軸のバインド先は同階層のコンポーネントによって定義される必要があります。</p>	はい	23.0	
fill	Boolean	線の下の領域を塗りつぶすかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの false に設定されます。		23.0	
fillColor	String	線の下の領域を塗りつぶすために使用する色を HTML スタイルの(16進)色として指定する文字列。指定されていない場合、塗りつぶしの色は線の色と一致します。fill が true に設定されている場合にのみ使用されます。		26.0	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
l g	Boolean	マウスポインタを重ねたときに系列の折れ線の各点を強調表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		23.0	
t l gr gd h h	String	系列の折れ線が強調表示されるときに上に重ねて描画される線の太さを指定する文字列。		26.0	
d	String	ページの他のコンポーネントがグラフコンポーネントを参照できるようにする識別子。		23.0	global
r m ll	String	HTML スタイルの(16進)色として指定されるこの系列のデータポイントマーカーの色。指定されていない場合、マーカーの色は線の色と一致します。		23.0	
z r m	Integer	この系列の各データポイントマーカーのサイズ。指定されていない場合、この値はデフォルトの「3」に設定されます。		23.0	
r m	String	この系列の各データポイントマーカーの形状。有効なオプションは、次のとおりです。 <ul style="list-style-type: none">• circle• cross 指定されていない場合、マーカーの形状は一連の形状から選択されます。		23.0	
o	String	この系列の折れ線と重なっている塗りつぶされた領域の不透明度を表す 0 ~ 1 までの小数値。指定されていない場合、デフォルトの「0.3」に設定されます。fill が true に設定されている場合にのみ使用されます。		26.0	
r d r d	Boolean	グラフでグラフ系列を表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		23.0	
r nd r r	String	各データポイントが表示される方法を追加または上書きする JavaScript 関数の名前を指定する文字列。追加のスタイルを指定またはデータを追加するのに実装します。		26.0	
no	Boolean	このグラフ系列をグラフの凡例に追加するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		23.0	
oom	Integer	折れ線の平滑度を指定する整数。数値が低いほど、高い平滑度が適用されます。0(ゼロ)を指定すると、平滑化が無効になり、系列の点と点を結ぶ線には直線が使用されます。		26.0	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
color	Color	HTMLスタイルの(16進)色として、この系列の折れ線の色を指定する文字列。指定されていない場合、色はグラフの colorSet またはテーマから順に使用されます。		26.0	
strokeWidth	Integer	この系列の折れ線の太さを指定する整数。		26.0	
pointLabel	Boolean	マウスポインタを重ねたときに各データポイントマーカーのツールチップを表示するかどうかを指定する boolean 値。このツールチップの形式は <xField>: <yField> です。指定されていない場合、この値はデフォルトの true に設定されます。		23.0	
title	String	このグラフ系列のタイトル。グラフの凡例に表示されます。		23.0	
xLabel	String	系列のデータポイントごとの X 軸の値の取得元である、グラフデータに指定された各レコードの項目。この項目はグラフデータのすべてのレコードに存在している必要があります。	はい	23.0	
yLabel	String	系列のデータポイントごとの Y 軸の値の取得元である、グラフデータに指定された各レコードの項目。この項目はグラフデータのすべてのレコードに存在している必要があります。	はい	23.0	

apex:listViews

現在選択されているビューのレコードの関連付けられたリストを含む、オブジェクトのリストビュー選択リストです。標準の Salesforce アプリケーションでは、このコンポーネントは特定のオブジェクトのメインタブに表示されます。

listViews

```
<apex:repeat items="listViews">
    <apex:repeat items="listView">
        <apex:repeat items="listViewFields">
            <apex:repeat items="listViewFieldLabels">
                <apex:repeat items="listViewFieldValues">
                    <apex:repeat items="listViewFieldTypes">
                        <apex:repeat items="listViewFieldRelationships">
                            <apex:repeat items="listViewFieldRelationships">
                                <apex:repeat items="listViewFieldRelationships">
                                    <apex:repeat items="listViewFieldRelationships">
                                        <apex:repeat items="listViewFieldRelationships">
                                            <apex:repeat items="listViewFieldRelationships">
                                                <apex:repeat items="listViewFieldRelationships">
                                                    <apex:repeat items="listViewFieldRelationships">
                                                        <apex:repeat items="listViewFieldRelationships">
                                                            <apex:repeat items="listViewFieldRelationships">
                                                                <apex:repeat items="listViewFieldRelationships">
                                                                    <apex:repeat items="listViewFieldRelationships">
                                                                        <apex:repeat items="listViewFieldRelationships">
                                                                            <apex:repeat items="listViewFieldRelationships">
                                                                                <apex:repeat items="listViewFieldRelationships">
                                                                                    <apex:repeat items="listViewFieldRelationships">
                                                                                        <apex:repeat items="listViewFieldRelationships">
                                                                                            <apex:repeat items="listViewFieldRelationships">
                                                                                                <apex:repeat items="listViewFieldRelationships">
                                                                                                    <apex:repeat items="listViewFieldRelationships">
                                                                                                        <apex:repeat items="listViewFieldRelationships">
                                                                                                            <apex:repeat items="listViewFieldRelationships">
                                                                                                                <apex:repeat items="listViewFieldRelationships">
                                                                                                                    <apex:repeat items="listViewFieldRelationships">
................................................................
```

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
id	String	ページの他のコンポーネントが listViews コンポーネントを参照できるようにする識別子。		10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
					ス
r	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	10.0	global	
p	String	type="Account" または type="My_Custom_Object__c" はいなど、リストビューを表示する Salesforce オブジェクト。	10.0	global	

Facet

facet名	説明	APIバージョン
b	レコードの表示リストの本文に表示されるコンポーネント。表示リストの本文の表示は、name="body" を含む facet によって制御されるため、listViews コンポーネントに表示される body facet の順序は重要ではありません。また、body facet を定義すると、この facet によって、通常リストビューの一部として表示されるレコードのリストが置き換えられます。	10.0
o	レコードの表示リストのフッターに表示されるコンポーネント。表示リストの最下部の表示は、name="footer" を含む facet によって制御されるため、listViews コンポーネントの本文に表示される footer facet の順序は重要ではありません。	10.0
d	レコードの表示リストのヘッダーに表示されるコンポーネント。リストの最上部の表示は、name="header" を含む facet によって制御されるため、listViews コンポーネントの本文に表示される header facet の順序は重要ではありません。	10.0

apex:logCallPublisher

apex:logCallPublisher（「活動の記録」パブリッシャー）では、ケースフィードを使用するサポートエージェントは、顧客の活動のログを作成できます。このコンポーネントは、ケースでケースフィード、Chatter、およびフィード追跡を有効にしている組織でのみ使用できます。

この例は「活動の記録」パブリッシャーを表示します。

```

<apex:logCallPublisher>
</apex:logCallPublisher>

```

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
udlE	do C p Boolean a	「活動の記録」本文が空の場合に本文を折りたたんで上下のサイズを小さくするかどうかを指定する boolean 値。		25.0	
d	I id i	y「活動の記録」パブリッシャーを表示するレコードのエンティティ ID。現在のバージョンでは、ケースレコード ID のみがサポートされています。	はい	25.0	
d	String i	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
lobllg	do C	パブリッシャーが表示されるときの「活動の記録」本文の最初のテキスト値。		25.0	
t Boll glo	C g h String	H a「活動の記録」本文の高さ(em 単位)。		25.0	
obu mru	String	iS F活動の記録に失敗した場合に呼び出される JavaScript。		25.0	
obu m	cc String	iS 活動が正しく記録されたときに呼び出される JavaScript。		25.0	
r d r d	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global
r d r	Object R	活動が正しく記録されたときに再作成される 1 つ以上のコンポーネントの ID。この値には、単一の ID、ID のカンマ区切りのリスト、または ID のリストまたはコレクションの差し込み項目の式を使用できます。		25.0	
An dd d	ld Boolean s i i	パブリッシャーのレイアウトに定義された追加項目を表示するかどうかを示す boolean 値。		25.0	
obu m	Boolean s i s	送信ボタンを表示するかどうかを指定する boolean 値。		25.0	
bu m o m	String s i	「活動の記録」パブリッシャーの送信ボタンの名前。		25.0	
ku m o mc	String s i F	活動ログを公開するために JavaScript からコールできる関数の名前。		25.0	
l	String i	「活動の記録」パブリッシャーヘッダーに表示されるタイトル。		25.0	
d	h String i	パブリッシャーの幅(ピクセル(px) またはパーセント(%) 単位)。		25.0	

apex:message

警告またはエラーなど、特定のコンポーネントに対するメッセージです。またコンポーネントがページに含まれていない場合、ほとんどの警告およびエラーメッセージはデバッグログにのみ表示されます。

例

```

e   ex ro te e   lhe m o d r pr Ebl i se ope p an o e ec li yoy s sscá V pa i s a
e   et lnd h @t u rn cd i c v á ha a i U L R .
e   ro l m t ept f F t nta iquln t 00 000R003h S g ddDb h , s: U L R s
e   e   he neap xe fc sc m /0sp i s a /a I y Ra1B 00 000R003 5
e   e   el hro e l fo V tSdt itpa & secur o l sntgTm info maf i a i ia .

t en g : < ! -- Pa >-
t en 1 gortpla mp c t set ld MaC b talAn e'u ycc " a Sy" "
{ m o orre ro lato bly d to i an fjr oh :g ; i- s ;
{ l m re re ro plone rw ct .to y n fjr oh :g ; i- s ;
1 < s / y >

x ex ro m p f < a >
e ex te B e l g o p d p cll <@ k r a r } a i mH $" U ! s E s N a >
e e w@t u rn t egn n du tY ac m a o i cc ! a . a a .
e t n tb r m en po x oet n c < e { :Nu ldp lt ym as out abce m ov } o affc ! a cN L i
t n t lE bee htp rc dm ec had a hawt ce o a a n k , h ih v psa s p a as<
e t n ee ee b r m e pl xo et fe < e { :Nu ldp lt ym ys ouee abce m lv m aff ! p a N o
t n t lE bee htp rc dm ec had a hawt ce o a a n k , h ih v psa s p a as<
x t t obEndmm wo o < e a l u a ia = ts " a" v Sa > /
e x e m go n o f <@ d a oss lta l = m io @ oir ad a" E y ss=a pi a " < > /
e x e eem go t m f <@ d peoss deal = lgy m ir ad ap E y ss=a p y " < > /
x ex B lgo p p c < k/a a >
x ro m p f < /a a >

n bl t el n pc or ld M C *** / *** /
c An t om cou cc i ss a y a ;
e e e e ble en { grc f c i Ra & a ( )
r t td n ou p cc y a a ;
} e x t e { l ncm d o Ec p D a ( i ) Ba . ass sa ( ; )
e x A e e e gdx p g M
} n m ru lh ;
} t e blte { r p cg g m i is N a ( ) ;
n rterm or ld M C i y N a ( ) ;
etblAit nou fcdh i ( a == ) )
t en e tou e cd ne d n ceem br a in a [ tli b ip, m a, o fcrAno ysmcquf cc i as
w e rex dA eh te negrntpe e t ge =gt rm Ba . gd Pa ( . ) Pa a s ( . ) i ( ' ; )
```

n rt rm ou cc a ;

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
direction	String	生成された HTML コンポーネントの読み取り方向。使用可能な値には「RTL」(右から左) または「LTR」(左から右) があります。		10.0	global
referenceId	String	メッセージに関連付けられているコンポーネントの ID。		10.0	global
componentId	String	ページの他のコンポーネントがメッセージコンポーネントを参照できるようにする識別子。		10.0	global
language	String	「en」または「en-US」など、生成された HTML 出力の基本言語。この属性についての詳細は、 W3C 仕様 を参照してください。		10.0	global
rendered	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		10.0	global
style	String	メッセージの表示に使用されるスタイル。主に、インライン CSS スタイルを追加するために使用されます。		10.0	global
styleClass	String	メッセージの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。		10.0	global
title	String	ユーザがコンポーネントにマウスポインタを重ねたときにツールチップとして表示されるテキスト。		10.0	global

apex:messages

現在のページのすべてのコンポーネントに生成されたすべてのメッセージです。また、コンポーネントがページに含まれていない場合、ほとんどの警告およびエラーメッセージはデバッグログにのみ表示されます。

このコンポーネントでは、「html-」プレフィックスを使用した **HTML パススルー属性**がサポートされています。パススルー属性は、生成された `message` タグに適用されます (各メッセージはリスト項目に含まれます)。

例

```

e   e   he n erop xe fc sc m /oSp i s a /a I /y EalB 00 000R003 5
e   e   el hro e l for V t Sdt i ts & secur o l so t@fotm info m a f i a i a . i a .

g : < ! -- Pa >-
t e n g orple pc t set l@ MaC b talAn e'u y cc " a Sy" " >
x e m @ < a ss sa > /
x ex ro m p f < a >
e e w e stern tegn n du TY ac m aio cc ! a . a a .
e t tb r m en po x oet fnc < e {Nu ldp lt yen as ou t@bce m ov }o effc ! a cN L
t n @ t le bee ktp rc dm ec had a hawt ce o a a n k , h ih w ppa s p a as<
e ee ee b r m es pl x o etfe < e {Nu ldp lt yen ys ouee abce m lv on eff ! pa N
t n @ t le bee ktp rc dm ec had a hawt ce o a a n k , h ih w ppa s p a as<
x tit obEndmm po < e a l u a ia = hs " a" v Sa > /
ex ro m p f p < /a >
g p p < /a >

n or l@ r C *** / ***
bli t el n pc or l@ M C i ss a y a ;
c An t on cou cc

e e e e ble en { grc f c i Ra w a ( )
r t d n ou p cc y a a ;
} e x t e { l nc m dh o Ec p D a ( i ) Ba . ass sa ( ; )
e A e e e gdx p g M
} n m ru lu ;
} t e blte { r p cg g m i is N a ( )
n rterm or l@ M C y ; ' '
} etblA@t t n nou fc@l i ( a == ) )
t ene tou e cd ne d@ ceem br @n @ [ t@ b p m@ a, o ffc@Ano ysmc@uf cc i as
e re x d A eh te negrntpe e t ge =gt rm Ba . gd Pa(.) ] Pa a s ( . ) i( ' ; ')
n rt rm ou cc a ;
}

```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
dir	String	i 生成された HTML コンポーネントの読み取り方 向。使用可能な値には「RTL」(右から左) または 「LTR」(左から右) があります。		10.0	global
nlBody	Boolean	nl タライアント ID に関連付けられていないメッセー ジのみを表示するかどうかを指定する boolean 値。		10.0	global

	属性名	属性型	説明	必須項目	API バージョン	アクセス
d	label	String	指定されていない場合、この値はデフォルトの false に設定されます。			
l	language	String	ページの他のコンポーネントがメッセージコンポーネントを参照できるようにする識別子。	10.0	global	
l	outputType	String	「en」または「en-US」など、生成された HTML 出力の基本言語。この属性についての詳細は、 W3C 仕様 を参照してください。	10.0	global	
r d r d	rendered	Boolean	エラーメッセージの表示に使用されるレイアウトの種別。この属性の使用できる値には、「list」または「table」があります。指定されていない場合、この値はデフォルトの「list」に設定されます。	10.0	global	
l	style	String	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	10.0	global	
l l	styleClass	String	メッセージの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。	10.0	global	
l	title	String	ユーザがコンポーネントにマウスポインタを重ねたときにツールチップとして表示されるテキスト。	10.0	global	

apex:milestoneTracker

マイルストントラッカーを表示します。

次の例はマイルストントラッカーを表示します。

```
example t      n_tegndprde por lde & ec      aet o&cd r dh ra u      =! s " a &      " >
example et     end m totm re <cd T aji k Id c a i y= ! s ia " > /
example ex     g p p < /a >
```

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
d	I String	i マイルストンを表示するレコードのエンティティ ID。	はい	29.0	
d	String	i ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
r d r d	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global

apex:outputField

Salesforce オブジェクトの項目の表示ラベルと値の参照のみ表示です。 `<apex:outputField value="マイルストンID" />` コンポーネントは、ユーザへの表示方法などの関連項目の属性を考慮します。たとえば、指定された `<apex:outputField value="通貨項目ID" />` コンポーネントが通貨項目である場合、適切な通貨記号が表示されます。同様に `<apex:outputField value="URL" />` コンポーネントが参照項目または URL である場合、項目の値はリンクとして表示されます。

[設定] の項目にカスタムヘルプが定義されている場合は、その項目は `label="カスタムヘルプ" value="カスタムヘルプの値" />` または `label="カスタムヘルプ" value="カスタムヘルプの値" />` で表示されます。Visualforce ページにカスタムヘルプを表示するためには Salesforce のページヘッダーを表示する必要があります。カスタムヘルプの表示を上書きするには、
`<apex:outputField value="カスタムヘルプの本文" style="font-family: inherit; font-size: inherit; font-weight: inherit; color: inherit; margin-left: 10px;" />`

リッチテキストエリアデータ型の使用は、Salesforce.com API バージョン 18.0 より後のバージョンを実行するページのこのコンポーネントに限られます。

このコンポーネントでは、「html-」プレフィックスを使用した `HTML パススルー属性` がサポートされています。パススルー属性は、生成されたコンテナタグ `<apex:outputField value="マイルストンID" />` に適用されます。

例

```

<apex:outputField value="マイルストンID" />
<apex:outputField value="通貨項目ID" />
<apex:outputField value="URL" />
<apex:outputField value="カスタムヘルプの値" />
<apex:outputField value="カスタムヘルプの本文" style="font-family: inherit; font-size: inherit; font-weight: inherit; color: inherit; margin-left: 10px;" />

```

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
dir	String	i 生成された HTML コンポーネントの読み取り方 向。使用可能な値には「RTL」(右から左) または 「LTR」(左から右) があります。	10.0	global	
label	String	i ページの他のコンポーネントが出力項目コンポー ネントを参照できるようにする識別子。	10.0	global	
labelText	String	a コンポーネントの表示ラベルとして使用される文 字列値。	23.0		
language	String	a 「en」または「en-US」など、生成された HTML 出力の基本言語。この属性についての詳細は、 W3C 仕様 を参照してください。	10.0	global	
rendered	Boolean	 コンポーネントをページに表示するかどうかを指 定する boolean 値。指定されていない場合、この値 はデフォルトの true に設定されます。	10.0	global	
style	String	s y 出力項目コンポーネントの表示に使用されるス タイル。主に、インライン CSS スタイルを追加する ために使用されます。この属性を設定できない値 もあります。テキストにクラス名が必要な場合は、 ラップ用 span タグを使用します。	10.0	global	
styleClass	String	s y 出力項目コンポーネントの表示に使用されるス タイルクラス。主に、外部 CSS スタイルシートを使 用するときに適用される CSS スタイルを指定する ために使用されます。この属性を設定できない値 もあります。テキストにクラス名が必要な場合は、 ラップ用 span タグを使用します。	10.0	global	
title	String	i ユーザがコンポーネントにマウスポインタを重ね たときにツールチップとして表示されるテキスト。	10.0	global	
value	Object	a この出力項目に関連付けられている Salesforce 項目 を参照する差し込み項目。たとえば、取引先の名 前項目の出力項目を表示する必要がある場合は、 value="={!account.name}" を使用します。期間指定換 算レートを使用して項目の値が計算される場合、 この出力項目を currency 型の差し込み項目と関連 付けることはできません。	10.0	global	

apex:outputLabel

入力または出力項目の表示ラベルです。このコンポーネントを使用して、Salesforce オブジェクトの項目に対応しないコントローラメソッドに表示ラベルを作成します。

このコンポーネントでは、「html-」プレフィックスを使用した **HTML パススルー属性**がサポートされています。パススルー属性は、[生成された](#) **タグ**に適用されます。[a>](#)

例

```
te n e ou eu b k l xp e t o o d o x v k d c " k = " > /  
e u b o t k a < e t k d u d p x v b a i c a" ik = " > /
```

上述の例では次の HTML を表示します。

```
e tle b l ro ex f k <c b o l k = " k " > < / a>  
d r u d s t e p b d k i n t h i l o k c x m " k d y="k c " k = a " > /
```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
cc	String	表示ラベルおよびその関連項目にフォーカスを置くキーボードのアクセキー。		10.0	global
dir	String	生成された HTML コンポーネントの読み取り方向。使用可能な値には「RTL」(右から左) または「LTR」(左から右) があります。		10.0	global
escape	Boolean	このコンポーネントが生成する HTML 出力で、特殊な HTML および XML 文字をエスケープするかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。たとえば、表示ラベルに「>」記号を追加するには、記号の文字エスケープシーケンスを使って escape="false" に設定する必要があります。escape="false" が指定されていない場合、文字エスケープシーケンスは記述されたとおりに表示されます。		10.0	global
for	String	表示ラベルが関連付けられるコンポーネントの ID。表示ラベルにフォーカスがある場合、この属性によって指定されたコンポーネントもフォーカスされます。		10.0	global
label	String	ページの他のコンポーネントが表示ラベルコンポーネントを参照できるようにする識別子。		10.0	global
lang	String	「en」または「en-US」など、生成された HTML 出力の基本言語。この属性についての詳細は、 W3C仕様 を参照してください。		10.0	global
onblur	String	onblur イベントが発生した場合(フォーカスが表示ラベルから離れた場合)に呼び出される JavaScript。		10.0	global
onclick	String	onclick イベントが発生した場合(ユーザが表示ラベルをクリックした場合)に呼び出される JavaScript。		10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
dblclick	String	i ondblclick イベントが発生した場合(ユーザが表示ラベルをダブルクリックした場合)に呼び出されるJavaScript。	10.0	global	
focus	String	s onfocus イベントが発生した場合(表示ラベルにフォーカスがある場合)に呼び出されるJavaScript。	10.0	global	
keydown	String	y onkeydown イベントが発生した場合(ユーザがキーボードのキーを押した場合)に呼び出されるJavaScript。	10.0	global	
keypress	String	y onkeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出されるJavaScript。	10.0	global	
keyup	String	y onkeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出されるJavaScript。	10.0	global	
mousedown	String	s onmousedown イベントが発生した場合(ユーザがマウスボタンをクリックした場合)に呼び出されるJavaScript。	10.0	global	
mousemove	String	s onmousemove イベントが発生した場合(ユーザがマウスポインタを移動した場合)に呼び出されるJavaScript。	10.0	global	
mouseout	String	s onmouseout イベントが発生した場合(ユーザが表示ラベルからマウスポインタを移動した場合)に呼び出されるJavaScript。	10.0	global	
mouseover	String	s onmouseover イベントが発生した場合(ユーザが表示ラベルにマウスポインタを重ねた場合)に呼び出されるJavaScript。	10.0	global	
mouseup		onmouseup イベントが発生した場合(ユーザがマウスボタンを放した場合)	10.0	global	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
label	String	ルが選択される順序。この値は、ユーザがTabキーを押したときに選択される最初のコンポーネントを0として、0 ~ 32767の整数である必要があります。			
linkText	Object	ユーザがコンポーネントにマウスポインタを重ねたときにツールチップとして表示されるテキスト。	10.0	global	
labelText	Object	表示ラベルとして表示されるテキスト。	10.0	global	

apex:outputLink

URLへのリンク。このコンポーネントは、`h` 属性と共にアンカータグとしてHTMLに表示されます。HTMLと同様に、`u` `p` の本文は、`a` リンクとして表示されるテキストまたは画像です。クエリ文字列パラメータをリンクに追加するには、ネストされた `mp` `p` コンポーネントを使用します。

例

このコンポーネントでは、「html-」プレフィックスを使用した **HTML パススルー属性**がサポートされています。パススルー属性は、生成された `a` タグに適用されます。

例

上述の例では次のHTMLを表示します。

```
<a href="http://www.google.com">Google</a>
```

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
cc	String	リンクにフォーカスを置くキーボードのアクセスキー。リンクにフォーカスがあるときにEnterキーを押す操作は、リンクをクリックする操作と同じです。		10.0	global
encoding	String	指定URLの符号化に使用される文字セット。指定されていない場合、この値はデフォルトのISO-8859-1に設定されます。		10.0	global
shape	String	出力リンクに使用される画面上のホットスポットの位置と形状(クライアント側の画像マップ用)。カンマ区切り値の数および順序は定義される形状		10.0	global

属性名	属性型	説明	必須項目	API バージョン	アクセス
dr	String	に依存します。たとえば、長方形を定義するには、 coords="left-x, top-y, right-x, bottom-y" を使用します。円形を定義するには、 coords="center-x, center-y, radius" を使用します。多角形を定義するには、 coords="x1, y1, x2, y2, ..., xN, yN" を使用します。ここで、 x1 = xN および y1 = yN です。座標はピクセルまたはパーセントで表すことができます。また、座標は対応付けられる画像の左上からの距離を表します。「shape 属性」も参照してください。			
dbl d	Boolean	i 生成された HTML コンポーネントが読み取られる方向。使用可能な値には「RTL」(右から左) または「LTR」(左から右) があります。	10.0	global	
h g f	String	a このリンクを無効な状態で表示するかどうかを指定する boolean 値。true に設定した場合、HTML の span タグが通常のアンカータグの場所に使用されているため項目は無効な状態で表示されます。指定されていない場合、この値はデフォルトの false に設定されます。	10.0	global	
l g	String	「en」、「en-US」など、このコマンドリンクで参照されるリソースの基本言語。この属性についての詳細は、 W3C 仕様 を参照してください。	10.0	global	
onblur	String	i ページの他のコンポーネントが outputLink コンポーネントを参照できるようにする識別子。	10.0	global	
ol	String	a 「en」または「en-US」など、生成された HTML 出力の基本言語。この属性についての詳細は、 W3C 仕様 を参照してください。	10.0	global	
dbl l	String	c c i ondblclick イベントが発生した場合 (ユーザが出力リンクをダブルクリックした場合) に呼び出される JavaScript。	10.0	global	
o o u	String	f s onfocus イベントが発生した場合 (出力リンクにフォーカスがある場合) に呼び出される JavaScript。	10.0	global	
node o	String	k y onkeydown イベントが発生した場合 (ユーザがキーボードのキーを押した場合) に呼び出される JavaScript。	10.0	global	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
onkeypress	String	onkeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出される JavaScript。	10.0	global	
onkeyup	String	onkeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出される JavaScript。	10.0	global	
onmousedown	String	onmousedown イベントが発生した場合(ユーザがマウスボタンをクリックした場合)に呼び出される JavaScript。	10.0	global	
onmousemove	String	onmousemove イベントが発生した場合(ユーザがマウス pointer を移動した場合)に呼び出される JavaScript。	10.0	global	
onmouseout	String	onmouseout イベントが発生した場合(ユーザが出力リンクからマウス pointer を移動した場合)に呼び出される JavaScript。	10.0	global	
onmouseover	String	onmouseover イベントが発生した場合(ユーザが出力リンクにマウス pointer を重ねた場合)に呼び出される JavaScript。	10.0	global	
onmouseup	String	onmouseup イベントが発生した場合(ユーザがマウスボタンを放した場合)に呼び出される JavaScript。	10.0	global	
url	String	現在のドキュメントからこのコマンドリンクで指定される URLへのリレーション。この属性の値は、リンクタイプのスペース区切りのリストです。この属性についての詳細は、 W3C仕様 を参照してください。	10.0	global	
rendered	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	10.0	global	
reverse	String	現在のドキュメントへのこのコマンドリンクで指定される URLからの逆リンク。この属性の値は、リンクタイプのスペース区切りのリストです。この属性についての詳細は、 W3C仕様 を参照してください。	10.0	global	
shape	String	クライアント側の画像マップのホットスポットの形状。有効な値は、default、circle、rect、および poly です。「coords 属性」も参照してください。	10.0	global	
style	String	出力リンクコンポーネントの表示に使用されるスタイル。主に、インライン CSS スタイルを追加するために使用されます。	10.0	global	
styleClass	String	出力リンクコンポーネントの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを	10.0	global	

属性名	属性型	説明	必須項目	API バージョン	アクセス
使用するときに適用される CSS スタイルを指定するために使用されます。					
	String	i a	ユーザが Tab キーを繰り返し押したときに、他のページコンポーネントと比較して、このリンクが選択される順序。この値は、ユーザが Tab キーを押したときに選択される最初のコンポーネントを 0 として、0 ~ 32767 の整数である必要があります。	10.0	global
	String	a	このコマンドリンクが取得するリソースが表示されるフレームの名前。この属性に使用できる値には、「_blank」、「_parent」、「_self」、「_top」があります。また、目的の移行先の name 属性に値を割り当てることにより、独自のターゲット名を指定することもできます。	10.0	global
	String	i	ユーザがコンポーネントにマウスポインタを重ねたときにツールチップとして表示されるテキスト。	10.0	global
p	String	ity	この出力リンクによって指定されたリソースの MIME コンテンツタイプ。この属性の使用できる値には、「text/html」、「image/png」、「image/gif」、「video/mpeg」、「text/css」、および「audio/basic」があります。使用できる値の完全なリストなど、詳細は、W3C 仕様を参照してください。	10.0	global
	Object	a	出力リンクに使用される URL。	10.0	global

apex:outputPanel

グループ化された一連のコンテンツです。HTML `<div>` タグや `` タグを使用して、またはいずれのタグも使用せずに表示されます。`<div>` `` `` `<p>` `` タグを使用して、AJAX の更新に使用するコンポーネントをグループ化します。

このコンポーネントでは、「`html-`」プレフィックスを使用した **HTML パススルー属性**がサポートされています。`html-` パススルー属性は、`html-` 属性に応じて、生成されたコンテナタグである `div` または `span` に適用されます。

Span の例

上述の例では次の HTML を表示します。

Div の例

```
<div style="text-align: right; border: 1px solid black; padding: 5px; margin-bottom: 10px;">
    <h1>Hello World!</h1>
    <p>This is a sample output panel. You can use it to output any content you want.</p>

```

上述の例では次の HTML を表示します。

```
<div style="text-align: right; border: 1px solid black; padding: 5px; margin-bottom: 10px;">
    <h1>Hello World!</h1>
    <p>This is a sample output panel. You can use it to output any content you want.</p>

```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
dir	String	i 生成された HTML コンポーネントの読み取り方向。使用可能な値には「RTL」(右から左) または「LTR」(左から右) があります。		10.0	global
label	String	i ページの他のコンポーネントが outputPanel コンポーネントを参照できるようにする識別子。		10.0	global
lang	String	a 「en」または「en-US」など、生成された HTML 出力の基本言語。この属性についての詳細は、 W3C 仕様 を参照してください。		10.0	global
layout	String	ya パネルのレイアウトスタイル。使用できる値には、「block」(HTML div タグを生成)、「inline」(HTML span タグを生成)、および「none」(HTML タグを生成しない) があります。指定されていない場合、この値はデフォルトの「none」に設定されます。ただし、表示される属性が「false」に設定されたそれぞれの子要素についてレイアウトが「none」に設定されている場合、outputPanel によって、それぞれの子の ID と「display:none」に設定された style 属性を持つ span タグが生成されます。したがって、コンテンツが表示されない間も、JavaScript は DOM ID によって要素にアクセスできます。		10.0	global
onclick	String	i onclick イベントが発生した場合(ユーザが出力パネルをクリックした場合)に呼び出される JavaScript。		10.0	global
ondblclick	String	i ondblclick イベントが発生した場合(ユーザが出力パネルをダブルクリックした場合)に呼び出される JavaScript。		10.0	global
onkeydown	String	yonkeydown イベントが発生した場合(ユーザがキーボードのキーを押した場合)に呼び出される JavaScript。		10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
onkeypress	String	onkeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出される JavaScript。	10.0	global	
onkeyup	String	onkeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出される JavaScript。	10.0	global	
onmousedown	String	onmousedown イベントが発生した場合(ユーザがマウスボタンをクリックした場合)に呼び出される JavaScript。	10.0	global	
onmousemove	String	onmousemove イベントが発生した場合(ユーザがマウスポインタを移動した場合)に呼び出される JavaScript。	10.0	global	
onmouseout	String	onmouseout イベントが発生した場合(ユーザが出力パネルからマウスポインタを移動した場合)に呼び出される JavaScript。	10.0	global	
onmouseover	String	onmouseover イベントが発生した場合(ユーザが出力パネルにマウスポインタを重ねた場合)に呼び出される JavaScript。	10.0	global	
onmouseup	String	onmouseup イベントが発生した場合(ユーザがマウスボタンを放した場合)に呼び出される JavaScript。	10.0	global	
rendered	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	10.0	global	
style	String	s outputPanel コンポーネントの表示に使用されるスタイル。主に、インライン CSS スタイルを追加するために使用されます。	10.0	global	
styleClass	String	c outputPanel コンポーネントの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。	10.0	global	
title	String	i ユーザがコンポーネントにマウスポインタを重ねたときにツールチップとして表示されるテキスト。	10.0	global	

apex:outputText

Visualforce ページにテキストを表示します。CSS スタイルを使用して `u` `p` の表示をカスタマイズできます。この場合、生成されるテキストは、HTML `p` タグに囲まれます。また、特殊な HTML および XML 文字が含まれる場合、表示されるテキストをエスケープできます。このコンポーネントは、ローカライズを考慮します。

ネストされた param タグを使用してテキスト値の書式を設定します。{n} は param タグのネストの深さを示します。属性では、Java の MessageFormat クラスと同じ構文がサポートされています。詳細は、MessageFormat クラスの JavaDocs を参照してください。

警告: ユーザーに埋め込まれた暗号化カスタム項目は、プレーンテキストで表示されます。コンポーネントは、ユーザの「暗号化されたデータの参照」権限を考慮しません。機密情報を未承認のユーザに表示しないようにするには、代わりに `u_lph p <タグを使用してください。`

このコンポーネントでは、「html-」プレフィックスを使用した HTML パススルー属性がサポートされています。パススルー属性は、生成されたコンテナタグに適用されます。

基本的な書式設定の例

```
g p p < a >
x e r lmpu p m < a va a " " y > /
x e re lmpun ip ug a va a " " a s' > /
x et tx ou u p p < T /a >
g p p < /a >
```

上述の例では次の HTML を表示します。

```
pt etml t <T s l i aF1 e t tx sw cy h m tS -nyr i gmp i si s y ip a <s . s
```

日付形式の設定の例

```
g p p < a >
x e r lmpu p p < T /a >
g p p < /a >
```

上述の例では次の HTML を表示します。

```
tt et u ro h tm T fmr m g o h : a i 2 2 : i 1M / 09 00 4 P
br et ro bd fmr m g o h t A:a 2i i 2 i 2 : 11 200 M 0T 9 a 4 0 G
```

通貨形式の設定の例

```
ex ro te e lne m d r pr Eel i se opetan o e ec lu yoy s sscá V pa is a
et lnd h etrn cd i cv á ha a i U L R .
ro l en ept f F t n eM iqulh t 00 000000 h , s: U L R s
e he n erap xe fc sc ne bsga i s a /a I kM al 00 000000
e ee el hro e l fr V t Sdt itpa B escuro l sntqloTm info maf i a i ia .
```

上述の例では次の HTML を表示します。

```
ro I h : i s 500 000 0.00 00
```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
direction	String	i 生成された HTML コンポーネントが読み取られる方向。使用可能な値には「RTL」(右から左) または「LTR」(左から右) があります。		10.0	global
escape	Boolean	s このコンポーネントが生成する HTML 出力で、特殊な HTML および XML 文字をエスケープかどうかを指定する boolean 値。escape="false" が指定されていない場合、文字エスケープシーケンスは記述されたとおりに表示されます。この値を「false」に設定すると、悪質な方法で使用されるおそれのある JavaScript などの任意のコンテンツが許可されるため、セキュリティのリスクとなる可能性があります。		10.0	global
label	String	i ページの他のコンポーネントが outputText コンポーネントを参照できるようにする識別子。		10.0	global
language	String	a 出力テキストの横に表示ラベルを表示するためのテキスト値。		23.0	
renderdirect	Boolean	a 「en」または「en-US」など、生成された HTML 出力の基本言語。この属性についての詳細は、 W3C 仕様 を参照してください。		10.0	global
style	String	s outputText コンポーネントの表示に使用されるスタイル。主に、オンライン CSS スタイルを追加するために使用されます。		10.0	global
styleClass	String	s outputText コンポーネントの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。		10.0	global
title	String	i ユーザがコンポーネントにマウスポインタを重ねたときにツールチップとして表示されるテキスト。		10.0	global
value	Object	a このコンポーネントと共に表示されるテキスト。この値は、Java の MessageFormat クラスと同じ構文をサポートしています。MessageFormat クラスについての詳細は、		10.0	global

属性名	属性型	説明	必須項目	API バージョン	アクセス
http://java.sun.com/j2se/1.4.2/docs/api/java/text/MessageFormat.html を参照してください。					

apex:page

単一の Visualforce ページです。すべてのページは、単一の page コンポーネントタグ内でラップされている必要があります。

このコンポーネントでは、「html-」プレフィックスを使用した [HTML パススルー属性](#)がサポートされています。パススルー属性は、生成された `<h>` タグに適用されます。>

例

```
example:tn{tn t e
  n l t b do o or nh u C o wn dnp T g p p : <!-- Pa -->
  r gA p pd amnr sf y> det} y c i Ma y u ' a ; s ' s / y >
  g p p < /a >
  <1 h > ai <a b >
  < >i s i p s y D < F P >/
```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
c	ApexPages.Action	このページをサーバが要求したときに呼び出される action メソッド。式の言語を使用して action メソッドを参照します。たとえば、action="![doAction]" はコントローラの doAction() メソッドを参照します。アクションが指定されていない場合、ページは通常どおり読み込まれます。action メソッドが null を返す場合、ページは単に更新されます。このメソッドは、ページが表示される前にコールされるため、ユーザを別のページにリダイレクトすることもできます。このアクションは、初期化には使用できません。		10.0	global
o	p	ページの表示と実行に使用される API のバージョン。		10.0	global
do	Boolean	生成される HTML 出力に、 <code><apex:page></code> タグを Visualforce で自動的に追加するかどうかを指定する boolean 値。たとえば、 <code><apex:page></code>		27.0	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
l l m _app	T H Boolean	<p>をマークアップで静的に設定する場合などのように、応答への <code>h</code> タグの追加を無効にするには、<code>false</code> に設定します。指定されていない場合、この値は <code>applyHtmlTag</code> 属性の値がデフォルトで適用されます。設定されているか <code>true</code> の場合、<code>applyHtmlTag</code> の値は適用されません。</p> <p>a t</p> <p>生成される HTML 出力に <code>h</code> タグを Visualforce で自動的に追加するかどうかを指定する boolean 値。たとえば、<code>h</code> タグをマークアップで静的に設定する場合などのように、応答への <code>h</code> タグの追加を無効にするには、<code>false</code> に設定します。指定されていない場合、この値はデフォルトの <code>true</code> に設定されます。</p>	< 27.0	>	
c ch	Boolean	ブラウザがこのページをキャッシュするかどうかを指定する boolean 値。 <code>true</code> に設定すると、ブラウザはページをキャッシュします。指定されていない場合、この値はデフォルトの <code>false</code> に設定されます。Force.com サイトのページについては、この属性が指定されていない場合、この値はデフォルトの <code>true</code> に設定されます。サイトページのキャッシュについての詳細は、Salesforce オンラインヘルプの「Force.com サイトページのキャッシュ」を参照してください。	10.0	global	
o c ct	String	<p>表示されるページのフォーマットに使用される MIME コンテンツタイプ。この属性の使用できる値には、"text/html"、"image/png"、"image/gif"、"video/mpeg"、"text/css"、および "audio/basic" があります。使用できる値の完全なリストなど、詳細は、W3C 仕様を参照してください。</p> <p>表示されるページのファイル名は、MIME タイプに "#" とファイル名を順に追加して定義することもできます。たとえば、「application/vnd.ms-excel#contacts.xls」とすることができます。注意: 一部のブラウザでは、ファイル名を指定してページのキャッシュ属性を "true" に設定しない限り、そのファイルを開くことができません。</p>	10.0	global	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
standardController	String	このページの動作を制御するために使用する、Apexで記述されたカスタムコントローラクラスの名前。standardController属性が存在する場合、この属性は指定できません。		10.0	global
defaultCommand	Boolean	ページの準備ができる前にコマンドボタンおよびリンクをクリックしないようするかどうかを指定する boolean 値。true に設定すると、ボタンまたはリンクの最後のクリックがエンキューされ、ページの準備ができたときに処理されます。この値のデフォルトは false です。		26.0	
contentType	String	表示されるページの構造を記述する HTML ドックタイプ定義(DTD)、つまり doctype。指定されていない場合、この値はデフォルトの "html-4.01-transitional" に設定されるため、doctype は		23.0	
		B C E T M T D I C ! O M L R U L P C M T -B/ H / L 1 A 0 o l T E \$ à aN/ / t h e r a p t g l l m o o h d T // 3 . R / / 4 \$. " >			
		になります。この属性に使用できる値には、主に "html401-strict"、"xhtml10-transitional"、"xhtml11-basic"、"html50" があります。			
		HTML の doctype 声明についての詳細は、W3C 仕様を参照してください。			
cacheTime	Integer	キャッシュ属性の有効期限(秒)。キャッシュ属性が true に設定されており、属性が指定されていない場合、この値はデフォルトの 0 に設定されます。Force.com サイトのページについては、キャッシュが false に設定されていない場合、この値はデフォルトの 600 秒に設定されます。サイトページのキャッシュについての詳細は、Salesforce オンラインヘルプの「Force.com サイトページのキャッシュ」を参照してください。		14.0	
componentName	String	このページにロジックを追加する、Apexで記述された 1 つ以上のカスタムコントローラ拡張の名前。		11.0	global
id	String	ページの他のコンポーネントが参照できるようにするページの識別子。		10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
label	String	Salesforce 設定ツールでページを参照するために使用される表示ラベル。	10.0	global	
language	String	関連付けられている翻訳が Salesforce にあるラベルの表示に使用される言語。この値は、ページを表示しているユーザの言語より優先されます。この属性に使用できる値には、"en"、"en-US"などの Salesforce でサポートされている言語の言語キーがあります。	10.0	global	
mf	String	生成される <code>h</code> タグに、オフライン使用でキャッシュマニフェストファイルを参照するマニフェスト属性を追加します。マニフェスト属性を設定するには、 <code>docType="html-5.0"</code> を設定し、 <code>applyHtmlTag</code> を "false" に設定しない必要があります。	>	27.0	
name	String	Force.com API でページを参照するために使用される一意の名前。	10.0	global	
pageStyle	String	pageStyle 属性は、Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。	10.0	global	
readOnly	Boolean	Visualforce ページで参照のみモードを有効化する boolean 値。参照のみモードの場合、ページの DML 操作は実行されませんが、取得されるレコード数の制限は、50,000 行から 1,000,000 行に増加します。反復コンポーネントによって処理されるコレクション内の項目数も、1,000 から 10,000 に増加します。指定されていない場合、この値はデフォルトの false に設定されます。		23.0	
recordName	String	recordSetName 属性は、Salesforce API バージョン 16.0 では使用できなくなりました。ページへの影響はありません。代わりに recordSetVar を使用してください。	a	14.0	
recordSet	String	この属性は、ページでセット指向の標準コントローラが使用されることを示します。属性の値は、ページに渡されるレコードのセットの名前を示します。このレコードセットを式で使用し、ページでの表示に使用する値を返したり、レコードのセットに対してアクションを実行したりできます。たとえば、ページで標準取引先コントローラが使用されており、recordSetVar が "accounts"	a	14.0	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
に設定されている場合、次のコードの実行によって単純な pageBlockTable という取引先レコードを作成できます。					
content	String	<p>サポートされるコンテンツコンバータの名前。現在、サポートされているコンテンツコンバータは PDF のみです。この属性を "pdf" に設定すると、ページが PDF として表示されます。</p> <p>Visualforce ページを PDF として表示する機能は、印刷用にデザインされ、最適化されたページのためのものです。印刷用の書式設定が容易ではないか、入力やボタンなどのフォーム要素が含まれる標準コンポーネント、および書式設定に JavaScript が必要なコンポーネントは使用しないでください。これにはフォーム要素を必要とするコンポーネントなどが含まれますが、これに限定されません。リリースする前に、表示されるページのフォーマットを確認してください。</p> <p>PDF すべての文字が表示されない場合は、CSS のフォントを調整して要件に対応するフォントを使用します。たとえば、次のスタイル定義をページのスタイルに追加します。</p> <pre>{ t b d o o l a n f n r f l : d o y c i a y j u ' a } M ; s '</pre> <p>pageBlock と sectionHeader コンポーネントは、PDF に表示されるとき、ダブルバイトのフォントをサポートしません。</p>	13.0	global	
display	Boolean	ページを表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	10.0	global	
style	Boolean	ページが標準の Salesforce 設定ページのスタイルを使用するかどうかを指定する boolean 値。true の場合、設定のスタイルが使用さ	10.0	global	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
Chatterヘッダー					
chatterHeader	Boolean	Chatter Messenger チャットウィジェットをページに含めるかどうかを指定する boolean 値。true の場合、チャットウィジェットが表示されます。指定されていない場合、値は、[設定] の [カスタマイズ] [Chatter] [チャットの設定] で選択された Visualforce 設定の値がデフォルトで適用されます。	10.0	global	
header	Boolean	ページに Salesforce タブのヘッダーを含めるかどうかを指定する boolean 値。true の場合、タブのヘッダーは表示されます。指定されていない場合、この値はデフォルトの true に設定されます。	10.0	global	
sidebar	Boolean	標準の Salesforce サイドバーをページに含めるかどうかを指定する boolean 値。true の場合、サイドバーは表示されます。指定されていない場合、この値はデフォルトの true に設定されます。	10.0	global	
control	String	このページの動作を制御するために使用する Salesforce オブジェクトの名前。この属性は、コントローラ属性も存在する場合は指定できません。	10.0	global	
header	Boolean	showHeader 属性が false に設定されている場合に、標準の Salesforce スタイルシートを生成されたページのヘッダーに追加するかどうかを指定する boolean 値。true に設定すると、生成されるページのヘッダーに標準のスタイルシートが追加されます。指定されていない場合、この値はデフォルトの true に設定されます。この設定を false にすると、Salesforce.com の CSS を必要とするコンポーネントは正しく表示されず、リリースごとに異なるスタイルが適用される場合があります。	11.0	global	
body	String	このページの色、スタイル、および選択されたタブを制御する Salesforce オブジェクトまたはカスタム Visualforce タブ。カスタムオブジェクトを使用している場合、属性にオブジェクトの開発者名を指定する必要があります。たとえば、MyCustomObject に関連付けられたスタイルを使用するには、	10.0	global	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
tabStyle	String	tabStyle="MyCustomObject__c" を使用します。標準コントローラが指定されている場合、デフォルトで、関連付けられたコントローラのスタイルが使用されます。カスタムコントローラが定義されている場合は、デフォルトで [ホーム] タブに設定されます(カスタムコントローラの場合)。カスタムの Visualforce タブを使用するには、属性をタブ名(表示ラベルではない) + アンダースコア 2 個(_) + 単語「tab」に設定します。たとえば、Source という名前と Sources という表示ラベルが設定された Visualforce タブのスタイルを使用するには、tabStyle="Source_tab" を使用します。			
title	String	ブラウザに表示されるページのタイトル。開発者モードでページを編集している場合、ページタイトルは表示されません。	10.0	global	
isStandard	Boolean	ページが標準の Salesforce ウィザードページのスタイルを使用するかどうかを指定する boolean 値。true の場合、ウィザードのスタイルが使用されます。指定されていない場合、この値はデフォルトの false に設定されます。	10.0	global	

apex:pageBlock

Salesforce の詳細ページの外観に似た、デフォルトのコンテンツを使用しないスタイルを使用するページの領域です。

このコンポーネントでは、「html-」プレフィックスを使用した [HTML パススルー属性](#)がサポートされています。パススルー属性は、生成されたコントナタグ [に適用されます](#)。

例

```

<apex:pageBlock>
    <div style="background-color: #f0f0f0; padding: 10px; border: 1px solid #ccc; margin-bottom: 10px;">
        <div style="border: 1px solid #ccc; padding: 5px; display: inline-block;">
            <span>Standard Page Block</span>
        </div>
    </div>
    <div style="border: 1px solid #ccc; padding: 5px; background-color: #fff; margin-bottom: 10px;">
        <div style="border: 1px solid #ccc; padding: 5px; display: inline-block;">
            <span>Custom Page Block</span>
        </div>
    </div>
    <div style="border: 1px solid #ccc; padding: 5px; background-color: #fff; margin-bottom: 10px;">
        <div style="border: 1px solid #ccc; padding: 5px; display: inline-block;">
            <span>Custom Page Block</span>
        </div>
    </div>
</apex:pageBlock>

```

e	e	x	t ₁₁	oBndmm p ₁₀ o	θ	ʃ	ɛ	a	l	u	a	i	a	="	ks	h	a" v	s a >	/
e	ex		t ₁₂	lbgo p u p c <	k/a	a						s >							
e	ex	e	t ₁₃	legn p pə t let ₁₄	nt k ə	n M	C	əs	lo <u>ə</u> u	əl c	y	2	is	"			s=" "	>	
e	x	et n	e { u ldp lt ep n	n< ou	á cc m̩ v	á'	!	a	.	'a	>	/							
e	x	et n	e { u ldp lt ep n	< ou	á cc E v	á'	!	a	.	's	"	>	/						
e	x	et n	e { u ldp lt ep n	< ou	á cc E v	p	á'	!	a	.	y"	>	/						
e	x	et n	e { u ldp lt p n	teom	á cou b wcm	á'	!	a	.	a N	"	>	/						
e	ex	e	t ₁₅	l g ₁₆ p p o c <c	k/a	a													
ex		B	l go p p c <	k/a	a >														
x		ro	m p f <	/a	>														
		g	p p <	/a	>														

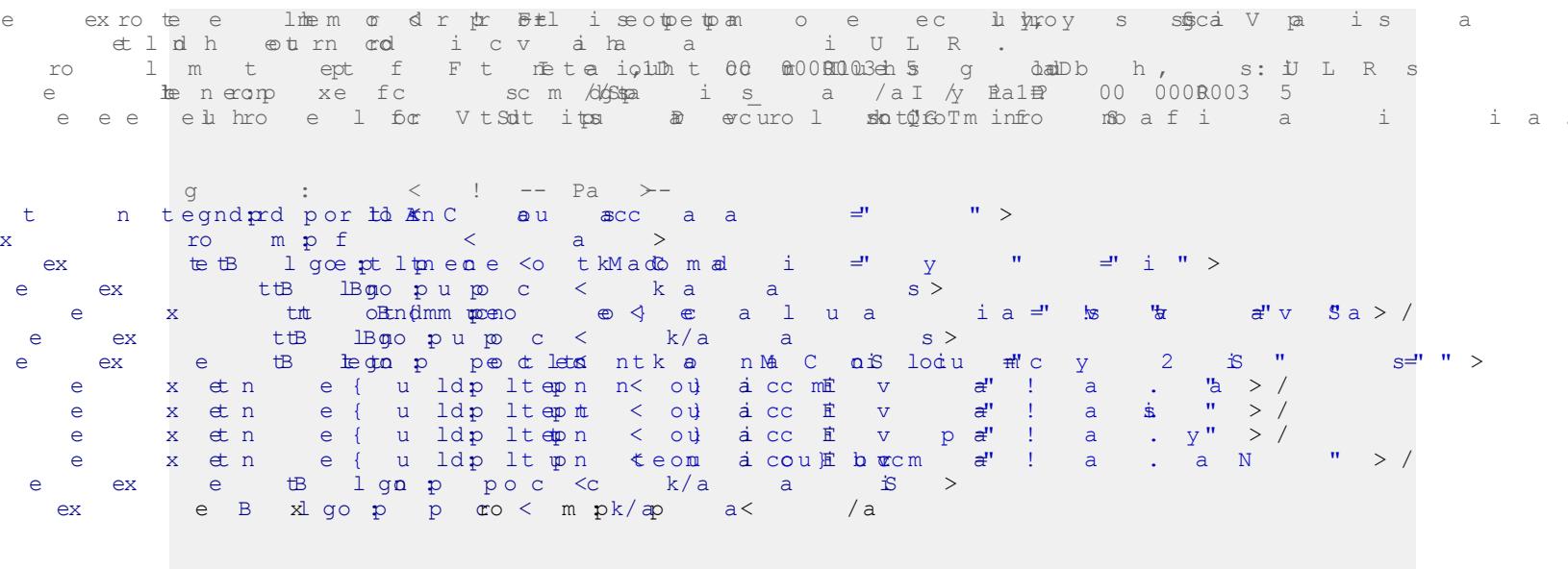
属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
enableinlineediting	Boolean	E - 詳細モードと同じようにデータが表示されますが、サポートされる子コンポーネントはオンライン編集用に有効化されます。			
event	String	表示される線は必須要件には関係なく、視覚的な分離線でしかありません。詳細ページを見やすくするのが目的です。指定されていない場合、この属性はデフォルトの i に設定されます。			
onclick	String	onclick イベントが発生した場合(ユーザがページブロックをクリックした場合)に呼び出される JavaScript。		10.0	global
ondblclick	String	ondblclick イベントが発生した場合(ユーザがページブロックをダブルクリックした場合)に呼び出される JavaScript。		10.0	global
onkeydown	String	onkeydown イベントが発生した場合(ユーザがキーボードのキーを押した場合)に呼び出される JavaScript。		10.0	global
onkeypress	String	onkeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出される JavaScript。		10.0	global
onkeyup	String	onkeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出される JavaScript。		10.0	global
onmousedown	String	onmousedown イベントが発生した場合(ユーザがマウスボタンをクリックした場合)に呼び出される JavaScript。		10.0	global
onmousemove	String	onmousemove イベントが発生した場合(ユーザがマウスポインタを移動した場合)に呼び出される JavaScript。		10.0	global
onmouseout	String	onmouseout イベントが発生した場合(ユーザがページブロックからマウスポインタを移動した場合)に呼び出される JavaScript。		10.0	global
onmouseover	String	onmouseover イベントが発生した場合(ユーザがページブロックにマウスポインタを重ねた場合)に呼び出される JavaScript。		10.0	global
onmouseup	String	onmouseup イベントが発生した場合(ユーザがマウスボタンを放した場合)に呼び出される JavaScript。		10.0	global
renderd	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		10.0	global

このコンポーネントでは、「html-」プレフィックスを使用した **HTML パススルー属性**がサポートされています。パススルー属性は、`< t k>` ボタンを含む生成された `< t k>` タグに適用されます。この `< t k>` タグは、`< >`

`tB lBgo p u p c < t k>` コンポーネントの `case` 属性に応じて、`l gá p a p c` の先頭、`k 最後、ma` またはその両方に配置できます。

例



属性名	属性型	説明	必須項目	APIバージョン	アクセス	
ol	c c	String	i	onclick イベントが発生した場合(ユーザが pageBlockButtons コンポーネントの任意の場所をクリックした場合)に呼び出される JavaScript。	11.0	global
dbl l	c c	String	i	ondblclick イベントが発生した場合(ユーザが pageBlockButtons コンポーネントをダブルクリックした場合)に呼び出される JavaScript。	11.0	global
no d o	d o	String	k	onkeydown イベントが発生した場合(ユーザがキーボードのキーを押した場合)に呼び出される JavaScript。	11.0	global
o r	r	String	p k	onkeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出される JavaScript。	11.0	global
o u	u	String	p k	onkeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出される JavaScript。	11.0	global
w on o und o	on o und o	String	s	onmousedown イベントが発生した場合(ユーザがマウスボタンをクリックした場合)に呼び出される JavaScript。	11.0	global
o o um o m	o m	String	s	onmousemove イベントが発生した場合(ユーザがマウス pointer を移動した場合)に呼び出される JavaScript。	11.0	global
o o um ou	ou	String	s	onmouseout イベントが発生した場合(ユーザが pageBlockButtons コンポーネントからマウス pointer を移動した場合)に呼び出される JavaScript。	11.0	global
o o um or	or	String	s	onmouseover イベントが発生した場合(ユーザが pageBlockButtons コンポーネントにマウス pointer を重ねた場合)に呼び出される JavaScript。	11.0	global
o o um u	u	String	p	onmouseup イベントが発生した場合(ユーザがマウスボタンを放した場合)に呼び出される JavaScript。	11.0	global
r d r d		Boolean		コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	11.0	global
l		String	s	pageBlockButtons コンポーネントの表示に使用されるスタイル。インライン CSS スタイルの追加に主に使用されます。	11.0	global
l l	C	String	s	pageBlockButtons コンポーネントの表示に使用されるスタイルクラス。外部 CSS スタイルシートを使用するときに適用される CSS スタイルの指定に主に使用されます。	11.0	global

属性名	属性型	説明	必須項目	API バージョン	アクセス
String	i	ユーザがコンポーネントにマウスポインタを重ねたときにツールチップとして表示されるテキスト。	11.0	global	

apex:pageBlockSection

標準の Salesforce ページレイアウト定義内のセクションに類似した、コンポーネント内のデータのセクションです。

`1 g a p p o c < k`コンポーネントは、`1`つ以上の列で構成されており、各列には、項目の表示ラベルとその値の2つのセルがあります。`1 g a p p o c < k`の本文に含まれる各コンポーネントは、列数に達するまで、行の次のセルに配置されます。列数に達したら、その次のコンポーネントは次の行の最初のセルに配置されます。

Salesforce オブジェクトから `lga_p poc & e k` に項目を追加するには、`u ldp p` または `a ou u ldp p` <コンポーネントを使用します。各コンポーネントは、項目の関連付けられた表示ラベルと共に自動的に表示されます。Salesforce オブジェクトの項目に基づかない変数またはメソッドの項目を追加する、または Salesforce オブジェクト項目の表示ラベルのフォーマットをカスタマイズするには、`be t gop poc & m ka` コンポーネントを使用します。`u xltd p o & u ldp p <` または `t gop poc & m ka` の各コンポーネントは、1つの列の2つのセルにまたがって配置されます。

このコンポーネントでは、「`html-`」プレフィックスを使用した **HTML パスルーアтриビュート**がサポートされています。パスルーアтриビュートは、生成されたコンテナタグに適用されます。

例

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
label	Boolean	ユーザがページロックセクションを展開したり折りたたんだりできるかどうかを指定する boolean 値。 true の場合、ユーザはセクションの展開および折りたたみを実行できます。指定されていない場合、この値はデフォルトの true に設定されます。		11.0	global
location	Integer	ページロックセクションの单一の行に含めることができる列の数。 単一の列は 2 つのセル(項目の表示ラベルおよびその値)にまたがります。 pageBlockSection に子 inputField、子 outputField、または子 pageBlockSectionItem コンポーネントを使用する場合、それぞれの子コンポーネントは、1 つの列の 2 つのセルにまたがって表示されます。 pageBlockSection に他のコンポーネントを使用する場合は、列の一番右のセルにのみ表示され、一番左のセルは空のままになります。 pageBlockSection には 1 つ以上の列を指定できますが、Salesforce スタイルシートは 1 つまたは 2 つの列に対して最適化されています。 指定されていない場合、この値はデフォルトの 2 に設定されます。		11.0	global
direction	String	i 生成された HTML コンポーネントの読み取り方向。 使用可能な値には「RTL」(右から左) または「LTR」(左から右) があります。		10.0	global
dynamic	String	i ページの他のコンポーネントが pageBlockSection コンポーネントを参照できるようにする識別子。		10.0	global
language	String	a 「en」または「en-US」など、生成された HTML 出力の基本言語。 この属性についての詳細は、 W3C仕様 を参照してください。		10.0	global
onClick	String	i onclick イベントが発生した場合(ユーザがページロックセクションをクリックした場合)に呼び出される JavaScript。		10.0	global
onDoubleClick	String	i ondblclick イベントが発生した場合(ユーザがページロックセクションをダブルクリックした場合)に呼び出される JavaScript。		10.0	global
onKeyDown	String	y onkeydown イベントが発生した場合(ユーザがキーボードのキーを押した場合)に呼び出される JavaScript。		10.0	global
onKeyPress	String	y onkeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出される JavaScript。		10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
onkeyup	String	onkeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出される JavaScript。	10.0	global	
onmousedown	String	onmousedown イベントが発生した場合(ユーザがマウスボタンをクリックした場合)に呼び出される JavaScript。	10.0	global	
onmousemove	String	onmousemove イベントが発生した場合(ユーザがマウスポインタを移動した場合)に呼び出される JavaScript。	10.0	global	
onmouseout	String	onmouseout イベントが発生した場合(ユーザがページロックセクションからマウスポインタを移動した場合)に呼び出される JavaScript。	10.0	global	
onmouseover	String	onmouseover イベントが発生した場合(ユーザがページロックセクションにマウスポインタを重ねた場合)に呼び出される JavaScript。	10.0	global	
onmouseup	String	onmouseup イベントが発生した場合(ユーザがマウスボタンを放した場合)に呼び出される JavaScript。	10.0	global	
display	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	10.0	global	
displayHeader	Boolean	ページロックセクションのタイトルを表示するかどうかを指定する boolean 値。true に設定すると、ヘッダーが表示されます。指定されていない場合、この値はデフォルトの true に設定されます。	11.0	global	
title	String	ページロックセクションのタイトルとして表示されるテキスト。	10.0	global	

Facet

facet名	説明	APIバージョン
body	ページロックセクションの本文に表示されるコンポーネント。指定されている場合、この facet のコンテンツが、pageBlockSection タグの本文より優先されます。セクション本文の表示は、name="body" を含む facet によって制御されるため、body facet がページロックセクションコンポーネントの本文に表示される順序は重要ではありません。	11.0
header	ページロックセクションのタイトルに表示されるコンポーネント。指定されている場合、この facet のコンテンツが、タイトル属性の値より優先されます。セクションタイトルの表示は、name="header" を含む facet によって制御されるため、header facet がページロックセクションコンポーネントの本文に表示される順序は重要ではありません。	10.0

apex:pageBlockSectionItem

1つの行の1つの列を占有するコンポーネントの单一データです。コンポーネントには、最大2つの子コンポーネントを含めることができます。コンテンツが指定されていない場合、列は空のスペースとして表示されます。1つの子コンポーネントが指定されている場合、そのコンテンツは列の2つのセルにまたがって表示されます。2つの子コンポーネントが指定されている場合、最初のコンテンツは列の「表示ラベル」セル(左側のセル)に表示され、2つ目のコンテンツは列の「データ」セル(右側のセル)に表示されます。

これらのコンポーネントが子である場合とは異なり、表示ラベルまたはカスタムヘルプテキストと共に表示されません。また、コンポーネントは再表示されませんが、代わりに子コンポーネントが再表示されます。

このコンポーネントでは、「html-」プレフィックスを使用した HTML パススルー属性がサポートされています。パススルー属性は、生成されたコンテナタグに適用されます。

例

```

<apex:pageBlockSectionItem id="sec1" title="Section 1" type="Section" style="background-color: #f0f0f0; border: 1px solid black; padding: 10px; margin-bottom: 10px;">
    <apex:pageBlockSectionItem id="sec1_1" title="Section 1.1" type="Section" style="background-color: #e0e0e0; border: 1px solid black; padding: 5px; margin-bottom: 5px;">
        <apex:pageBlockSectionItem id="sec1_1_1" title="Section 1.1.1" type="Section" style="background-color: #d0d0d0; border: 1px solid black; padding: 2px; margin-bottom: 2px;">
            Content 1.1.1
        </apex:pageBlockSectionItem>
        <apex:pageBlockSectionItem id="sec1_1_2" title="Section 1.1.2" type="Section" style="background-color: #d0d0d0; border: 1px solid black; padding: 2px; margin-bottom: 2px;">
            Content 1.1.2
        </apex:pageBlockSectionItem>
    </apex:pageBlockSectionItem>
    <apex:pageBlockSectionItem id="sec1_2" title="Section 1.2" type="Section" style="background-color: #e0e0e0; border: 1px solid black; padding: 5px; margin-bottom: 5px;">
        <apex:pageBlockSectionItem id="sec1_2_1" title="Section 1.2.1" type="Section" style="background-color: #d0d0d0; border: 1px solid black; padding: 2px; margin-bottom: 2px;">
            Content 1.2.1
        </apex:pageBlockSectionItem>
        <apex:pageBlockSectionItem id="sec1_2_2" title="Section 1.2.2" type="Section" style="background-color: #d0d0d0; border: 1px solid black; padding: 2px; margin-bottom: 2px;">
            Content 1.2.2
        </apex:pageBlockSectionItem>
    </apex:pageBlockSectionItem>

```

属性

	属性名	属性型	説明	必須項目	APIバージョン	アクセス
d	l	String	a pageBlockSection 列の右側の「データ」セルのコンテンツを表示するために使用される CSS スタイル。		11.0	global
d	l l	C String	a pageBlockSection 列の右側の「データ」セルのコンテンツを表示するために使用される CSS スタイルクラス。		11.0	global
d	l	T String	i a pageBlockSection 列の右側の「データ」セルにマウスオーバー時に表示されるテキスト。		11.0	global
d r		String i	生成された HTML コンポーネントが読み取られる方向。使用可能な値には「RTL」(右から左) または「LTR」(左から右) があります。		11.0	global
1	h p T	String	マウスオーバー時にフロート表示されるツールチップとして、この項目の横に表示されるヘルプテキスト。設定で項目にカスタムヘルプが定義されている場合に標準 Salesforce 項目の横に表示されるテキストに類似しています。このヘルプテキストは、親ページの showHeader 属性が true に設定されている場合にのみ表示されます。		12.0	global
d		String i	ページの他のコンポーネントが pageBlockSectionItem コンポーネントを参照できるようにする識別子。		11.0	global
l b l	l	String	a pageBlockSection 列の左側の「表示ラベル」セルのコンテンツを表示するために使用される CSS スタイル。		11.0	global
l b l	l	C String	a pageBlockSection 列の左側の「表示ラベル」セルのコンテンツを表示するために使用される CSS スタイルクラス。		11.0	global
l b l	l	T String	i a pageBlockSection 列の左側の「表示ラベル」セルにマウスオーバー時に表示されるテキスト。		11.0	global
1 g		String	a 「en」または「en-US」など、生成された HTML 出力の基本言語。この属性についての詳細は、 W3C 仕様 を参照してください。		11.0	global
o l	c c	Stringdk	a onDataclick イベントが発生した場合(ユーザが pageBlockSection 列の右側の「データ」セルをクリックした場合)に呼び出される JavaScript。		11.0	global
o dbl	l	c c StringD k	a onDatadblclick イベントが発生した場合(ユーザが pageBlockSection 列の右側の「データ」セルをダブルクリックした場合)に呼び出される JavaScript。		11.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
wo_n_d_o	String	onDatakeydown イベントが発生した場合(ユーザがキーボードのキーを押した場合)に呼び出される JavaScript。	11.0	global	
o_r	String	onDatakeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出される JavaScript。	11.0	global	
o_u	String	onDatakeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出される JavaScript。	11.0	global	
wo_n_owm_o	String	onDatamousedown イベントが発生した場合(ユーザがマウスボタンをクリックした場合)に呼び出される JavaScript。	11.0	global	
o_ou_m_o_m	String	onDatamousemove イベントが発生した場合(ユーザがマウスポインタを移動した場合)に呼び出される JavaScript。	11.0	global	
t_o_ou_mou	String	onDatamouseout イベントが発生した場合(ユーザが pageBlockSection 列の右側の「データ」セルからマウスポインタを移動した場合)に呼び出される JavaScript。	11.0	global	
o_oumor	String	onDatamouseover イベントが発生した場合(ユーザが pageBlockSection 列の右側の「データ」セルにマウスポインタを重ねた場合)に呼び出される JavaScript。	11.0	global	
o_oumu	String	onDatamouseup イベントが発生した場合(ユーザがマウスボタンを放した場合)に呼び出される JavaScript。	11.0	global	
o_b_l_l_c_c	String	onLabelclick イベントが発生した場合(ユーザが pageBlockSection 列の左側の「表示ラベル」セルをクリックした場合)に呼び出される JavaScript。	11.0	global	
o_b_ldbl_l_c_c	String	onLabeldblclick イベントが発生した場合(ユーザが pageBlockSection 列の左側の「表示ラベル」セルをダブルクリックした場合)に呼び出される JavaScript。	11.0	global	
wo_bnl_d_o	String	onLabelkeydown イベントが発生した場合(ユーザがキーボードのキーを押した場合)に呼び出される JavaScript。	11.0	global	
o_b_l_r	String	onLabelkeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出される JavaScript。	11.0	global	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
onLabelkeyup	String	onLabelkeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出される JavaScript。		11.0	global
onLabelmousedown	String	onLabelmousedown イベントが発生した場合(ユーザがマウスボタンをクリックした場合)に呼び出される JavaScript。		11.0	global
onLabelmousemove	String	onLabelmousemove イベントが発生した場合(ユーザがマウスポインタを移動した場合)に呼び出される JavaScript。		11.0	global
onLabelmouseout	String	onLabelmouseout イベントが発生した場合(ユーザが pageBlockSection 列の左側の「表示ラベル」セルからマウスポインタを移動した場合)に呼び出される JavaScript。		11.0	global
onLabelmouseover	String	onLabelmouseover イベントが発生した場合(ユーザが pageBlockSection 列の左側の「表示ラベル」セルにマウスポインタを重ねた場合)に呼び出される JavaScript。		11.0	global
onLabelmouseup	String	onLabelmouseup イベントが発生した場合(ユーザがマウスボタンを放した場合)に呼び出される JavaScript。		11.0	global
display	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		11.0	global

apex:pageBlockTable

標準の Salesforce ページの関連リストまたはリストビューに類似した、`list` コンポーネントまたは `table` は `list` コンポーネントのいずれかにテーブルとして表示されるデータのリストです。同様に、`table` と `list` は一連のデータを反復することによって定義され、1 行あたり 1 つのデータ項目に関する情報を表示します。このデータセットには、最大 1,000 項目を含めることができます。

本 文には、`list` テーブルに似た、データの各項目に関して表示する情報を指定する 1 つ以上の列コンポーネントが含まれます。`list` コンポーネントとは異なり、`list` のデフォルトのスタイルは、標準の Salesforce スタイルに一致します。`list` 属性で指定されたその他のスタイルは、標準の Salesforce スタイルに追加されます。

列の `header` 属性として `sObject` 項目を指定した場合、デフォルトで、その項目の関連する表示ラベルが列ヘッダーとして使用されます。この動作を上書きするには、列の `header` 属性か、または列の `headerFacet` を使用します。

Salesforce.com API バージョン 20.0 以降が稼動している Visualforce ページでは、このコンポーネントに `p` タグを含めて列を生成できます。

このコンポーネントでは、「html-」プレフィックスを使用した **HTML パススルー属性**がサポートされています。パススルー属性は、生成されたテーブルの `td` タグに適用されます。

例

```

<apex:pageBlockTable border="1">
    <apex:column>
        <p>Left Column Content</p>
    </apex:column>
    <apex:column>
        <p>Center Column Content</p>
    </apex:column>
    <apex:column>
        <p>Right Column Content</p>
    </apex:column>
</apex:pageBlockTable>

```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
align	String	表示される HTML テーブルのページに対する位置。使用できる値には、「left」、「center」、または「right」があります。指定されていない状態にすると、この値はデフォルトの「left」に設定されます。		12.0	global
logo	String	この属性は Salesforce API バージョン 18.0 では使用できなくなりました。ページへの影響はありません。		12.0	global
border	String	表示される HTML テーブルの周囲のフレームの幅(ピクセル単位)。		12.0	global
cols	String	caption facet が指定されている場合、表示される HTML テーブルのキャプションの表示に使用されるスタイルクラス。この属性は、主に、外部 CSS		12.0	global

属性名	属性型	説明	必須項目	API バージョン	アクセス
lc	p	String	captionFacet	12.0	global
cl	c g p	String	cellSpace	12.0	global
cl	c g c p	String	cellGutterSpace	12.0	global
lo u m	c	String	columnClasses	12.0	global
lo u m	c	Integer	columns	12.0	global
lo u m	c	String W	columnsWidth	12.0	global
d r		String	direction	12.0	global
r	f	Integer	firstIndex	12.0	global

属性名	属性型	説明	必須項目	API バージョン	アクセス
一連のレコードの最初の 2 つの要素を表示しない場合は、first="2" と設定します。					
oor_l	f c	String	footerfacet が指定されている場合、表示される HTML テーブルのフッター(一番下の行)を表示するために使用されるスタイルクラス。この属性は、主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。	12.0	global
r m	f	String	a このページブロックテーブルに引かれる境界線。使用できる値には、「none」、「above」、「below」、「hsides」、「vsides」、「lhs」、「rhs」、「box」、および「border」があります。指定されていない場合、この値はデフォルトの「border」に設定されます。	12.0	global
d r l	h c	String	a headerfacet が指定されている場合、表示される HTML テーブルのヘッダーの表示に使用されるスタイルクラス。この属性は、主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。	12.0	global
d		String i	ページの他のコンポーネントが pageBlockTable コンポーネントを参照できるようにする識別子。	12.0	global
l g		String	a 「en」または「en-US」など、生成された HTML 出力の基本言語。この属性についての詳細は、 W3C 仕様 を参照してください。	12.0	global
ol	c c	String i	onclick イベントが発生した場合(ユーザがページブロックテーブルをクリックした場合)に呼び出される JavaScript。	12.0	global
dbl l	c c	String i	ondblclick イベントが発生した場合(ユーザがページブロックテーブルをダブルクリックした場合)に呼び出される JavaScript。	12.0	global
no d o	k	String	onkeydown イベントが発生した場合(ユーザがキーボードのキーを押した場合)に呼び出される JavaScript。	12.0	global
o r	p k	String	onkeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出される JavaScript。	12.0	global
o u	p k	String	onkeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出される JavaScript。	12.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
w onmousedown	String	s onmousedown イベントが発生した場合(ユーザがマウスボタンをクリックした場合)に呼び出される JavaScript。		12.0	global
o o um o m	String	s onmousemove イベントが発生した場合(ユーザがマウスポインタを移動した場合)に呼び出される JavaScript。		12.0	global
o o um o u	String	s onmouseout イベントが発生した場合(ユーザがページブロックテーブルからマウスポインタを移動した場合)に呼び出される JavaScript。		12.0	global
o o um o r	String	sv onmouseover イベントが発生した場合(ユーザがページブロックテーブルにマウスポインタを重ねた場合)に呼び出される JavaScript。		12.0	global
o o um u	p String	s onmouseup イベントが発生した場合(ユーザがマウスボタンを放した場合)に呼び出される JavaScript。		12.0	global
o ol	C c String	R i onRowClick イベントが発生した場合(ユーザがページブロックテーブルの行をクリックした場合)に呼び出される JavaScript。		12.0	global
o ob1 l	C c String	R D onRowDblClick イベントが発生した場合(ユーザがページブロックリストテーブルの行をクリックした場合)に呼び出される JavaScript。		12.0	global
w o a ou o M	M String	R D onRowMouseDown イベントが発生した場合(ユーザがページブロックテーブルの行でマウスボタンをクリックした場合)に呼び出される JavaScript。		12.0	global
o o ou o M M	M String	R onRowMouseMove イベントが発生した場合(ユーザがマウスポインタをページブロックテーブルの行に重ねた場合)に呼び出される JavaScript。		12.0	global
t o o ou u M	M String	R onRowMouseOut イベントが発生した場合(ユーザがページブロックテーブルの行からマウスポインタを移動した場合)に呼び出される JavaScript。		12.0	global
o o ou r M	M String	R onRowMouseOver イベントが発生した場合(ユーザがページブロックテーブルの行にマウスポインタを重ねた場合)に呼び出される JavaScript。		12.0	global
o o ou M	M String	R U onRowMouseUp イベントが発生した場合(ユーザがページブロックテーブルの行の上でマウスボタンを放した場合)に呼び出される JavaScript。		12.0	global
r d r d	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		12.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
C	String	ページブロックテーブルの行に関連付けられた 1 つ以上のクラスのカンマ区切りのリスト。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。複数のクラスが指定されている場合、それらのクラスがすべての行に繰り返し適用されます。たとえば、columnRows="classA, classB" と指定すると、最初の行は classA でスタイル設定され、2 番目の行は classB でスタイル設定され、3 番目の行は classA でスタイル設定され、4 番目の行は classB でスタイル設定されます（以下同様）。	1	12.0	global
Integer	s	このページブロックテーブルの行の数。	1	12.0	global
String	s	ページブロックテーブルのセルの間に引かれる境界線。使用できる値には、「none」、「groups」、「rows」、「cols」、および「all」があります。指定されていない場合、この値はデフォルトの「none」に設定されます。	1	12.0	global
String	s	pageBlockTable コンポーネントの表示に使用されるスタイル。主に、インライン CSS スタイルを追加するために使用されます。	1	12.0	global
C	String	pageBlockTable コンポーネントの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。 セクション 508 のため用で	1	12.0	global

Facet

facet名	説明	APIバージョン
c p	ページブロックテーブルのキャプションに表示されるコンポーネント。テーブルのキャプションの表示は、name="caption" を含む facet によって制御されるため、caption facet が pageBlockTable コンポーネントの本文に表示される順序は重要ではありません。	12.0
f	ページブロックテーブルのフッター行に表示されるコンポーネント。テーブルの最終行の表示は、name="footer" を含む facet によって制御されるため、footer facet が pageBlockTable コンポーネントの本文に表示される順序は重要ではありません。	12.0
h	ページブロックテーブルのヘッダー行に表示されるコンポーネント。テーブルの冒頭行の表示は、name="header" を含む facet によって制御されるため、header facet が pageBlockTable コンポーネントの本文に表示される順序は重要ではありません。	12.0

apex:pageMessage

このコンポーネントは、特定の重要度に対するエラー、警告、およびその他の種類のメッセージ用の Salesforce パターンを使用してカスタムメッセージをページに表示するために使用されます。「pageMessages コンポーネント」も参照してください。

例

```

<apex:pageMessage type="error">
    <p>An error has occurred. Please try again.</p>
</apex:pageMessage>

```

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
d l	String	i 情報の詳細な説明。		14.0	
c p	Boolean	s このコンポーネントが生成するHTML出力で、特殊なHTMLおよびXML文字をエスケープするかどうかを指定するboolean値。escape="false"が指定されていない場合、文字エスケープシーケンスは記述されたとおりに表示されます。この値を「false」に設定すると、悪質な方法で使用されるおそれのあるJavaScriptなどの任意のコンテンツが許可されるため、セキュリティのリスクとなる可能性があります。		14.0	
d	String	i ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
r d r d	Boolean	 コンポーネントをページに表示するかどうかを指定するboolean値。指定されていない場合、この値はデフォルトのtrueに設定されます。		14.0	global
r	String	v s i メッセージの重要度。サポートされる値は、「confirm」、「info」、「warning」、または「error」です。	はい	14.0	
r g h	Integer	s メッセージの強度。これは、メッセージの横に表示されるアイコンの表示とサイズを制御します。画像を表示しない場合は0、表示する場合は1~3を使用します(3が最高強度で最大サイズのアイコンです)。		14.0	
u r mm	String	s 概要メッセージ。		14.0	
l	String	i メッセージのタイトルテキスト。		14.0	

apex:pageMessages

このコンポーネントは、Salesforceスタイルを使用して、現在のページのすべてのコンポーネントに対して生成されたすべてのメッセージを表示します。

例

```
t n t e g n d p r d p o r t l d s e o n r e u t a s r p a d a t e t c n e' r o u i " p p v s = " a i i s "
bt lt n e ro u e q p b r l a S y = " O f i " y $ = " a s " a >
n e t t l e p b e h r p e c c k r e t c l a f a h i a C a a " " D s " a ,
n e t e e l h w t c e o n k h i h v p s a s p a a a s . > /
x ro m p f < a >
ex B l g o p p c < k a a >
exe g p q M < a ass sa > /
ex t t B l B g o p u p c < k a a s >
```

```

e   x      tit     <!--> Bndmme w o  lt u < an a o } a v a" v Sa ia =" ls "a > /
e ex    tb     lbgo pu w c < k/a a s>
e ex    e Bel{go pbl p tc aut< n kroa }u a pprva o a" ! ppi i s "v a" a >
e ex    x e h lo u empxun o < m app v a" ! . "a > /
e ex    x e en do u endrelthu l< o aC V a a" Ds "a>
e ex    x etn e { u ldp d p t & l jo a o v a" ! Ds "a > /
e ex    n lo u m < /a >
e ex    e Bl go pbl p c < T k/a a a>
e ex    B l go p p c < k/a a >
x      ro m p f < /a >
g p p < /a >

```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
c	p	Boolean s	このコンポーネントが生成するHTML出力で、特殊なHTMLおよびXML文字をエスケープするかどうかを指定するboolean値。escape="false"が指定されていない場合、文字エスケープシーケンスは記述されたとおりに表示されます。この値を「false」に設定すると、悪質な方法で使用されるおそれのあるJavaScriptなどの任意のコンテンツが許可されるため、セキュリティのリスクとなる可能性があります。		14.0
d		String i	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0 global
r	d	r d	コンポーネントをページに表示するかどうかを指定するboolean値。指定されていない場合、この値はデフォルトのtrueに設定されます。		14.0 global
o	l	h	メッセージの詳細部分を表示するかどうかを指定するboolean値。指定されていない場合、この値はデフォルトのfalseに設定されます。		14.0

apex:panelBar

ユーザが関連するヘッダーをクリックしたときに展開できる、1つ以上の `< a r p pm I <` タグを含む `a a >` ページ領域です。`< a r p pm I <` が展開されると、他のすべての項目のコンテンツを非表示にした状態で項目のヘッダーとコンテンツが表示されます。別の `< a r p pm I <` が展開されると、元の項目のコンテンツは再度非表示になります。`< a r p pe` には、最大1,000個の `< a p pm I <` タグを含められます。

このコンポーネントでは、「html-」プレフィックスを使用した **HTML パススルー属性**がサポートされています。パススルー属性は、生成されたコントナタグ `< a >` に適用されます。

例

```
gB l r : p < ! -- Pa a a >-
```

属性名	属性型	説明	必須項目	APIバージョン	アクセス
contentClass	String	panelBar コンポーネントの panelBarItem のコンテンツの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。	10.0	global	
contentStyle	String	panelBar コンポーネントの panelBarItem のコンテンツの表示に使用されるスタイル。主に、インライン CSS スタイルを追加するために使用されます。	10.0	global	
headerClass	String	panelBar コンポーネントのすべての panelBarItem ヘッダーの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。	10.0	global	
headerContentClass	String	panelBarItem が展開されたときのヘッダーの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。	10.0	global	
headerContentStyle	String	panelBar コンポーネントのすべての panelBarItem ヘッダーの表示に使用されるスタイル。主に、インライン CSS スタイルを追加するために使用されます。	10.0	global	
headerContentStyleClass	String	panelBarItem が展開されたときのヘッダーの表示に使用されるスタイル。主に、インライン CSS スタイルを追加するために使用されます。	10.0	global	
height	String	展開されたときのパネルバーの高さ。利用可能な縦方向のスペースのパーセント (height="50%" など)、またはピクセル数 (height="200px" など) のいずれかで表されます。指定されていない場合、この値はデフォルトの 100% に設定されます。	10.0	global	

属性名	属性型	説明	必須項目	APIバージョン	アクセス	
d	String i	ページの他のコンポーネントが panelBar コンポーネントを参照できるようにする識別子。		10.0	global	
m	Object i s	panelBar が表示されるときに処理されるデータのコレクション。使用されている場合、panelBar コンポーネントの本文は、dataTable または repeat コンポーネントと同様に、コレクションのそれぞれの項目に対して 1 回繰り返されます。「var 属性」も参照してください。		11.0	global	
r d r d	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		10.0	global	
l	String s y	panelBar コンポーネントのすべての部分の表示に使用されるスタイル。主に、インライン CSS スタイルを追加するために使用されます。		10.0	global	
1 1	C	String s y	panelBar コンポーネントのすべての部分の表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。		10.0	global
dn	pf String si	panelBar 項目間の切り替えに使用される実装メソッド。使用できる値には、「client」、「server」、および「ajax」があります。指定されていない場合、この値はデフォルトの「server」に設定されます。		10.0	global	
l u	Object a	panelBar が表示されるときに最初に選択される panelBarItem の ID。		10.0	global	
r	Stringy	items 属性によって指定される、データのコレクション内の 1 つの要素を表す変数の名前。名前が指定されると、この変数を使用して panelBar コンポーネントタグの本文に要素自体を表示できます。		11.0	global	
d	String i	パネルバーの幅。利用可能な横方向のスペースのパーセント (width="50%" など)、またはピクセル数 (width="800px" など) のいずれかで表されます。指定されていない場合、この値はデフォルトの 100% に設定されます。		10.0	global	

apex:panelBarItem

ユーザがセクションヘッダーをクリックしたときに展開または折りたたむことができる `l r p p` のセクションです。`ex` 展開された場合、`l r p pm I <` のヘッダーおよびコンテンツが表示されます。折りたたまれた場合、`l r p pm I <` のヘッダーのみが表示されます。

このコンポーネントでは、「html-」プレフィックスを使用した HTML パススルー属性がサポートされています。パススルー属性は、生成されたコンテナタグに適用されます。

→

e	n	gB l r : p < ! -- Pa a a >-	a	→-
e	t n l e t o C ce m I &k mro! I -ie2t m deI kl tn pt e p, h nc 3 i sfn ya p	g p p < a a	a	→-
x	n	B l r p p < a a >a		
ex	at eB l er tp l pmltI & dxa I at a B l ra" pl p"m > II < a a / a a a >			
ex	at eB l er tp l pmltI & dxa I at a B l ra" p p"m > I < a a / a a a >			
ex	at eB l er tp l pmltI & dxa I at a B l ra" p p"m > I < a a / a a a >			
ex	n	B l r p p < /a a >a		
		g p p < /a a		
e	at e gB l rn: p m< I ! E -- Pa a av s >-			
		g p p < a a>		
e		g p gpM < a ass sa > /		
n	B l r p p < a a >a			
x	at B l r p pm I < a a a a			
		et el bl n m I a" o "		
e	t n e to e r m er tne r n Eg om = a a ('i i ;" ')			
e	n e t le r tne n g om v = a La (v i ai ;" ')			
		ent tn am Io c		
ex	at B l r p pm I < /a a a >			
x	at B l r p pm I < a a a a			
		et l bwl mo I T a" "		
e	t n e to e r m er tn trw Eg mo = a a ('i i ;" ')			
e	n e t le r tne t w g mov = a La (v i ai ;" ')			
t	w e t tn nmo Io c			
ex	at B l r p pm I < /a a a >			
n	B l r p p < /a a >a			
		g p p < /a a		

属性

	属性名	属性型	説明	必須項目	APIバージョン	アクセス
o l	c	C	panelBarItem コンポーネントのコンテンツの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。		10.0	global
o	lc		panelBarItem コンポーネントのコンテンツの表示に使用されるスタイル。主に、インライン CSS スタイルを追加するために使用されます。		10.0	global
d d	p	String	この panelBarItem のコンテンツを表示するかどうかを指定する boolean 値。		10.0	global
d r l	h	C	panelBarItem コンポーネントのヘッダーの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。		10.0	global
d r A	hc	C	panelBarItem コンポーネントのコンテンツが表示されるときに、 panelBarItem のヘッダーの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。		10.0	global
d r l h	lh		panelBarItem コンポーネントのヘッダーの表示に使用されるスタイル。主に、インライン CSS スタイルを追加するために使用されます。		10.0	global
d r A l	hc	C	panelBarItem コンポーネントのコンテンツが表示されるときに、 panelBarItem コンポーネントのヘッダーの表示に使用されるスタイル。主に、インライン CSS スタイルを追加するために使用されます。		10.0	global
d	i	String	ページの他のコンポーネントが panelBarItem を参照できるようにする識別子。		10.0	global
l b l		String	panelBarItem コンポーネントのヘッダーとして表示されるテキスト。		10.0	global
m		Object	panelBarItem の名前。この属性の値を使用して、 panelBar のデフォルトの展開済みの panelItem を指定します。		11.0	global
o r		String	panelBarItem が選択されていないときにユーザがコンポーネントをクリックして選択した場合に呼び出される JavaScript。		16.0	
o	v		ユーザが別の panelBarItem を選択したときに呼び出される JavaScript。		16.0	

属性名	属性型	説明	必須項目	API バージョン	アクセス
rendered	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		10.0	global

apex:panelGrid

HTML テーブル要素を表示します。`l r d p p` の本文に含まれる各コンポーネントは、列数に達するまで、最初の行の対応するセルに配置されます。列数に達したら、その次のコンポーネントは次の行の最初のセルに配置されます。

コンポーネントによって生成されるすべてのコンテンツは、単一の `l r d p p` セルに配置されます。`G >` コンポーネントは、一連のデータを反復変数によって処理しないため、`bl p` とは異なります。

`l r o p p` も参照してください。

このコンポーネントでは、「html-」プレフィックスを使用した **HTML パススルー属性** がサポートされています。パススルー属性は、生成されたコンテナタグに適用されます。

例

```
g p p < a >
  l r d p lepu m c<d a d ahG s="3" i="iG" >
  txe ou u p l aiP r <Tt d a r v h a" s" i="i s" > /
  txe oe u p l nepte <d a ac d v h ca" S" i="i s" > /
  txe ou u p l aiP r dT d ha Td v h a" Ti "i="i " > /
  txe ou u p l uept aGT t d a h au v h a" hF "i="i F " > /
  l r d p p < /a aiG >
  g p p < /a >
```

上述の例では次の HTML を表示します。

```
  r d h < a="iG" >
    < >
      < >
        < >
          < >
            < >
              < >
                < >
                  < >
                    < >
                      < >
                        < >
                          < >
```

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
b logo	c	String 表示される HTML テーブルの背景色。		10.0	global
b rd r		Integer 表示される HTML テーブルの周囲のフレームの幅(ピクセル単位)。		10.0	global
n ol	c p C	String i a caption facet が指定されている場合、表示される HTML テーブルのキャプションの表示に使用されるスタイルクラス。この属性は、主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。		10.0	global
n o	Lc p	String i a caption facet が指定されている場合、表示される HTML テーブルのキャプションの表示に使用されるスタイル。この属性は、主に、インライン CSS スタイルを追加するために使用されます。		10.0	global
ml dd	c g p	String 各テーブルセルの境界線とそのコンテンツ間のスペース。この属性の値がピクセル単位の長さである場合、4辺の余白のすべてでコンテンツからの距離がこの値に設定されます。属性の値がパーセント単位の長さである場合は、上下の余白は利用可能な縦方向のスペースのパーセントに基づいてコンテンツからの距離が上下均等の長さに分割され、左右の余白は利用可能な横方向のスペースのパーセントに基づいてコンテンツからの距離が左右均等の長さに分割されます。		10.0	global
hl	c g c p	String s 各テーブルセルの境界線とそのセルを囲む他のセルの境界線および/またはテーブルの境界の間のスペース。この値はピクセルまたはパーセント単位で指定する必要があります。		10.0	global
lo u l m	c	String columnClasses の列に関連付けられている 1 つ以上の CSS クラスのカンマ区切りのリスト。複数の CSS クラスが指定されている場合、クラスはすべての列に繰り返し適用されます。たとえば、columnClasses="classA, classB" と指定すると、最初の列は classA でスタイル設定され、2 番目の列は classB でスタイル設定され、3 番目の列は classA でスタイル設定され、4 番目の列は classB でスタイル設定されます(以下同様)。		10.0	global
lo u m	c	Integer s この panelGrid の列の数。		10.0	global
d r		String i 生成された HTML コンポーネントが読み取られる方向。使用可能な値には「RTL」(右から左) または「LTR」(左から右) があります。		10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
footer	String	footerFacet が指定されている場合、表示される HTML テーブルのフッター(一番下の行)を表示するに使用されるスタイルクラス。この属性は、主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。	10.0	global	
frame	String	a このテーブルに引かれる境界線。使用できる値には、「none」、「above」、「below」、「hsides」、「vsides」、「lhs」、「rhs」、「box」、および「border」があります。指定されていない場合、この値はデフォルトの「border」に設定されます。 「rules 属性」も参照してください。	10.0	global	
header	String	a headerFacet が指定されている場合、表示される HTML テーブルのヘッダーの表示に使用されるスタイルクラス。この属性は、主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。	10.0	global	
id	String	i ページの他のコンポーネントが panelGrid コンポーネントを参照できるようにする識別子。	10.0	global	
language	String	a 「en」または「en-US」など、生成された HTML 出力の基本言語。この属性についての詳細は、 W3C仕様 を参照してください。	10.0	global	
click	String	i onclick イベントが発生した場合(ユーザがパネルグリッドをクリックした場合)に呼び出される JavaScript。	10.0	global	
dblclick	String	i ondblclick イベントが発生した場合(ユーザがパネルグリッドをダブルクリックした場合)に呼び出される JavaScript。	10.0	global	
keydown	String	y onkeydown イベントが発生した場合(ユーザがキーボードのキーを押した場合)に呼び出される JavaScript。	10.0	global	
keypress	String	y onkeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出される JavaScript。	10.0	global	
keyup	String	y onkeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出される JavaScript。	10.0	global	
mousedown	String	s onmousedown イベントが発生した場合(ユーザがマウスボタンをクリックした場合)に呼び出される JavaScript。	10.0	global	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
onmousemove	String	s onmousemove イベントが発生した場合(ユーザがマウス pointer を移動した場合)に呼び出される JavaScript。		10.0	global
onmouseout	String	s onmouseout イベントが発生した場合(ユーザがパネル grid からマウス pointer を移動した場合)に呼び出される JavaScript。		10.0	global
onmouseover	String	sv onmouseover イベントが発生した場合(ユーザがパネル grid にマウス pointer を重ねた場合)に呼び出される JavaScript。		10.0	global
onmouseup	String	s onmouseup イベントが発生した場合(ユーザがマウス button を放した場合)に呼び出される JavaScript。		10.0	global
rowd	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		10.0	global
rowspan	C	ss テーブルの行に関連付けられた 1 つ以上の CSS クラスのカンマ区切りのリスト。複数の CSS クラスが指定されている場合、クラスはすべての行に繰り返し適用されます。たとえば、columnRows="classA, classB" と指定すると、最初の行は classA でスタイル設定され、2 番目の行は classB でスタイル設定され、3 番目の行は classA でスタイル設定され、4 番目の行は classB でスタイル設定されます(以下同様)。		10.0	global
rules	C	s このテーブルのセルの間に引かれる境界線。使用できる値には、「none」、「groups」、「rows」、「cols」、および「all」があります。指定されていない場合、この値はデフォルトの「none」に設定されます。「frame 属性」も参照してください。		10.0	global
style	String	s y panelGrid コンポーネントの表示に使用されるスタイル。主に、インライン CSS スタイルを追加するために使用されます。		10.0	global
styleClass	C	String s y panelGrid コンポーネントの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。		10.0	global
summary	C	String s セクション 508 の準拠に関するテーブルの目的と構造の概要。		10.0	global
title	String	i ユーザがコンポーネントにマウス pointer を重ねたときにツールチップとして表示されるテキスト。		10.0	global

属性名	属性型	説明	必須項目	API バージョン	アクセス
h	String	i テーブル全体の幅。利用可能な横方向の合計スペースに対する相対パーセント (width="80%" など)、またはピクセル数 (width="800px" など) のいずれかとして表されます。	10.0	global	

Facet

facet名	説明	API バージョン
c	p テーブルのキャプションに表示されるコンポーネント。テーブルのキャプションの表示は、name="caption" を含む facet によって制御されるため、caption facet が panelGrid コンポーネントの本文に表示される順序は重要ではありません。	10.0
f	f テーブルのフッター行に表示されるコンポーネント。テーブルの最終行の表示は、name="footer" を含む facet によって制御されるため、footer facet が panelGrid コンポーネントの本文に表示される順序は重要ではありません。	10.0
h	h テーブルのヘッダー行に表示されるコンポーネント。テーブルの最初の行の表示は、name="header" を含む facet によって制御されるため、header facet が panelGrid コンポーネントの本文に表示される順序は重要ではありません。	10.0

apex:panelGroup

複数の子コンポーネントを単一の panelGrid セルに表示できるようにする、複数の子コンポーネントのコンテナです。 apex:panelGroup は、子コンポーネントである必要があります。

例

```

<apex:panelGrid columns="3">
    <apex:panelGroup>
        <apex:inputText value="1" />
        <apex:inputText value="2" />
        <apex:inputText value="3" />
    </apex:panelGroup>
    <apex:panelGroup>
        <apex:inputText value="4" />
        <apex:inputText value="5" />
        <apex:inputText value="6" />
    </apex:panelGroup>
    <apex:panelGroup>
        <apex:inputText value="7" />
        <apex:inputText value="8" />
        <apex:inputText value="9" />
    </apex:panelGroup>
</apex:panelGrid>

```

上述の例では次の HTML を表示します。

```

<table border="1">
    <tr>
        <td>1</td>
        <td>2</td>
        <td>3</td>
    </tr>
    <tr>
        <td>4</td>
        <td>5</td>
        <td>6</td>
    </tr>
    <tr>
        <td>7</td>
        <td>8</td>
        <td>9</td>
    </tr>
</table>

```

end	el n	p d h	tnd < > s c ida="	p s " >	s	s </> a	*
end	d	r p ou h	< < > ps i a="	G " >			
ent	d	tpd h ndh	< Ths iTa="	p i " >	i <	s / > a	
ent	t d	t opu h em	h s ha="	Fp " F <	s <	s / > a	
tn	d	p	< s </> a	*			
r			< *				
b do			< / > y				
bl			< / a>				

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
	String i	ページの他のコンポーネントが panelGrid コンポーネントを参照できるようにする識別子。	10.0		global
	String ya	パネルグループのレイアウトスタイル。使用できる値には、「block」(HTML div タグを生成)、「inline」(HTML span タグを生成)、および「none」(HTML タグを生成しない)があります。指定されていない場合、この値はデフォルトの「inline」に設定されます。	10.0		global
	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	10.0		global
	String s	panelGroup コンポーネントの表示に使用されるスタイル。主に、インライン CSS スタイルを追加するため使用されます。	10.0		global
C	String s	panelGroup コンポーネントの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するため使用されます。	10.0		global

apex:param

親コンポーネントのパラメータです。 `mp` `p` コンポーネントは、次のコンポーネントの子としてのみ指定できます。

n	n.	n	upc	o	c <	a	i	a	F	i	>
n	t	•	o	upro	pp	a	i	a	S	>	
n	no	dmm	xe		<	k	L	ia		>	
	no	u	u	p	p <	ka	L	i		>	
t x	o	u	u	p	p <T	a				>	
w	•	r	f	•	<	i	v	i		>	

内では、Java の `MessageFormat` クラスの構文に一致させるための
ポートされています。詳細は、[MessageFormat クラスの JavaDocs](#)を参照してください。

apex:outputLink の例

```

e   ex ro te e l m o d r p Etl i se opt e p an o e ec li p ro y s s c a V p i s a
t etnl d h o e trn crd c i c v à ha a i U L R .
ro   l m t ept f tF tn Eteoi,1ltc00 000B003h5 g ddDb a h , s: U L R s
e   he n erap xe fc sc m /0sp i s a /a I /y Eal 00 000B003 5
e e el hro e l fc V t Sdt i ps a & scuro l s at G Tm infro m a f i a i
t   n tegnd prd pér h k C o a ac ca a " " " a>
x t t enou utt p le up & diag & op i nv c a! ch // . s / ' a>
e   r od g S a G
x t t n re { mp mpt ltnau < no a} ca amc=a "v a! ! a. "a > /
nou u p p < K a L i >
g p p < /a >

```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
param	Object	この param の値を関連する Visualforce コントローラの変数に割り当てる setter メソッド。この属性を使用する場合は、getter メソッドおよび setter メソッドまたは get 値および set 値を含むプロパティを定義する必要があります。		10.0	global
label	String	ページの他のコンポーネントが param コンポーネントを参照できるようにする識別子。		10.0	global
name	String	name="Location" など、このパラメータのキー。		10.0	global
value	Object	value="San Francisco, CA" など、このパラメータには関連付けられるデータ。value 属性は、string、number、または boolean 値に設定する必要があります。value 以外に文字列の置換操作を行うときに必要な属性はないため、param コンポーネントの唯一の必須属性です。たとえば、outputText コンポーネントの値として「My {0}」を使用して、outputText コンポーネントの本文に param を含める場合は、param タグの値によって出力文字列の {0} が置き換えられます。		10.0	global

apex:pieSeries

Visualforce の円グラフで系列として表示されるデータ系列です。少なくとも、それぞれの円グラフの系列のラベルと値のペアとして使用する、データコレクションの項目を指定する必要があります。

- 注: このコンポーネントは apex:chart コンポーネントで囲む必要があります。グラフには apex:chart を 1 つだけ使用できます。

例

```

x t t : < ! -- Pa >-
x e e e r dt epn } t < wtp a g h ahd =a al i h'a ai = 3 " 00 = " 4 80
x t r ph < /a >

```

属性

	属性名	属性型	説明	必須項目	APIバージョン	アクセス
lоро	c	String	円グラフの系列の塗りつぶしの色として順に使用される一連の色の値。色は、HTMLスタイル(16進)の色をカンマ区切りで指定します。たとえば、 です。# F0F0F0, #000000		23.0	
d	ld	String	系列内の各円グラフ系列のデータ値の取得元である、グラフデータに指定された各レコードの項目。この項目はグラフデータのすべてのレコードに存在している必要があります。	はい	23.0	
d o u	rd	Integer	円グラフの中心に配置する穴の半径を、円の半径のパーセントとして表す整数。値が指定されていない場合は0が使用され、通常の穴のない円グラフが作成されます。		26.0	
l g	rg_h_h	Boolean	マウスポインタを重ねたときに各円グラフの系列を強調表示するかどうかを指定するboolean値。指定されていない場合、この値はデフォルトのtrueに設定されます。		23.0	
d	label	String	ページの他のコンポーネントがグラフコンポーネントを参照できるようにする識別子。		23.0	global
l b l	ld	String	系列内の各円グラフ系列のラベルの取得元である、グラフデータに指定された各レコードの項目。この項目はグラフデータのすべてのレコードに存在している必要があります。指定されていない場合、この値はデフォルトの「name」に設定されます。		23.0	
r d r d	rd	Boolean	グラフでグラフ系列を表示するかどうかを指定するboolean値。指定されていない場合、この値はデフォルトのtrueに設定されます。		23.0	
r n d r r	rd	String	円グラフの各系列が表示される方法を追加または上書きするJavaScript関数の名前を指定する文字列。追加のスタイルを指定またはデータを追加するために実装します。		26.0	
no	dg_I	Boolean	凡例が有効な場合に、グラフの凡例にこの系列を表示するかどうかを指定するboolean値。指定されていない場合、この値はデフォルトのtrueに設定されます。		23.0	

属性名	属性型	説明	必須項目	API バージョン	アクセス
p	Boolean	マウスポインタを重ねたときに各円グラフ系列のツールチップを表示するかどうかを指定する boolean 値。ツールチップの形式は <labelField>: <dataField> です。指定されていない場合、この値はデフォルトの true に設定されます。		23.0	

apex:radarSeries

放射状の Visualforce グラフで一連の線で結ばれた内側の領域として表示されるデータ系列です。レーダーグラフは、「クモの巣」グラフと呼ばれることもあります。少なくとも、各点の X 値および Y 値として使用するデータコレクションの項目、および目盛りとして使用する放射軸を指定する必要があります。

注: このコンポーネントは **apex:chart** コンポーネントで囲む必要があります。1 つのグラフに複数の **radarSeries** コンポーネントを含めることができます。

例

```
<apex:chart type="radar" title="Sales Performance Radar Chart" x-axis="Quarter" y-axis="Revenue" data="[[{"Quarter": "Q1", "Category": "Sales", "Value": 100}, {"Quarter": "Q1", "Category": "Marketing", "Value": 80}, {"Quarter": "Q1", "Category": "Customer Support", "Value": 70}, {"Quarter": "Q1", "Category": "Product Development", "Value": 60}, {"Quarter": "Q1", "Category": "Finance", "Value": 50}, {"Quarter": "Q2", "Category": "Sales", "Value": 120}, {"Quarter": "Q2", "Category": "Marketing", "Value": 90}, {"Quarter": "Q2", "Category": "Customer Support", "Value": 80}, {"Quarter": "Q2", "Category": "Product Development", "Value": 70}, {"Quarter": "Q2", "Category": "Finance", "Value": 60}, {"Quarter": "Q3", "Category": "Sales", "Value": 110}, {"Quarter": "Q3", "Category": "Marketing", "Value": 85}, {"Quarter": "Q3", "Category": "Customer Support", "Value": 75}, {"Quarter": "Q3", "Category": "Product Development", "Value": 65}, {"Quarter": "Q3", "Category": "Finance", "Value": 55}, {"Quarter": "Q4", "Category": "Sales", "Value": 130}, {"Quarter": "Q4", "Category": "Marketing", "Value": 100}, {"Quarter": "Q4", "Category": "Customer Support", "Value": 90}, {"Quarter": "Q4", "Category": "Product Development", "Value": 80}, {"Quarter": "Q4", "Category": "Finance", "Value": 70}], [{"x": 100, "y": 100}]]>
```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
f	String	線の内側の領域を塗りつぶすために使用する色を HTML スタイルの(16進)色として指定する文字列。指定されていない場合、色はグラフの colorSet またはテーマから順に使用されます。線とマーカーのみを使用し、塗りつぶしなしのグラフの場合は、塗りつぶしを「なし」に設定します。塗りつぶしなしに場合は、デフォルトで非表示になっているストロークとマーカーの属性を必ず設定してください。		26.0	
hg	Boolean	マウスポインタを重ねたときに各点を強調表示するかどうかを指定する boolean 値。指定されていな		26.0	

属性名	属性型	説明	必須項目	API バージョン	アクセス
d	String	i い場合、この値はデフォルトの true に設定されます。			
r m l1	k String	a HTML スタイルの(16進)色として指定されるこの系列のデータポイントマーカーの色。系列のマーカーがグラフに表示されるようにするには、少なくとも 1 つのマーカー属性を設定する必要があります。		23.0	global
z r m	k Integer	a i この系列の各データポイントマーカーのサイズ。系列のマーカーがグラフに表示されるようにするには、少なくとも 1 つのマーカー属性を設定する必要があります。		23.0	
r m	# String	a v この系列の各データポイントマーカーの形状。有効なオプションは、次のとおりです。 <ul style="list-style-type: none"> circle cross 系列のマーカーがグラフに表示されるようにするには、少なくとも 1 つのマーカー属性を設定する必要があります。		23.0	
o	c p	Integer i a 系列の塗りつぶされた領域の不透明度を表す 0 ~ 1 までの小数値。塗りつぶしが設定されている場合にのみ影響します。		26.0	
r d r d		Boolean a グラフでグラフ系列を表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		26.0	
no	# g I	Boolean s L a このグラフ系列をグラフの凡例に追加するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		26.0	
r o l o r o	C	kString s a HTML スタイルの(16進)色として、この系列の折れ線の色を指定する文字列。指定されていない場合、線は塗りつぶしと同じ色になるため、線が見えなくなります。		26.0	
r o d	h	kInteger s i この系列の折れ線の太さを指定する整数。指定されていない場合、線は描画されません。塗りつぶしも「なし」に設定されていると、この系列はグラフに表示されなくなります。		26.0	
p	Boolean i	s マウスポインタを重ねたときに各データポイントマーカーのツールチップを表示するかどうかを指		26.0	

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
d	String i	ページの他のコンポーネントが relatedList コンポーネントを参照できるようにする識別子。		10.0	global
l	String i s	表示する関連リスト。これは、オブジェクトのページレイアウト上にある必要はありません。この値を指定するには、子リレーションの名前を関連オブジェクトに使用します。たとえば、取引先詳細ページに通常表示される [取引先責任者] 関連リストを表示するには、list="Contacts" を使用します。	はい	10.0	global
z	g p	関連リストにデフォルトで表示するレコード数。指定されていない場合、この値はデフォルトの 5 に設定されます。	ia	10.0	global
r d r d	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		10.0	global
b u j	String c s	データおよび関連リストの定義の取得元の親レコード。指定されていない場合、また、標準コントローラを使用している場合、この値は、ページ URL の ID クエリ文字列パラメータの値に自動的に設定されます。		10.0	global
l	String i	関連リストのタイトルとして表示されるテキスト。指定されていない場合、この値はデフォルトの、アプリケーションで指定されたタイトルに設定されます。		10.0	global

Facet

facet 名	説明	API バージョン
b do	関連リストの本文に表示されるコンポーネント。関連リストの本文の表示は、name="body" を含む facet によって制御されるため、body facet が relatedList コンポーネントに表示される順序は重要ではありません。指定されている場合、この facet は関連リストタグの他のコンテンツより優先されます。	10.0
oor	関連リストのフッター領域に表示されるコンポーネント。関連リストの下部の表示は、name="footer" を含む facet によって制御されるため、footer facet が relatedList コンポーネントの本文に表示される順序は重要ではありません。	10.0
d r	関連リストのヘッダー領域に表示されるコンポーネント。関連リストの上部の表示は、name="header" を含む facet によって制御されるため、header	10.0

facet名	説明	APIバージョン
	facetがrelatedListコンポーネントの本文に表示される順序は重要ではありません。	

apex:repeat

指定する構造に基づいてコレクションのコンテンツを出力できる反復コンポーネントです。コレクションには、最大 1,000 個の項目を含められます。

コンポーネントが生成したすべてのコンテンツは、**单一の** `l gn p pœx c n または l rad p p > <` セルに配置されます。

このコンポーネントを次のコンポーネントの直接の子として使用することはできません。

B	l	r	p	p	<	a	a	>a	
b	x		p̪	oɔ	k̪	c	<	ka	s >
b			p	c		<	as	L	i s >
b		d	p o	c		<	as	R i	a>
b		l	p		<	a	a	Pa>	

例

g : < ! -- Pa >-
e e goptlā ipcrnet < o da pCa g h ="
r t lp un p}r t < g nate re v adg a" ls i siv =" ap i " i =" R " a
txe ou u tp l up e)t{T e gl l vu hbra" ls i " iv =" 'x > / > /
r p p < /a >a
g p p < /a >a
r ld r C *** : / *** /
rn p c c o pC i ss a a
t tr { p ng g] g i } E is [is s () T S EE TW[N ' , ' G , ' R ' ; '
{ ru n r g] } E

上述の例では次の HTML を表示します。

t	p	t	g	h	s	l	i	p	a	=	:m	Pa	bR	E	V	ap	0	"a	x	Q	s	<	/
t	p	t	g	h	s	l	i	p	a	=	:m	Pa	bR	1	VT	ap	W	"a	x	O	s	<	/
t	p	t	g	h	s	l	i	p	a	=	:h n	Pa	Rb	EVT	a	p	H	>	R	s	<	/	

標準コンポーネントの例

```

e   ex ro te e     l m m o d r p E l i s o t p e p a n o e ec li h p o y s s g c a V p a i s a
e   et l d h @ t u r n c d i c v á h a a i U L R .
e   ro l m t e p t f F t n t e a i q u h t 66 000R0035 g d a D b h , s: U L R s
e   e he n e r o p x e f c s c m / o s a i s a / a I / y E a l R 00 000R003 5
e   e e elu h r o e l f c v t S d t i p s a e c u r o l s n t g o T m i n f o m a f i a i i a .

g : < ! -- Pa >-
t n t e g n d p r d p o r h a n c a u a c c a a & " " >
e bl b r d r < a & " " &
r < >
t e e t t h r G n t n r & g h > s a & < > a & < > a & < > a & < > a
t e t r t r & h t l t m t E < h u h a & a & a < > & s a & < > a
r < >
e ex t & e r p p l & e a n a o u v a c e " a s & & ! . s s a " >
r < >
{ t e e d e t k h r n c d < > ! s s a . s N a < > a
{ t e d t n } r c g d < > ! s s a & i < > a
{ t e d t e t n o t c } a m & d > ! s s a . a . i < a & < > a
{ t e t d t t } c u d < > ! s s a . s & s a & < > a
r < >
e ex t r p p < / a > a
bl < / a>
g p p < / a > a

```

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
f	Integer	視覚的に表示されるコレクションの最初の要素。 value 属性によって指定されたデータセットの最初の要素のインデックスは 0 です。たとえば、value 属性で指定されている一連のレコードの最初の 2 つの要素を表示しない場合は、first="2" と設定します。		10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
d	String i	ページの他のコンポーネントが繰り返しコンポーネントを参照できるようにする識別子。	10.0	global	
r d r d	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	10.0	global	
r o	Integer s	表示されるコレクションの最大項目数。この値がコレクションの項目数より小さい場合、コレクションの最後の項目は繰り返されません。	10.0	global	
l u	Object a	反復処理されるデータのコレクション。	10.0	global	
r	Stringv	反復内の現在の項目を表す変数の名前。	10.0	global	

apex:scatterSeries

Visualforce 線形グラフで(結ばれていない)個々の点として表示されるデータ系列です。少なくとも、各点の X 値および Y 値として使用するデータコレクションの項目、および目盛りとして使用する X 軸および Y 軸を指定する必要があります。

注: このコンポーネントはコンポーネントで囲む必要があります。1つのグラフに複数のコンポーネントを含めることができます。コンポーネントも追加できますが、判読しにくい結果になる可能性があります。

例

```
<apex:scatterSeries x="x" y="y" data="data" x-axis="xAxis" y-axis="yAxis" >
```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
axis	String	<p>このグラフの系列のバインド先の軸。グラフの 4 辺の境界の 1 つである必要があります。</p> <ul style="list-style-type: none"> left right top bottom <p>この軸のバインド先は同階層のコンポーネントによって定義される必要があります。</p>		26.0	
highlighted	Boolean	マウスポインタを重ねたときに各点を強調表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		26.0	
label	String	ページの他のコンポーネントがグラフコンポーネントを参照できるようにする識別子。		26.0	global
markerColor	String	HTML スタイルの(16進)色として指定されるこの系列のデータポイントマーカーの色。		26.0	
markerSize	Integer	この系列の各データポイントマーカーのサイズ。		26.0	
markerShape	String	<p>この系列の各データポイントマーカーの形状。有効なオプションは、次のとおりです。</p> <ul style="list-style-type: none"> circle cross <p>指定されていない場合、マーカーの形状は一連の形状から選択されます。</p>		26.0	
orderable	Boolean	グラフでグラフ系列を表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		26.0	
overrides	String	各データポイントが表示される方法を追加または上書きする JavaScript 関数の名前を指定する文字列。追加のスタイルを指定またはデータを追加するためには実装します。		26.0	
addAll	Boolean	このグラフ系列をグラフの凡例に追加するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		26.0	
toolTip	Boolean	マウスポインタを重ねたときに各データポイントマーカーのツールチップを表示するかどうかを指定する boolean 値。このツールチップの形式は		26.0	

	属性名	属性型	説明	必須項目	APIバージョン	アクセス
l			<xField>: <yField> です。指定されていない場合、この値はデフォルトの true に設定されます。			
ld	title	String	このグラフ系列のタイトル。グラフの凡例に表示されます。		26.0	
ld	xLabel	String	系列のデータポイントごとの X 軸の値の取得元であります、グラフデータに指定された各レコードの項目。この項目はグラフデータのすべてのレコードに存在している必要があります。	はい	26.0	
ld	yLabel	String	系列のデータポイントごとの Y 軸の値の取得元であります、グラフデータに指定された各レコードの項目。この項目はグラフデータのすべてのレコードに存在している必要があります。	はい	26.0	

apex:scontrol

Sコントロールを表示するインラインフレームです。

注意: SコントロールはVisualforceページに置き換えられました。2010年以降、新しい組織同様、Sコントロールを作成したことのない組織は、Sコントロールを作成できなくなります。既存のSコントロールに影響はありません。

例

```
e ro t n ov dm c o t e p! B u uims t k elmd e ornlqdy c sd v xf a as - i . >
x t n t er le por lo c <mll orad N H W = " " > /
g p p < /a >
```

属性

	属性名	属性型	説明	必須項目	APIバージョン	アクセス
or lo	name	String	表示されるSコントロールの名前。この値には、Sコントロールの表示ラベルではなく、名前項目を使用します。		10.0	global
g	height	Integer	Sコントロールを表示するインラインフレームの高さ。利用可能な縦方向の合計スペースのパーセント(height="50%"など)、またはピクセル数(height="300px"など)のいずれかで表されます。		10.0	global
d	componentId	String	ページの他のコンポーネントがSコントロールコンポーネントを参照できるようにする識別子。		10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
rendered	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	10.0	global	
scrollable	Boolean	Sコントロールがスクロール可能であるかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	10.0	global	
bound	Object	このSコントロールにデータを提供するレコードの ID。	10.0	global	
width	Integer	Sコントロールを表示するインラインフレームの幅。利用可能な横方向の合計スペースのピクセル数、またはパーセントのいずれかとして表されます。ピクセル数を指定するには、この属性を数字 + px (width="600px" など) に設定します。パーセントを指定するには、ハイフン+数字 (width="-80%" など) に設定します。	10.0	global	

apex:sectionHeader

ページのタイトルバーです。標準の Salesforce ページのタイトルバーは、タブバーの直下に表示される色付きのヘッダーです。

このコンポーネントでは、「html-」プレフィックスを使用した [HTML パススルー属性](#)がサポートされています。パススルー属性は、生成されたコンテナタグ `<apex:sectionHeader>` に適用されます。

例

```
example_text.htm <apex:sectionHeader title="Salesforce Home" style="background-color: #3366CC; color: white; padding: 5px; font-weight: bold;">Salesforce Home

```

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
text	String	色付きのタイトルバーのすぐ下に表示されるページの説明テキスト。	10.0	global	

	属性名	属性型	説明	必須項目	API バージョン	アクセス
1	helpUrl	String	ページのヘルプファイルの URL。この値が指定されている場合、このページリンクの [ヘルプ] は、色付きのタイトルバーの右側に自動的に表示されます。完全修飾、絶対、または相対 URL を使用する必要があります。JavaScript URL は使用できません。無効な URL を指定すると、ヘルプリンクの代わりに警告のアイコンが表示されます。		10.0	global
d	sectionHeaderId	String	ページの他のコンポーネントが sectionHeader コンポーネントを参照できるようにする識別子。		10.0	global
r	printableUrl	String	印刷可能なビューの URL。		18.0	
r	rendered	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		10.0	global
b	subTitle	String	色付きのタイトルバーのメインタイトルのすぐ下に表示されるテキスト。		10.0	global
l	titleText	String	色付きのタイトルバーの上部に表示されるテキスト。		10.0	global

apex:selectCheckboxes

テーブルに表示される、一連の関連するチェックボックス入力要素です。

このコンポーネントでは、「html」プレフィックスを使用した **HTML パススルー属性**がサポートされています。パススルー属性は、生成されたコンテナタグ `<a>` に適用されます。

例

```

x t en g : < ! -- Pa >- " s a " >
x e ex et e orpld ipcn l< mo a pa = " s a " >
x e ex et e ro m p f < a >
x e ex et e lx e { pb oc kcl & olaus c & a' ! i s " >
x e ex et e t t len{ ep& cl u p} am Oi & a' il s " > /
x e ex et e e lx pb oc kbrc< /as <s > > /
x x tm eBndm w to lt u < taeoe} Ta n vt r tddr s but ia =u ! s "u & " s sa"
x x ro m p f < /a >
x x nou t b d pou < a Pa = " " >
x x nou t tn ct p adet textut tra inas sa =s s a = si .!.. >
x x nou t te nt p fo o < pa a =s " >
x x nou t te nou u b p < a Pa>
x x nou t te nou u b p < a Pa>
x x nou t tn o p a < /a i a s sa>
x x nou u b p < /a Pa>
n or ld r C *** : / *** / 

```

```

    l { l   np clc mo   p i   ss s a
n etr ne g out {} c n   r }   isy [ ] i s =     i S      f
e e e e b l e   e tn t grc       f c   i       Ra           s ( )
} n n ru lli ; 

et e blu t t e l p t { o c g pl mi<sl s o i > s ( )
et e t t l n e a con stoepL t<sp sl no i >o ic s = pl i<s s o i > ;( )
et e @ewno did t p l n o c i sp . a( s o i u U 's, ' 's ')
et e @ewno did t p l n A AcAin sp . Cal( s o i u N , ' ' a a a ')
et e @ewno did t p l n e o cx i sp . MEa t G s o i ( x' o , 'i' ; ' ) 

tn r rm o o p i s;
}

t bln t r et p c g gour] c i is [ i s ( )
et r nru our c

e blu et dn p c our n er { n igv our sc i s ( i s [ i s )
et n othr h n c our c i s . i s = i s;

}

```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
cc	String	selectCheckboxes コンポーネントにフォーカスを置くキーボードのアクセキー。selectCheckboxes コンポーネントにフォーカスがある場合、ユーザはキーボードを使用して、各チェックボックスオプションを選択および選択解除できます。		10.0	global
border	Integer	表示される HTML テーブルの周囲のフレームの幅 (ピクセル単位)。		10.0	global
bordered	Boolean	チェックボックステーブルをラップする の周囲の境界線を表示するか、非表示にするかを制御します。デフォルト値は <code>false</code> (境界線なし) です。		29.0	sa
dir	String	生成された HTML コンポーネントの読み取り方向。使用可能な値には「RTL」(右から左) または「LTR」(左から右) があります。		10.0	global
disabled	Boolean	selectCheckboxes コンポーネントを無効な状態で表示するかどうかを指定する boolean 値。true に設定すると、チェックボックスが無効な状態で表示されます。指定されていない場合、この値はデフォルトの <code>false</code> に設定されます。		10.0	global
disabledClass	String	<code>disabled</code> 属性が <code>true</code> に設定されている場合に、selectCheckboxes コンポーネントの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシ		10.0	global

属性名	属性型	説明	必須項目	API バージョン	アクセス
トを使用するときに適用される CSS スタイルを指定するために使用されます。					
disabled	String	disabled 属性が false に設定されている場合、selectCheckboxes コンポーネントの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。		10.0	global
label	String	ページの他のコンポーネントが selectCheckboxes コンポーネントを参照できるようにする識別子。		10.0	global
disabled	Boolean	ページの項目に関連付けられている入力規則を処理することなく、このコンポーネントに関連付けられているアクションをすぐに実行するかどうかを指定する boolean 値。true に設定すると、アクションがすぐに実行され、入力規則はスキップされます。指定されていない場合、この値はデフォルトの false に設定されます。		11.0	global
label	String	コントロールの横に表示ラベルを表示し、エラーメッセージ内のコントロールを参照できるようにするテキスト値。		23.0	
language	String	「en」または「en-US」など、生成された HTML 出力の基本言語。この属性についての詳細は、 W3C 仕様 を参照してください。		10.0	global
layout	String	チェックボックスをテーブルに表示するメソッド。使用できる値は、チェックボックスを横方向に配置する「lineDirection」、または縦方向に配置する「pageDirection」です。指定されていない場合、この値はデフォルトの「lineDirection」に設定されます。		10.0	global
exampleText	Boolean	凡例テキストを表示するか、非表示にするかを制御します。デフォルト値は true (凡例テキストをすべてのユーザに表示) です。		29.0	sample
exampleText	String	に設定すると、にスタイル属性が追加され、DOM で凡例テキストが保持されますが、表示は画面外に移動します。これにより、テキストが画面に表示されずに入力フィールドからアクセスできるようになります。			
exampleText	String	チェックボックスグループの凡例として表示されるテキスト。境界線が表示されている場合、凡例は境界線の左上の端にはめ込まれます。		29.0	

属性名	属性型	説明	必須項目	API バージョン	アクセス
emptyMessage	String	が空の文字列か、設定されていない場合、凡例は追加されません。			
onblur	String	onblur イベントが発生した場合(フォーカスが selectCheckboxes コンポーネントから離れた場合)に呼び出される JavaScript。	10.0	global	
onChange	String	onchange イベントが発生した場合(selectCheckboxes コンポーネントのチェックボックスの値が変更された場合)に呼び出される JavaScript。	10.0	global	
onClick	String	onclick イベントが発生した場合(ユーザが selectCheckboxes コンポーネントをクリックした場合)に呼び出される JavaScript。	10.0	global	
DoubleClick	String	onclick イベントが発生した場合(ユーザが selectCheckboxes コンポーネントをダブルクリックした場合)に呼び出される JavaScript。	10.0	global	
OnFocus	String	onfocus イベントが発生した場合(selectCheckboxes コンポーネントにフォーカスがある場合)に呼び出される JavaScript。	10.0	global	
OnKeyDown	String	onkeydown イベントが発生した場合(ユーザがキーボードのキーを押した場合)に呼び出される JavaScript。	10.0	global	
OnKeyPress	String	onkeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出される JavaScript。	10.0	global	
OnKeyUp	String	onkeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出される JavaScript。	10.0	global	
OnMouseDown	String	onmousedown イベントが発生した場合(ユーザがマウスボタンをクリックした場合)に呼び出される JavaScript。	10.0	global	
OnMouseMove	String	onmousemove イベントが発生した場合(ユーザがマウスポインタを移動した場合)に呼び出される JavaScript。	10.0	global	
OnMouseOut	String	onmouseout イベントが発生した場合(ユーザが selectCheckboxes コンポーネントからマウスポインタを移動した場合)に呼び出される JavaScript。	10.0	global	
OnMouseOver	String	onmouseover イベントが発生した場合(ユーザが selectCheckboxes コンポーネントにマウスポインタを重ねた場合)に呼び出される JavaScript。	10.0	global	
OnMouseUp	String	onmouseup イベントが発生した場合(ユーザがマウスボタンを放した場合)に呼び出される JavaScript。	10.0	global	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
onSelect	String	onselect イベントが発生した場合(ユーザが selectCheckboxes コンポーネントのチェックボックスをオンにした場合)に呼び出される JavaScript。		10.0	global
readOnly	Boolean	この selectCheckboxes コンポーネントを参照のみとして表示するかどうかを指定する boolean 値。true に設定すると、チェックボックスの値は変更できません。選択されていない場合、この値はデフォルトの false に設定されます。		10.0	global
rendered	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		10.0	global
required	Boolean	この selectCheckboxes コンポーネントが必須項目であるかどうかを指定する boolean 値。true に設定された場合、ユーザは、1つ以上のチェックボックスを選択する必要があります。選択されていない場合、この値はデフォルトの false に設定されます。		10.0	global
style	String	selectCheckboxes コンポーネントの表示に使用されるスタイル。主に、インライン CSS スタイルを追加するために使用されます。		10.0	global
styleClass	String	selectCheckboxes コンポーネントの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。		10.0	global
tabIndex	String	ユーザが Tab キーを繰り返し押したときに、他のページコンポーネントと比較して、この selectCheckboxes コンポーネントが選択される順序。この値は、ユーザが Tab キーを押したときに選択される最初のコンポーネントを 0 として、0 ~ 32767 の整数である必要があります。		10.0	global
title	String	ユーザがコンポーネントにマウスポインタを重ねたときにツールチップとして表示されるテキスト。		10.0	global
value	Object	この selectCheckboxes コンポーネントに関連付けられているコントローラクラス変数を参照する差し込み項目。たとえば、コントローラクラスの関連付けられている変数の名前が myCheckboxSelections である場合、value="={!myCheckboxSelections}" を使用して変数を参照します。		10.0	global

apex:selectList

ユーザが multiselect 属性の値に応じて、1 つの値または複数の値を一度に選択できるようにするオプションのリストです。

このコンポーネントでは、「html-」プレフィックスを使用した HTML パススルー属性がサポートされています。パススルー属性は、生成された `c` タグに適用されます。

例

```

g : < ! -- Pa -->
gorplà pc n l< mo a p a " s a " >
ro m p f < a >
tel cm < a >
len{ ep cl u p} as alu m sv ra" u! c i s" i s = " " >
t l p c < p / as L i s > > /
tit eBndm pto lt u < a > T a n v r t d r s "ut i a =u ! s "u = " s sa"
ro m p f < /a >

nou t b d pou < a Pa = " " >
tn d t p a det text tra ina s sa = g s s a = si .! . >
te nt p m o < pa a = as " >
xt te nou u b p < a Pa>
e e e e e tou lph d c < : X p a < >/
xt te nou u b p < /a Pa>
xt te nou u b p < /a i a s sa>
xt te nou u b p < /a Pa>
g p p < /a a

or ld r C *** : / ***
k { l mp clc mo p i ss a
n etr ne g out {} cn r } i s [ ] i s = i S E
e e e e ble e tn t grc f c i Ra s ( )
t n m ru lh ;
}

et e blu t t e p a t { o c g pl mi< s Oi > s ( )
et t t l t e a con stepL t < s p sl n Oi > o ic s = pl i < s s Oi > ;( )
t en e uno dd t p l n o c i sp . a( s Oi U U 's, ' 's ')
t en e uno dd t p l n A AcAin sp . Cal( s Oi U N ' , ' a a a ')
t tn r rm o o p i s;
}

t et dln t r et p c g gour] C i i s [ i s ( )
t ntr nr u our c
}

e blu et dn p c our n etr { n igvour ] s c i s ( i s [ i s )
t et n odr h n c our c i s . i s = i s i s;
}

```

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
cc	String	selectList にフォーカスを置くキーボードのアクセキー。selectList にフォーカスがあるときに、ユーザはリストのオプションを選択または選択解除できます。		10.0	global
dir	String	生成された HTML コンポーネントの読み取り方向。使用可能な値には「RTL」(右から左) または「LTR」(左から右) があります。		10.0	global
disabled	Boolean	この selectList を無効な状態で表示するかどうかを指定する boolean 値。true に設定すると、selectList が無効な状態で表示されます。指定されていない場合、この値はデフォルトの false に設定されます。		10.0	global
disabledClass	String	disabled 属性が true に設定されている場合に、selectList コンポーネントの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。		10.0	global
disabledText	String	disabled 属性が false に設定されている場合に、selectList コンポーネントの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。		10.0	global
label	String	ページの他のコンポーネントが selectList コンポーネントを参照できるようにする識別子。		10.0	global
labelText	String	コントロールの横に表示ラベルを表示し、エラーメッセージ内のコントロールを参照できるようにするテキスト値。		23.0	
language	String	「en」または「en-US」など、生成された HTML 出力の基本言語。この属性についての詳細は、 W3C仕様 を参照してください。		10.0	global
multiple	Boolean	ユーザがこの selectList から同時に複数のオプションを選択できるかどうかを指定する boolean 値。true に設定すると、ユーザは同時に複数のオプションを選択できます。指定されていない場合、この値はデフォルトの false に設定されます。multiselect が true の場合、value 属性は string[] 型または文字列の list 型である必要があります。そうでない場合、string 型である必要があります。		10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
onblur	String	onblur イベントが発生した場合(フォーカスが selectList コンポーネントから離れた場合)に呼び出される JavaScript。		10.0	global
onchange	String	onchange イベントが発生した場合(selectList コンポーネントの値が変更された場合)に呼び出される JavaScript。		10.0	global
onclick	String	i onclick イベントが発生した場合(ユーザが selectList コンポーネントをクリックした場合)に呼び出される JavaScript。	i	10.0	global
dblclick	String	i onclick イベントが発生した場合(ユーザが selectList コンポーネントをダブルクリックした場合)に呼び出される JavaScript。	i	10.0	global
onfocus	String	s onfocus イベントが発生した場合(selectList コンポーネントにフォーカスがある場合)に呼び出される JavaScript。	s	10.0	global
onkeydown	String	y onkeydown イベントが発生した場合(ユーザがキーボードのキーを押した場合)に呼び出される JavaScript。	y	10.0	global
onkeypress	String	y onkeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出される JavaScript。	y	10.0	global
onkeyup	String	y onkeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出される JavaScript。	y	10.0	global
onmousedown	String	s onmousedown イベントが発生した場合(ユーザがマウスボタンをクリックした場合)に呼び出される JavaScript。	s	10.0	global
onmousemove	String	s onmousemove イベントが発生した場合(ユーザがマウスポインタを移動した場合)に呼び出される JavaScript。	s	10.0	global
onmouseout	String	s onmouseout イベントが発生した場合(ユーザが selectList コンポーネントからマウスポインタを移動した場合)に呼び出される JavaScript。	s	10.0	global
onmouseover	String	sv onmouseover イベントが発生した場合(ユーザが selectList コンポーネントにマウスポインタを重ねた場合)に呼び出される JavaScript。	sv	10.0	global
onmouseup	String	s onmouseup イベントが発生した場合(ユーザがマウスボタンを放した場合)に呼び出される JavaScript。	s	10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
onSelect	String	onSelect イベントが発生した場合(ユーザが selectList コンポーネントのオプションを選択した場合)に呼び出される JavaScript。		10.0	global
readOnly	Boolean	この selectList コンポーネントを参照のみとして表示するかどうかを指定する boolean 値。true に設定すると、リストオプションの選択は変更できません。選択されていない場合、この値はデフォルトの false に設定されます。		10.0	global
rendered	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		10.0	global
required	Boolean	この selectList コンポーネントが必須項目であるかどうかを指定する boolean 値。true に設定された場合、ユーザは、リストオプションを少なくとも 1 つ選択する必要があります。選択されていない場合、この値はデフォルトの false に設定されます。		10.0	global
size	Integer	同時に表示される selectList オプションの数。この数がオプションの合計数より小さい場合、selectList にスクロールバーが表示されます。指定されていない場合、利用できるすべてのオプションが表示されます。		10.0	global
style	String	selectList コンポーネントの表示に使用されるスタイル。主に、インライン CSS スタイルを追加するために使用されます。		10.0	global
styleClass	String	selectList コンポーネントの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。		10.0	global
tabIndex	String	ユーザが Tab キーを繰り返し押したときに、他のページコンポーネントと比較して、この selectList コンポーネントが選択される順序。この値は、ユーザが Tab キーを押したときに選択される最初のコンポーネントを 0 として、0 ~ 32767 の整数である必要があります。		10.0	global
title	String	ユーザがコンポーネントにマウスポインタを重ねたときにツールチップとして表示されるテキスト。		10.0	global
value	Object	この selectList に関連付けられているコントローラクラス変数を参照する差し込み項目。たとえば、コントローラクラスの関連付けられている変数の名前が myListSelections である場合、value="={!myListSelections}" を使用して変数を参照し		10.0	global

属性名	属性型	説明	必須項目	API バー	アクセス
			目	ス	ジョン
ます。multiselect が true の場合、value 属性は string[] 型または文字列の list 型である必要があります。そうでない場合、string 型である必要があります。					

apex:selectOption

el x p oe c & t または p c s > コンポーネントに使用できる値です。
t l n p o c p コンポーネントは、これらのいずれかのコンポーネントの子である必要があります。

このコンポーネントでは、「html-」プレフィックスを使用した HTML パススルー属性がサポートされています。パススルー属性は、p oe c & または d p o c s > <親コンポーネント内>のコンポーネントに対して生成された e ex p タグまたは i p > c 親コンポーネント内>のコンポーネントに対して生成された p タグに適用されます。>

例

```

g : < ! -- Pa >-
gorpl&pc os lora Ca = s " >
ro m p f < a >
t l e pd ce fo <lora dsDen i s z = g s " v a" ls i " $1 = " >
t t e ne epo ce tlepr ed as b Moivdi a" " il a" R " > /
t t e ne p o w eltp aas th boiv i a" i " il W a" i " > /
t t e ne p o c et p u bl ueas b Moivli u a" " il a" " > /
e ex t t l p c < /as L i s >
ro m p f < /a >
g p p < /a >

en or l& r C *** / ***
led l{ p coo lora C i ss a s ;
n e r g bl u i s s = ' ;
t e blit t r{ p cg g g i i s i s ( )
n r ru s;
e blit dt n p & r n rg g i v i s i s ( i s s )
h i s s = s;
}

```

上述の例では次の HTML を表示します。

```

el d e on lora <c & oo izo& s C " =a s " $1 = " >
ene eo l u tpd dn o< o i v p a" " & i/ >
en oe wto le t p t hn< o hio v a" i " >i< i/ >
en & eleutbl p ei elt de li c& l a" o so = sp " >< i/ >
1 c < s / >

```

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
direction	String	i 生成された HTML コンポーネントの読み取り方 向。使用可能な値には「RTL」(右から左) または 「LTR」(左から右) があります。		10.0	global
label	String	i ページの他のコンポーネントが selectOption コン ポーネントを参照できるようにする識別子。		10.0	global
labelDescription	String	o c p D i 開発ツールで使用する、 selectOption コンポーネ ントの説明。	s i	10.0	global
disabled	Boolean	d i s selectOption コンポーネントを無効な状態で表示す るかどうかを指定する boolean 値。 true に設定され ていると、オプションが無効の状態で表示されま す。指定されていない場合、この値はデフォルト の false に設定されます。		10.0	global
itemEscaped	Boolean	m d E c p Boolean s このコンポーネントが生成する HTML 出力で、特 殊な HTML および XML 文字をエスケープするか どうかを指定する boolean 値。 指定されていない場 合、この値はデフォルトの true に設定されます。 たとえば、表示ラベルに「>」記号を追加するに は、記号のエスケープシーケンスを使用して itemEscaped="false" に設定する必要があります。 itemEscaped="false" を指定しない場合、文字エス ケープシーケンスは記述されたとおりに表示され ます。		10.0	global
labelText	String	b m iL このオプションをユーザに表示するために使用さ れる表示ラベル。		10.0	global
labelValue	Object	l mu i ユーザがこのオプションを選択した場合に、サー バに送信される値。		10.0	global
language	String	a 「en」または「en-US」など、生成された HTML 出力の基本言語。この属性についての詳細は、 W3C 仕様 を参照してください。		10.0	global
onClick	String	c c i onclick イベントが発生した場合 (ユーザが selectOption コンポーネントをクリックした場合) に呼び出される JavaScript。		10.0	global
onDoubleClick	String	dbl l c c i onclick イベントが発生した場合 (ユーザが selectOption コンポーネントをダブルクリックした 場合) に呼び出される JavaScript。		10.0	global
onKeyDown	String	no d o k yonkeydown イベントが発生した場合 (ユーザがキー ボードのキーを押した場合) に呼び出される JavaScript。		10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
onkeypress	String	Yonkeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出される JavaScript。	10.0	global	
onkeyup	String	Yonkeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出される JavaScript。	10.0	global	
onmousedown	String	Sonmousedown イベントが発生した場合(ユーザがマウスボタンをクリックした場合)に呼び出される JavaScript。	10.0	global	
onmousemove	String	Sonmousemove イベントが発生した場合(ユーザがマウスポインタを移動した場合)に呼び出される JavaScript。	10.0	global	
onmouseout	String	Sonmouseout イベントが発生した場合(ユーザが selectOption からマウスポインタを移動した場合)に呼び出される JavaScript。	10.0	global	
onmouseover	String	Sonmouseover イベントが発生した場合(ユーザがマウスポインタを selectOption に重ねた場合)に呼び出される JavaScript。	10.0	global	
onmouseup	String	Sonmouseup イベントが発生した場合(ユーザがマウスボタンを放した場合)に呼び出される JavaScript。	10.0	global	
rendered	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	10.0	global	
label	String	Sy この属性は Salesforce API バージョン 17.0 では使用できなくなりました。ページへの影響はありません。	10.0	global	
caption	String	Sy この属性は Salesforce API バージョン 17.0 では使用できなくなりました。ページへの影響はありません。	10.0	global	
title	String	I ユーザがコンポーネントにマウスポインタを重ねたときにツールチップとして表示されるテキスト。	10.0	global	
value	Object	A この selectOption コンポーネントに関連付けられている、SelectItem 型のコントローラクラス変数を参照する差し込み項目。たとえば、コントローラクラスの関連付けられた変数の名前が myOption である場合、value="{'!myOption}" を使用して変数を参照します。	10.0	global	

apex:selectOptions

このコンポーネントは、`l`、`kased`、`p`、`exc`、`t`、`s`、`as`、`pr`、`ica`、`コンポーネント`に使用できる値のコレクションです。`n`、`po`、`c`、`p`、`コンポーネント`は、これらのいずれかのコンポーネントの子である必要があります。また、カスタム Visualforce コントローラの `selectOption` オブジェクトのコレクションにバインドされている必要があります。

このコンポーネントでは、「`html-`」プレフィックスを使用した HTML パススルー属性がサポートされています。`x`、`p`、`o`、`e`、`c`、`t`、`d`、`p`、`o`、`c`、`s`、`<親コンポーネント内`のコンポーネントに対して生成された`e`、`ex`、`p`、`タグ`または`i`、`p`、`c`、`親コンポーネント内`のコンポーネントに対して生成された`p`、`タグ`に適用されます。

例

```

g : < ! -- Pa >-
gorplä pc n l< mo a pa =s a " >
ro m p f < a >
e lxe { pb oc ktlc m te toas e c l s on a" kbur i s" i =s a " y
t t len{ ep& cl u p} as Oi s a" il s" > /
e l x pb oc kbr< /as <s > > /
x tit eBndem wto lt u < e m taeoe e} Ta n vt r tdatr s'ut ia =u ! s "u =s " s sa"
x ro m p f < /a >
xt tn o t p a det teut tra ina s sa =s s a =s si .". >
e ex te nt p m o < pa a =s " >
e xt te nou u bp p < a Pa>
e e e e tou lph d c< : X p a < >/
e xt te nou u bp p < /a a>
e ex t tn o p a < /a i a s sa>
x te nou u bp p < /a Pa>
g p p < /a >

n or ld r C *** : / ***
keu {l mp clc m o p i ss a
n str ne g out {} cn r } isy [ ] i s = i S E
e e e e b le e tn t grc f c i Ra s ( )
n n ru lh ; 

ste e blit t e ht t { o c g d mi< s s Oi > s ( )
ste t t l nt e n con stoep t< sp Sl nOi > o ic s =L i< s s Oi > ;( )
en e wno dd t p l n o c i sp . a( s Oi U 'S, 'S ')
en e wno dd t p l n A AcAin sp . C a( s Oi U N , ' a a a ')
en e wno dd t p l n e o cx i sp . Mba t o s Oi ( X ' O , ' i ' ; ' )
tn r rm o o p i s;
t obin t r et p c g gour] C i is [ i s ( )
ntr nru our c
e blit dn p c our n er { n igvour sc i s ( i s [ i s )
et n othr h n c our c i s . i s = i s; [ i s )
}

```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
d	String	i	ページの他のコンポーネントが selectOptions コンポーネントを参照できるようにする識別子。	10.0	global
r d r d	Boolean		コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	10.0	global
l u	Object	a	この selectOptions コンポーネントに関連付けられている、SelectItem 型の集合となるコントローラクラス変数を参照する差し込み項目。たとえば、コントローラクラスの関連付けられている変数の名前が mySetOfOptions である場合、value="{!mySetOfOptions}" を使用して変数を参照します。	はい 10.0	global

apex:selectRadio

テーブルに表示される、一連の関連するラジオボタン入力要素です。チェックボックスとは異なり、一度に選択できるラジオボタンは 1 つのみです。

このコンポーネントでは、「html-」プレフィックスを使用した [HTML パススルー属性](#)がサポートされています。パススルー属性は、生成されたコンテナタグに適用されます。[a>](#)

例

```

g : < ! -- Pa >-
gorpld pcn l< mo a pa = s a " >
ro m pf < a >
el{ d potlc m our asc R i va a" ! "y>
t t len{ ep& cl up am Oi s a il s" > /
t l d po c < p/as R i< a> > / 
tt edBnemm upto lt{u < n te oe} Tcan tr r tda st'ut ia a ! s u = " s se"
ro m pf < /a >
nou tu p d pou < a Pa = " >
tn d t p a det textut tra ina s sa = s s a = si .".>
te nt p fn o < pa a = as " >
te nou u p p < a Pa>
e e e e tou lphd c< : x py a < >/
x t x t x e phu u p tl mp < our ac v a" ! "y > /
te nou u p p < /a a> Pa>
tn o p a < /a i a s sa>
nou u p p < /a >
g p p < /a >
n or ld r C *** / *** / 
{1 mp clc mo p i ss s a y ;
n t rn gur clu i s Ra s ( )
e e e eble e tn t grc f c i Ra s ( )

```

```

    n n ru li ;
}

ete bl t t e lpt { o c g d mi<sl s > s ( )
ete t t lnt e n con stoed t<sp sl nOi >o ic s =pl i<s s Oi > ;( )
en @ewno dd t p l n o c i sp . a( s Oi U u 's, 's' )
en @ewno dd t p l n A AcAin sp . Cal( s Oi N , ' ' a a a' )
en @ewno dd t p l n e o cx i esp .t MEa l@tn s @m o ( a' o , 'ip' ; ' ) i s;

t e blt r { npcg gour c i is ; y( )
ntr nru our c

e bl t dnt pc ou m t{ rrC t giurv ti cns tourm y(iCS ojur c y ) i s . y ; y
}

```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
cc	String	ラジオボタンにフォーカスを置くキーボードのアクセスキーコード。ラジオボタンにフォーカスがあるときに、ユーザはラジオボタンの値を選択または選択解除できます。		10.0	global
border	Integer	表示される HTML テーブルの周囲のフレームの幅(ピクセル単位)。		10.0	global
bordered	Boolean	ラジオボタンテーブルをラップするの周囲の境界線を表示するか、非表示にするかを制御します。デフォルト値は false (境界線なし)です。		< 29.0	s a
dir	String	生成された HTML コンポーネントの読み取り方向。使用可能な値には「RTL」(右から左)または「LTR」(左から右)があります。		10.0	global
disabled	Boolean	selectRadio コンポーネントを無効な状態で表示するかどうかを指定する boolean 値。true に設定されている場合、ラジオボタンが無効な状態で表示されます。指定されていない場合、この値はデフォルトの false に設定されます。		10.0	global
disabledClass	String	disabled 属性が true に設定されている場合に、selectRadio コンポーネントの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。		10.0	global
disabledStyle	String	disabled 属性が false に設定されている場合に、selectRadio コンポーネントの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。		10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
d	String	i ページの他のコンポーネントが selectRadio コンポーネントを参照できるようにする識別子。		10.0	global
d mm	Boolean	i ページの項目に関連付けられている入力規則を処理することなく、このコンポーネントに関連付けられているアクションをすぐに実行するかどうかを指定する boolean 値。 true に設定すると、アクションがすぐに実行され、入力規則はスキップされます。 指定されていない場合、この値はデフォルトの false に設定されます。		11.0	global
l b l	String	a コントロールの横に表示ラベルを表示し、エラーメッセージ内のコントロールを参照できるようにするテキスト値。		23.0	
l g	String	a 「en」または「en-US」など、生成された HTML 出力の基本言語。この属性についての詳細は、 W3C仕様 を参照してください。		10.0	global
l ou	String	ya ラジオボタンをテーブルに表示するメソッド。 使用できる値は、ラジオボタンを横方向に配置する「lineDirection」、またはラジオボタンを縦方向に配置する「pageDirection」です。 指定されていない場合、この値はデフォルトの「lineDirection」に設定されます。		10.0	global
nl d g bl I	Boolean	v i 凡例テキストを表示するか、非表示にするかを制御します。 デフォルト値は f (凡例テキストをすべてのユーザに表示) です。		29.0	s a
el r e e t c		r a e e t c に設定すると、 にスタイル属性 が追加され、 DOM で凡 例テキストが保持されますが、表示は画面外に移 動します。これにより、テキストが画面に表示さ れずにスクリーンリーダーからアクセスできるよ うになります。	>		
l d g	T String	ラジオボタングループの凡例として表示されるテキスト。 境界線が表示されている場合、凡例は境界線の左上の端にはめ込まれます。 が T 空の文字列か、設定されていない場合、凡例は追加されません。		29.0	
bl ru	String	onblur イベントが発生した場合 (フォーカスが selectRadio コンポーネントから離れた場合) に呼び出される JavaScript。		10.0	global
o g dn	String	onchange イベントが発生した場合 (selectRadio コンポーネントのラジオボタンの値が変更された場合) に呼び出される JavaScript。		10.0	global

	属性名	属性型	説明	必須項目	APIバージョン	アクセス
ol	c_c	String	i onclick イベントが発生した場合(ユーザが selectRadio コンポーネントをクリックした場合)に呼び出される JavaScript。		10.0	global
dbl_l	c_c	String	i onclick イベントが発生した場合(ユーザが selectRadio コンポーネントをダブルクリックした場合)に呼び出される JavaScript。		10.0	global
o_o_u	f	String	s onfocus イベントが発生した場合(selectRadio コンポーネントにフォーカスがある場合)に呼び出される JavaScript。		10.0	global
no_d_o	k	String	y onkeydown イベントが発生した場合(ユーザがキーボードのキーを押した場合)に呼び出される JavaScript。		10.0	global
o_r	p_k	String	y onkeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出される JavaScript。		10.0	global
o_u	pk	String	y onkeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出される JavaScript。		10.0	global
w_on_oud_o	String	s onmousedown イベントが発生した場合(ユーザがマウスボタンをクリックした場合)に呼び出される JavaScript。			10.0	global
o_o_um_o_m	String	s vonmousemove イベントが発生した場合(ユーザがマウスポインタを移動した場合)に呼び出される JavaScript。			10.0	global
o_o_um_o_u	String	s onmouseout イベントが発生した場合(selectRadio コンポーネントからマウスポインタを移動した場合)に呼び出される JavaScript。			10.0	global
o_o_um_o_r	String	s vonmouseover イベントが発生した場合(ユーザが selectRadio コンポーネントにマウスポインタを重ねた場合)に呼び出される JavaScript。			10.0	global
o_o_um_u_p	p	String	s onmouseup イベントが発生した場合(ユーザがマウスボタンを放した場合)に呼び出される JavaScript。		10.0	global
o_l	c	String	s onselect イベントが発生した場合(ユーザが selectRadio コンポーネントのラジオボタンを選択した場合)に呼び出される JavaScript。		10.0	global
r_d_d		Boolean	a この selectRadio コンポーネントを参照のみとして表示するかどうかを指定する boolean 値。true に設定すると、選択されたラジオボタンを変更できなくなります。選択されていない場合、この値はデフォルトの false に設定され、選択されたラジオボタンを変更できなくなります。		10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
rendered	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		10.0	global
selected	Boolean	この selectRadio コンポーネントが必須項目であるかどうかを指定する boolean 値。true に設定されている場合、ユーザはラジオボタンを選択する必要があります。選択されていない場合、この値はデフォルトの false に設定されます。	i	10.0	global
style	String	selectRadio コンポーネントの表示に使用される CSS スタイル。主に、INLINE CSS スタイルを追加するために使用されます。	s	10.0	global
styleClass	String	selectRadio コンポーネントの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。	s	10.0	global
TabIndex	String	ユーザが Tab キーを繰り返し押したときに、他のページコンポーネントと比較して、この selectRadio コンポーネントが選択される順序。この値は、ユーザが Tab キーを押したときに選択される最初のコンポーネントを 0 として、0 ~ 32767 の整数である必要があります。	i	10.0	global
title	String	ユーザがコンポーネントにマウスポインタを重ねたときにツールチップとして表示されるテキスト。	i	10.0	global
value	Object	この selectRadio コンポーネントに関連付けられているコントローラクラス変数を参照する差し込み項目。たとえば、コントローラクラスの関連付けられている変数の名前が myRadioButtonSelection である場合、value="={!myRadioButtonSelection}" を使用して変数を参照します。	a	10.0	global

apex:stylesheet

Visualforce ページでコンポーネントにスタイルを適用するために使用できるスタイルシートへのリンクです。指定されている場合、このコンポーネントは、生成された HTML ページの head 要素にスタイルシートの参照を挿入します。

このコンポーネントでは、「html-」プレフィックスを使用した **HTML パススルー属性**がサポートされています。パススルー属性は、生成された `<tr>` タグに適用されます。 >

例

上述の例では次の HTML を表示します。

```
<el ee r t < el e t txk < ie e p ts g s h 'b u f y e' o c s/ c cs / s ss/ $/ a ss' >
```

Zip リソースの例

```
e t e { l p le h e < et as o uys & c bl ! R F & scc Z Syi , $' a ss " ')> /
```

上述の例では次の HTML を表示します。

```
<el ee r t < el e t txk < ie e e p ts g s h 'g df dy="b s/ c [ a $/ a ss' > /
```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
id	String	ページの他のコンポーネントがスタイルシートコンポーネントを参照できるようにする識別子。		10.0	global
l u	Object	スタイルシートファイルのURL。これは静的リソースへの参照にすることができます。	はい	10.0	global

apex:tab

このコンポーネントは、单一のタブです。コンポーネントは、子である必要があります。

このコンポーネントでは、「html-」プレフィックスを使用した **HTML パススルー属性**がサポートされています。パススルー属性は、タブのコンテンツをラップする生成された `<a>` タグに適用されます。>

例

```
et e g : < ! -- Pa >- dg p p g h < a ia =! 'Pa> xe n tw ke lep te ene & dh alp nal Paest c e yel im Zos lh =! & 'ai =! a Pa > xe e bdnb lnp n t<en rel t ba na't Oe' e mbl bx 'a& E' o > p <a /a xe b lnp a t<m wIrel t ba 2 na't o't we #bT bx 'ai oE' o > p <a /a tx e n b l p < /a a Pa> g p p < /a >
```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
selected	Boolean	タブを選択して参照できるかどうかを指定する boolean 値。true に設定すると、タブを選択できません。指定されていない場合、この値はデフォルトの false に設定されます。		10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
contentId	String	s タブのコンテンツが表示されているときにフォーカスされる子コンポーネントの ID。		10.0	global
label	String	i ページの他のコンポーネントがタブコンポーネントを参照できるようにする識別子。		10.0	global
disabled	Boolean	i ページの項目に関連付けられている入力規則を処理することなく、このコンポーネントに関連付けられているアクションをすぐに実行するかどうかを指定する boolean 値。true に設定すると、アクションがすぐに実行され、入力規則はスキップされます。指定されていない場合、この値はデフォルトの false に設定されます。		11.0	global
labelText	String	a タブのヘッダーに表示されるテキスト。		10.0	global
labelWidth	String	a タブのヘッダーの長さ (ピクセル単位)。指定されていない場合、この値はデフォルトの表示ラベルのテキストの幅に設定されます。		10.0	global
name	Object	a タブの名前。この属性の値を使用して、tabPanel でデフォルトで選択されるタブを指定します。		10.0	global
onClick	String	i onclick イベントが発生した場合 (ユーザがタブをクリックした場合) に呼び出される JavaScript。		10.0	global
onComplete	String	oncomplete イベントが発生した場合 (タブが選択されており、ページにそのコンテンツが表示されている場合) に呼び出される JavaScript。		10.0	global
onDoubleClick	String	i onclick イベントが発生した場合 (ユーザがタブをダブルクリックした場合) に呼び出される JavaScript。		10.0	global
onFocus	String	y onkeydown イベントが発生した場合 (ユーザがキーボードのキーを押した場合) に呼び出される JavaScript。		10.0	global
onKeyPress	String	y onkeypress イベントが発生した場合 (ユーザがキーボードのキーを押したか、押したままにした場合) に呼び出される JavaScript。		10.0	global
onKeyUp	String	y onkeyup イベントが発生した場合 (ユーザがキーボードのキーを放した場合) に呼び出される JavaScript。		10.0	global
onMouseDown	String	s onmousedown イベントが発生した場合 (ユーザがマウスボタンをクリックした場合) に呼び出される JavaScript。		10.0	global
onMouseMove	String	s onmousemove イベントが発生した場合 (ユーザがマウスポインタを移動した場合) に呼び出される JavaScript。		10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
onmouseout	String	onmouseout イベントが発生した場合(ユーザがタブからマウスポインタを移動した場合)に呼び出される JavaScript。		10.0	global
onmouseover	String	onmouseover イベントが発生した場合(ユーザがタブにマウスポインタを重ねた場合)に呼び出される JavaScript。		10.0	global
onmouseup	String	onmouseup イベントが発生した場合(ユーザがマウスボタンを放した場合)に呼び出される JavaScript。		10.0	global
ontabenter	String	ontabenter イベントが発生した場合(フォーカスがタブコンポーネントに置かれた場合)に呼び出される JavaScript。		11.0	global
ontableave	String	ontableave イベントが発生した場合(フォーカスがタブの外側にあるコンポーネントに置かれた場合)に呼び出される JavaScript。		11.0	global
rendered	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		10.0	global
results	Object	R AJAX 更新要求の結果がクライアントに返されるときに再作成する 1 つ以上のコンポーネントの ID。この値には、単一の ID、ID のカンマ区切りのリスト、または ID のリストまたはコレクションの差し込み項目の式を使用できます。また、この値は、switchType 属性の値が「ajax」である場合にのみ適用できます。		10.0	global
status	String	s sa AJAX 更新要求の状況を表示する関連付けられているコンポーネントの ID。「actionStatus コンポーネント」を参照してください。この値は、switchType 属性の値が「ajax」に設定されている場合にのみ適用できます。	s sa	10.0	global
style	String	s y タブコンポーネントのすべての部分の表示に使用されるスタイル。主に、インライン CSS スタイルを追加するために使用されます。	s y	10.0	global
styleClass	String	s y タブコンポーネントのすべての部分の表示に使用される CSS スタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。	s y	10.0	global
switchMethod	String	si このタブに切り替えるための実装メソッド。使用できる値には、「client」、「server」、および「ajax」があります。指定されていない場合、この値はデフォルトの「server」に設定されます。指定	si	10.0	global

属性名	属性型	説明	必須項目	API バージョン	アクセス
switchTab					
om	Integer i	Ajax 更新要求がタイムアウトするまでの時間(ミリ秒)。この値は、switchType 属性の値が「ajax」に設定されている場合にのみ適用できます。	10.0	global	
l	String i	ユーザがコンポーネントにマウスオーバーしたときにツールチップとして表示されるテキスト。	10.0	global	

apex:tabPanel

タブのセットとして表示されるページの領域です。ユーザがタブのヘッダーをクリックすると、タブの関連コンテンツが表示され、他のタブのコンテンツは非表示になります。

このコンポーネントでは、「html-」プレフィックスを使用した **HTML パススルー属性** がサポートされています。パススルー属性は、すべてのタブを含む生成された `<a>` タグに適用されます。

単純な例

```

g : < ! -- Pa >-
dg p p g h < a ia ='' 'Pa>
x e n w ke le p te ene & dh alP nal P&st c e yd' in &os lh ='' & "ai ='' 
tx e e bdnb lrep n t<m nal t ba na't Oe'' e &bl bx 'a& f'' O& > p
tx e bdnb lrep o t<m wTnal t ba 2 na't d't ve &T bx 'ai o'' & > p
tx e n b l p < /a a Pa>
g p p < /a >

```

高度な例

```

e ex ro te e l m & dr p B&l i se ope p an o e ec li &oy s scia V p is a
et l d h et r n cd i c v a ha a i U L R .
ro l m t ept f F t &te i q llt 00 000R003h 5 g daIDb h , s: U L R s
e he n erap xe fc sc m &sa i s a /a I y Ral 00 000R003 5
e e ei hro e l &c V t Sdt i ts & &curo l &ot&Tm info & a f i a i i a .
ex w l wen &t &ph o !uh -i s bnlet ma sd sc tt &oslc t r k a s a i & a ass a i a s
t tg d lauh t f oge t babtrfhet on &seiy ht & dha N o f i a s y i i i s ,
don er ogn enc en go kter: en &taeo oir dt a di &luhi mfgv i h h a i a
d loqpn f oem l osc &dln &utwp .Y &tm g as a v i h cy i a .ss s ys . >-
t n tegndprd por h &neC ew au &co dar &r u ='' " & a" " >
e ne n b & fl < Tp D c --i a a .ss s ys >-
{ 1 < s / y >
b bn r oj Tdoro B &kwt : laro a c &on :& ognBnE ; &go k i : ; a i - a
b &on r oj&T dor&l &clarg : g a h o r o b c &on : & ogn nc &ky go k : ; a a i - a
1 < s / y >
e r n b C l <Tp ! -- a a a >-
x e n w ke le p te ene & dh alP nal P&st c t y &it &us b ccl ='' a Tai ='' a Pa
et bl et G b c &u cass=a &caT b iac ' & a Tass=a ' & a & a
tx e e bdnb lrep n t<m nal t ba na't Oe'' e &bl bx 'a& f'' O& > p
tx e bdnb lrep o t<m wTnal t ba 2 na't d't ve &T bx 'ai o'' & > p

```

tabPanel

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
selectedTabClass	String	tabPanelが選択されたときのタブのヘッダーの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。		10.0	global
selectedTabContentClass	String	tabPanelコンポーネントのタブのコンテンツの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。		10.0	global
selectedTabStyle	String	tabPanelコンポーネントのタブのコンテンツの表示に使用されるスタイル。主に、インライン CSS スタイルを追加するために使用されます。		10.0	global
direction	String	生成された HTML コンポーネントの読み取り方向。使用可能な値には「RTL」(右から左) または「LTR」(左から右) があります。		10.0	global
disabledTabClass	String	tabPanelが無効にされたときのタブのヘッダーの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。		10.0	global
headerAlign	String	タブのヘッダーが整列する基準となる tabPanel の側面。使用できる値には、「left」または「right」があります。指定されていない場合、この値はデフォルトの「left」に設定されます。		10.0	global
headerClass	String	選択されているかどうかに関係なく、すべてのタブヘッダーの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。		11.0	global
headerGap	String	隣り合わせの 2 つのタブヘッダー間の距離(ピクセル単位)。指定されていない場合、この値はデフォルトの 0 に設定されます。		10.0	global
height	String	タブバーの高さ。利用可能な縦方向のスペースのパーセント(height="50%" など)、またはピクセル数(height="200px" など)のいずれかで表されます。指定されていない場合、この値はデフォルトの 100% に設定されます。		10.0	global
id	String	ページの他のコンポーネントが tabBar コンポーネントを参照できるようにする識別子。		10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
disabled	Boolean	i ページの項目に関連付けられている入力規則を処理することなく、このコンポーネントに関連付けられているアクションをすぐに実行するかどうかを指定する boolean 値。true に設定すると、アクションがすぐに実行され、入力規則はスキップされます。指定されていない場合、この値はデフォルトの false に設定されます。		11.0	global
headerClass	String	i tabPanel が選択されていないときのタブのヘッダーの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。		10.0	global
language	String	a 「en」または「en-US」など、生成された HTML 出力の基本言語。この属性についての詳細は、 W3C仕様 を参照してください。		10.0	global
onClick	String	i onclick イベントが発生した場合(ユーザが tabPanel をクリックした場合)に呼び出される JavaScript。		10.0	global
onDoubleClick	String	i ondblclick イベントが発生した場合(ユーザが tabPanel をダブルクリックした場合)に呼び出される JavaScript。		10.0	global
onKeyDown	String	y onkeydown イベントが発生した場合(ユーザがキーボードのキーを押した場合)に呼び出される JavaScript。		10.0	global
onKeyPress	String	y onkeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出される JavaScript。		10.0	global
onKeyUp	String	y onkeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出される JavaScript。		10.0	global
onMouseDown	String	s onmousedown イベントが発生した場合(ユーザがマウスボタンをクリックした場合)に呼び出される JavaScript。		10.0	global
onMouseMove	String	s onmousemove イベントが発生した場合(ユーザがマウスポインタを移動した場合)に呼び出される JavaScript。		10.0	global
onMouseOut	String	s onmouseout イベントが発生した場合(ユーザが tabPanel コンポーネントからマウスポインタを移動した場合)に呼び出される JavaScript。		10.0	global
onMouseOver	String	s onmouseover イベントが発生した場合(ユーザが tabPanel コンポーネントにマウスポインタを重ねた場合)に呼び出される JavaScript。		10.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
onmouseup	String	イベントが発生した場合(ユーザがマウスボタンを放した場合)に呼び出されるJavaScript。		10.0	global
rendered	Boolean	コンポーネントをページに表示するかどうかを指定するboolean値。指定されていない場合、この値はデフォルトのtrueに設定されます。		10.0	global
requestId	Object	AJAX更新要求の結果がクライアントに返されるときに再作成される1つ以上のコンポーネントのID。この値には、単一のID、IDのカンマ区切りのリスト、またはIDのリストまたはコレクションの差し込み項目の式を使用できます。この値は、switchType属性が「ajax」に設定されている場合にのみ適用されます。		10.0	global
selectedTab	Object	ページが読み込まれたときに、デフォルトで選択されるタブの名前。この値は、子のタブコンポーネントのname属性と一致している必要があります。value属性が定義されている場合、selectedTab属性は無視されます。		10.0	global
style	String	tabPanelコンポーネントの表示に使用されるスタイル。主に、INLINE CSSスタイルを追加するために使用されます。		10.0	global
styleClass	String	tabPanelコンポーネントの表示に使用されるスタイルクラス。主に、外部CSSスタイルシートを使用するときに適用されるCSSスタイルを指定するために使用されます。		10.0	global
tabClientSide	String	タブ間の切り替えに使用される実装メソッド。使用できる値には、「client」、「server」、および「ajax」があります。指定されていない場合、この値はデフォルトの「server」に設定されます。		10.0	global
tabServerSide	String	tabPanelコンポーネントの表示に使用されるスタイルクラス。主に、外部CSSスタイルシートを使用するときに適用されるCSSスタイルを指定するために使用されます。		10.0	global
toolTipText	String	ユーザがコンポーネントにマウスポインタを重ねたときにツールチップとして表示されるテキスト。		10.0	global
value	Object	現在有効なタブ。式を使用してこれを指定し、有効なタブを動的に制御できます。たとえば、value="={!TabInFocus}"とします。TabInFocusはカスタムコントローラによって設定される変数です。この属性の値は、selectedTabに設定されている値より優先されます。		10.0	global

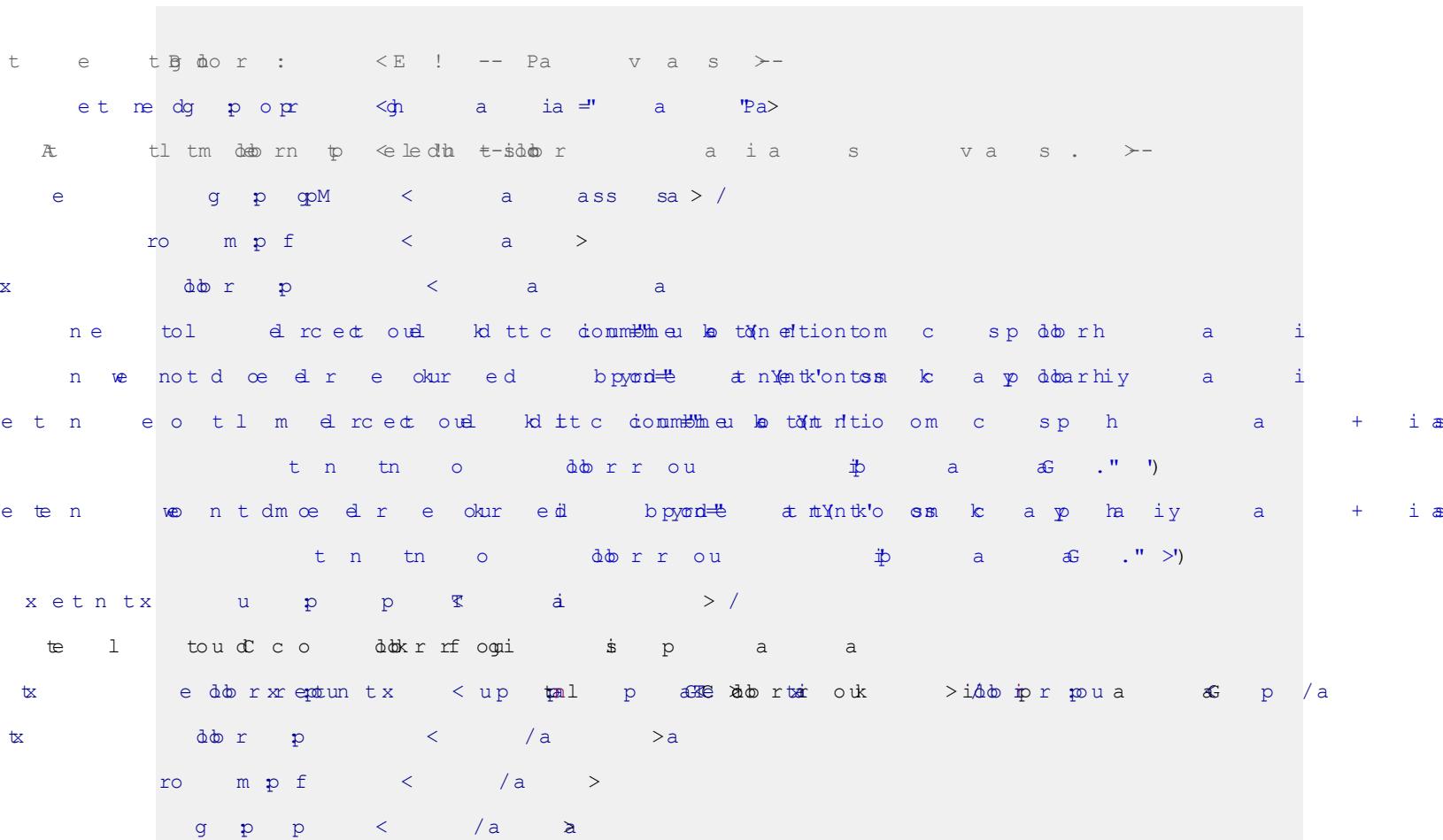
属性名	属性型	説明	必須項目	API バージョン	アクセス
width	String	i tabPanel の幅。利用可能な横方向のスペースのパーセント (width="50%" など)、またはピクセル数 (width="800px" など) のいずれかで表されます。指定されていない場合、この値はデフォルトの 100% に設定されます。	10.0	global	

apex:toolbar

子コンポーネントをいくつでも含められるスタイル設定された横方向のツールバーです。デフォルトでは、すべての子コンポーネントはツールバーの左側に整列されます。1つ以上の子コンポーネントを右に整列するには、`align="right"` を指定します。

例

e	g l m ðø r	ɸ	! T -- Pas	a	a>-
e t e	dg p p g h <	a ia ="	'Pa>		
A ex	l mt l m Ø < ðø ðr f-Ø	a	a	a	a >-
	et ðø r dp ðø r < h a T i a ="	" à			
x t et	t x e ou ue p l up	xt m ððar p v T æ" Sa	" a> /		
tx	e ðø r r ðø u e m n ðø l a pd	ðø inr ou s a æ" i ði =" k Læ i s" >			
e x t t	enou utt p l wep	ø ðkal E prå v o ðc c //s sa	. " >		
e e	l ro fc	s sa			
e x t t	nou u p p <	ka L i >			
e x t t	enou utt p el uep	ø d ða ok pi v ro p æ" o ðc vc// s sa	. " >		
e e ex e e	d ml or p ro p	av k			
e x t t	nou u p p <	ka L i >			
tx	ðø r r ðø u < p /a	æ >			
tx	e ðø r r ðø u e m n ðø l a p to	ðø ir c ðd a ððar r' ou rå a = mi ði ="	æ F		
e x	et ro d m p fro	h < a i =" F " >			
e x et n tx	et u et p t x e p t e	ø I á r p et ðit x" uT p" > p < T /á >			
e x	n eno edmm p l u et s	d k a n ð noia d m C æ's "ai =" k L ia " > /			
e x	ro m p f <	/a >			
tx	ðø r r ðø u < p /a	æ >			
x	ðø r p < /a	>a			
	g p p < /a	à			



属性

	属性名	属性型	説明	必須項目	APIバージョン	アクセス
o	l	c	String		10.0	global
o	lc		String		10.0	global
g	gh	h	String	i	10.0	global
d			String	i	10.0	global

	属性名	属性型	説明	必須項目	APIバージョン	アクセス
m	ro	p	String i	ツールバーのコンポーネントを区切るために使用されるシンボル。使用できる値には、「none」、「line」、「square」、「disc」、および「grid」があります。指定されていない場合、この値はデフォルトの「none」に設定されます。	10.0	global
ol	c	c	String i	onclick イベントが発生した場合(ユーザがツールバーをクリックした場合)に呼び出される JavaScript。	16.0	
dbl l	c	c	String i	ondblclick イベントが発生した場合(ユーザがツールバーをダブルクリックした場合)に呼び出される JavaScript。	16.0	
o l m	c	c	String k i	ユーザが toolbarGroup コンポーネントに含まれないツールバーのコンポーネントをクリックした場合に呼び出される JavaScript。	16.0	
o dbl ln	c	c	String ki	ユーザが toolbarGroup コンポーネントに含まれないツールバーのコンポーネントをダブルクリックした場合に呼び出される JavaScript。	16.0	
w o n d m o			String i	ユーザが toolbarGroup コンポーネントに含まれないツールバーのコンポーネントでキーボードのキーを押したときに呼び出される JavaScript。	16.0	
o m		p	String i	ユーザが toolbarGroup コンポーネントに含まれないツールバーの項目でキーボードのキーを押したとき、または押したままにしたときに呼び出される JavaScript。	16.0	
o mu			String i	ユーザが toolbarGroup コンポーネントに含まれないツールバーの項目でキーボードのキーを放したときに呼び出される JavaScript。	16.0	
w o n o u m o			String i	ユーザが toolbarGroup コンポーネントに含まれないツールバーの項目でマウスボタンをクリックしたときに呼び出される JavaScript。	16.0	
o o u m o m			String i	toolbarGroup コンポーネントに含まれないツールバーの項目がフォーカスされているときに、ユーザがマウスポインタを移動した場合に呼び出される JavaScript。	16.0	
t o o u m o u			String i	ユーザが toolbarGroup コンポーネントに含まれないツールバーの項目からマウスポインタを移動したときに呼び出される JavaScript。	16.0	
o ou m o r			String i	ユーザが toolbarGroup コンポーネントに含まれないツールバーの項目にマウスポインタを重ねたときに呼び出される JavaScript。	16.0	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
onmousedown	String	ユーザが toolbarGroup コンポーネントに含まれないツールバーの項目でマウスボタンを放したときに呼び出される JavaScript。		16.0	
onkeydown	String	onkeydown イベントが発生した場合(ユーザがキーボードのキーを押した場合)に呼び出される JavaScript。		16.0	
onkeypress	String	onkeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出される JavaScript。		16.0	
onkeyup	String	onkeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出される JavaScript。		16.0	
onmousedown	String	onmousedown イベントが発生した場合(ユーザがマウスボタンをクリックした場合)に呼び出される JavaScript。		16.0	
onmousemove	String	onmousemove イベントが発生した場合(ユーザがマウス pointer を移動した場合)に呼び出される JavaScript。		16.0	
onmouseout	String	onmouseout イベントが発生した場合(ユーザがツールバーからマウス pointer を移動した場合)に呼び出される JavaScript。		16.0	
onmouseover	String	onmouseover イベントが発生した場合(ユーザがツールバーにマウス pointer を重ねた場合)に呼び出される JavaScript。		16.0	
onmouseup	String	onmouseup イベントが発生した場合(ユーザがマウスボタンを放した場合)に呼び出される JavaScript。		16.0	
showToolbar	Boolean	ページにツールバーを表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	10.0	global	
role	String	ツールバーコンポーネントの区切りの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。「itemSeparator 属性」も参照してください。	10.0	global	
style	String	ツールバーの表示に使用されるスタイル。主に、インライン CSS スタイルを追加するために使用されます。	10.0	global	
class	String	ツールバーの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。	10.0	global	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
h	String	ツールバーの幅。画面の合計の幅の相対パーセント(width="5%"など)、またはピクセル単位の絶対数(width="10px"など)として表されます。指定されていない場合、この値はデフォルトの100%に設定されます。	10.0	global	

apex:toolbarGroup

ツールバーの左または右に整列できる、ツールバー内のコンポーネントのグループです。
コンポーネントは子コンポーネントである必要があります。

例

```
<apex:toolbarGroup id="group1" position="left">
    <apex:button value="Left Button" />
    <apex:button value="Left Button" />
</apex:toolbarGroup>
```

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
name	String	ページの他のコンポーネントが toolbarGroup コンポーネントを参照できるようにする識別子。	10.0	global	
separator	String	toolbarGroup でツールバーコンポーネントを区切るために使用するシンボル。使用できる値には、「none」、「line」、「square」、「disc」、および「grid」があります。指定されていない場合、この値はデフォルトの「none」に設定されます。	10.0	global	
position	String	ツールバーの toolbarGroup の位置。使用できる値には、「left」または「right」があります。指定さ	10.0	global	

属性名	属性型	説明	必須項目	API バージョン	アクセス
左側の要素をクリックした場合に呼び出される JavaScript					
onleftclick	String	onclick イベントが発生した場合(ユーザが toolbarGroup をクリックした場合)に呼び出される JavaScript。		11.0	global
ondblclick	String	ondblclick イベントが発生した場合(ユーザが toolbarGroup をダブルクリックした場合)に呼び出される JavaScript。		11.0	global
ondown	String	keydown イベントが発生した場合(ユーザがキーボードのキーを押した場合)に呼び出される JavaScript。		11.0	global
onpress	String	onkeypress イベントが発生した場合(ユーザがキーボードのキーを押したか、押したままにした場合)に呼び出される JavaScript。		11.0	global
onkeyup	String	onkeyup イベントが発生した場合(ユーザがキーボードのキーを放した場合)に呼び出される JavaScript。		11.0	global
onmousedown	String	onmousedown イベントが発生した場合(ユーザがマウスボタンをクリックした場合)に呼び出される JavaScript。		11.0	global
onmousemove	String	onmousemove イベントが発生した場合(ユーザがマウスポインタを移動した場合)に呼び出される JavaScript。		11.0	global
onmouseout	String	onmouseout イベントが発生した場合(ユーザが toolbarGroup コンポーネントからマウスポインタを移動した場合)に呼び出される JavaScript。		11.0	global
onmouseover	String	onmouseover イベントが発生した場合(ユーザが toolbarGroup コンポーネントにマウスポインタを重ねた場合)に呼び出される JavaScript。		11.0	global
onmouseup	String	onmouseup イベントが発生した場合(ユーザがマウスボタンを放した場合)に呼び出される JavaScript。		11.0	global
display	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		10.0	global
role	String	ツールバー・コンポーネントの区切りの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。「itemSeparator 属性」も参照してください。		10.0	global

属性名	属性型	説明	必須項目	API バージョン	アクセス	
	String	s	y	ツールバーのグループの表示に使用される CSS スタイル。主に、インライン CSS スタイルを追加するため使用されます。	10.0	global
C	String	s	y	ツールバーのグループの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。	10.0	global

apex:variable

コンポーネントの本文内の指定された式の代わりに使用できるローカル変数です。bl p を使用し va i a て、ページ内で繰り返される冗長な式の数を減らします。

注意: `bl x tpe` は `ble wa ita ax` または `a p a apa` などの反復コンポーネント内の再割り当てをサポートしていません。カウンタとして `bl p` の増分などを行った結果については、サポートまたは定義されていません。

例

e ex ro te e l m o d r p B l i s o o p e p n o e ec h poy s s f c a V p i s a
t etnl d h o e t r n c rd c i c v a ha a i U L R .
ro l m t ept f tF tn B taoi,1M t c00 000B0035 g ddDb a h , s: U L R s
e he n ecap xe fc sc m /d s a i s a /a I y B a l 00 000B0035
e e el hro e l fc V t Sdt ipa B secur o l s o t G Tm info mo a f i a i
i a
g : < ! -- Pa >-
x te n go e p l d p c r n b l o a a C " i a a " >
x e r k l { p r t ltn c } va i acva c=" a v " ! " a > /
t { n r te g p } c < m > G i s p ! . s Na < a . > /
g p p < / a a
n or l d r C *** / *** /
n b l e l { p r n b l o C i s w a i a a /
n t t a o c c c c a ; a
t et h l t to t n p c C g o c C á a a ()
t t n { on f c l h c i (á =)
t t n e e o t tec l c t d e n w o m af srC L c s N a a
e ex d A e te negr ntpe e t g e = g t rm Ba . g d Pa (.)] Pa a s (.) i (' ; ')
} n r t n ru o c c ; a

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
d	String i	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。	10.0	global	
r d r d	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	10.0	global	
l u	Object ^a	変数コンポーネントの本文内で変数によって表現できる式。	はい	10.0	global
r	String ^v	変数コンポーネントの本文内で値の式を表現するために使用できる変数の名前。	はい	10.0	global

apex:vote

サポートするオブジェクトの投票コントロールを表示するコンポーネントです。

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
d	String i	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。	14.0	global	
bo jd	c I	投票するオブジェクトの識別子。	はい	26.0	
r d r d	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	14.0	global	
r r dr	String	アクションが実行されたときに更新されるページの領域。		29.0	

chatter:feed

レコードでは Chatter EntityFeed、ユーザでは UserProfileFeed を表示します。Force.com サイトの Visualforce ページでは Chatter コンポーネントは使用できません。このコンポーネントを使用するページには、バージョン 3 より前の Ext JS を含めることはできません。

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
d	I id i	フィードを表示するレコードのエンティティ ID。例: Contact.Id。	はい	20.0	
d	mf I String	Chatter EntityFeed または UserProfileFeed を絞り込むフィード項目種別。許容値についての詳細は、「Object Reference for Salesforce and Force.com」の「FeedItem」(Type の下) を参照してください。		20.0	
d	String i	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
o ol m	C p String	投稿またはコメントをフィードに追加した後にコードする JavaScript 関数。		20.0	
r d r d	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global
r d r	Object R	action メソッドの結果がクライアントに返されるときに再作成される 1 つ以上のコンポーネントの ID。この値には、単一の ID、ID のカンマ区切りのリスト、または ID のリストまたはコレクションの差し込み項目の式を使用できます。		20.0	
o bli	hr h Boolean s	Chatter パブリッシャーを表示します。アーカイブ済みグループでは、パブリッシャーは指定された値に関係なく非表示になります。		20.0	

chatter:feedWithFollowers

レコードの Chatter フィードおよびフォロワーのリストを表示する統合された UI コンポーネントです。Force.com サイトの Visualforce ページでは Chatter コンポーネントは使用できません。このコンポーネントを使用するページには、バージョン 3 より前の Ext JS を含めることはできません。`m p f` タグ内にこのコンポーネントを指定しないでください。

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
d	I id i	フィードを表示するレコードのエンティティ ID。例: Contact.Id。	はい	20.0	
d	String i	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
ol_m	C p	String	AJAX 更新要求の結果がクライアントで完了したときに呼び出される JavaScript。	20.0	
dr_d		Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	14.0	global
dr_r		Object R	action メソッドの結果がクライアントに返されるときに再作成される 1 つ以上のコンポーネントの ID。この値には、単一の ID、ID のカンマ区切りのリスト、または ID のリストまたはコレクションの差し込み項目の式を使用できます。	20.0	
dr_h		Boolean	UI タグ、[表示]/[非表示] ボタン、[フォローする]/[フォロー解除] ボタンを含むメタバーのヘッダーを表示する。	20.0	

chatter:follow

Chatter レコードをフォローまたはフォローを解除するボタンを表示します。Force.com サイトの Visualforce ページでは Chatter コンポーネントは使用できません。このコンポーネントを使用するページには、バージョン 3 より前の Ext JS を含めることはできません。

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
id	I	id i	【フォローする】または【フォロー解除】ボタンを表示するレコードのエンティティ ID。例: Contact.Id。	はい	20.0
		String i	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。	14.0	global
ol_m	C p	String	イベントのフォローまたはフォロー解除の完了後にコールする JavaScript 関数。	20.0	
dr_d		Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	14.0	global
dr_r		Object R	action メソッドの結果がクライアントに返されるときに再作成される 1 つ以上のコンポーネントの ID。この値には、単一の ID、ID のカンマ区切りのリスト、または ID のリストまたはコレクションの差し込み項目の式を使用できます。	20.0	

chatter:follower

レコードの Chatter フォロワーのリストを表示します。Force.com サイトの Visualforce ページでは Chatter コンポーネントは使用できません。このコンポーネントを使用するページには、バージョン 3 より前の Ext JS を含めることはできません。

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
d	I id i	フォロワーのリストを表示するレコードのエンティティ ID。例: Contact.Id。	はい	20.0	
d	String i	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
r d r d	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global

chatter:newsfeed

現在のユーザの Chatter ニュースフィードを表示します。Force.com サイトの Visualforce ページでは Chatter コンポーネントは使用できません。このコンポーネントを使用するページには、バージョン 3 より前の Ext JS を含めることはできません。

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
d	String i	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
o ol m	C p String	投稿またはコメントをフィードに追加した後にコードする JavaScript 関数。		24.0	
r d r d	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global
r d r	Object R	action メソッドの結果がクライアントに返されるときに再作成される 1 つ以上のコンポーネントの ID。この値には、単一の ID、ID のカンマ区切りのリスト、または ID のリストまたはコレクションの差し込み項目の式を使用できます。		24.0	

chatter:userPhotoUpload

ユーザの写真を Chatter プロファイルページにアップロードします。このコンポーネントを使用するには、組織で Chatter を有効にする必要があります。ユーザは、標準ユーザ、ポータルユーザ、大規模ポータルユーザ、または Chatter 外部ユーザのプロファイルに属している必要があります。

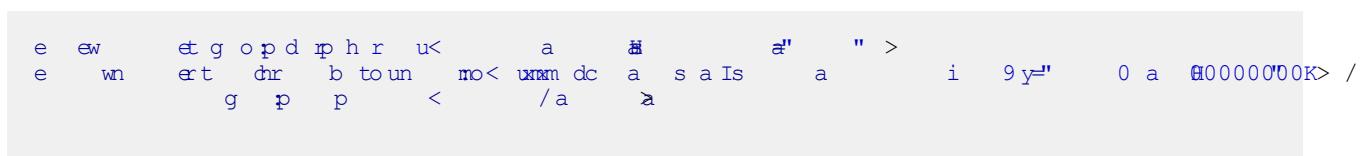
属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
componentLabel	String	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
display	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global
isLarge	Boolean	デフォルトのトリミングされた形式ではなく、元の形式で写真を表示します。		28.0	

chatteranswers:aboutme

ユーザの写真、ユーザ名、[私の設定を編集] リンク、[サインアウト] リンクが含まれる Chatter アンサーのプロファイルボックス。このプロファイルボックスには、認証されたユーザのみがアクセスできます。Chatter アンサーユーザ向けにカスタマイズした操作を作成するために、他の Chatter アンサーコンポーネントと一緒に使用します。

この例では、Chatter アンサーの aboutme コンポーネントが表示されます。



属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
displayZone	String	フィードを表示するゾーン。	はい	29.0	
componentLabel	String	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global

属性名	属性型	説明	必須項目	APIバージョン	アクセス
noSignInFlag	Boolean	フィードのサインオンオプションを無効にするフラグ。		29.0	
showComponent	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global

chatteranswers:allfeeds

フィード、フィルタ、プロファイル、および[サインアップ] および[サインイン] ボタンを含む Chatter アンサー アプリケーションを表示します。このコンポーネントを使用するページには、バージョン 3 より前の Ext JS を含めることはできません。

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
language	String	記事の取得に使用する言語。		24.0	
zone	String	フィードを表示するゾーン。	はい	24.0	
of	String	「AllQuestions」、 「UnansweredQuestions」、 「UnsolvedQuestions」、 「SolvedQuestions」、「MyQuestions」、 「MostPopular」、「DatePosted」、 「RecentActivity」のいずれかのオプションを、[Q&A] フィードで条件として選択できます。		24.0	
useAoBmdirC	Boolean	この属性は Salesforce API バージョン 29.0 では使用できなくなりました。ページへの影響はありません。		24.0	
parentComponentId	String	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
version	String	JavaScript API バージョン		24.0	
signInFlag	Boolean	フィードのサインオンオプションを無効にするフラグ。		24.0	
showComponent	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない		14.0	global

属性名	属性型	説明	必須項目	API バージョン	アクセス
url_rewrite	Boolean	場合、この値はデフォルトの true に設定されます。			
		サイト URL 書き換え機能に基づいて URL を書き換えるフラグ。		24.0	

chatteranswers:changepassword

Chatter アンサーのパスワードの変更ページを表示します。このコンポーネントを使用するページには、バージョン 3 より前の Ext JS を含めることはできません。

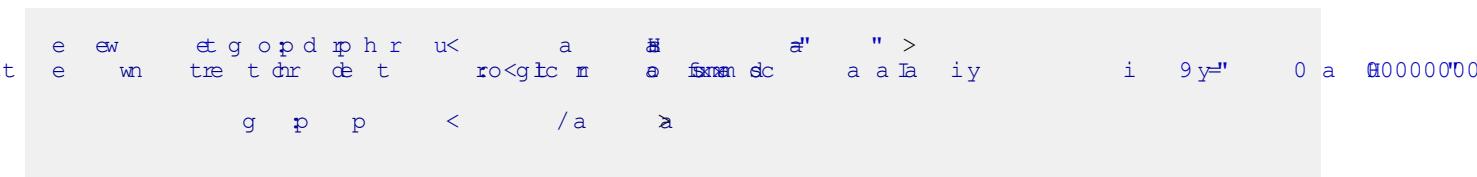
属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
id	String	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
rendered	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global

chatteranswers:datacategoryfilter

ユーザがフィードをデータカテゴリで絞り込みできるようにする、Chatter アンサーのデータカテゴリ条件です。Chatter アンサーユーザ向けにカスタマイズした操作を作成するために、他の Chatter アンサーコンポーネントと一緒に使用します。

次の例は、Chatter アンサーの datacategoryfilter コンポーネントを表示します。



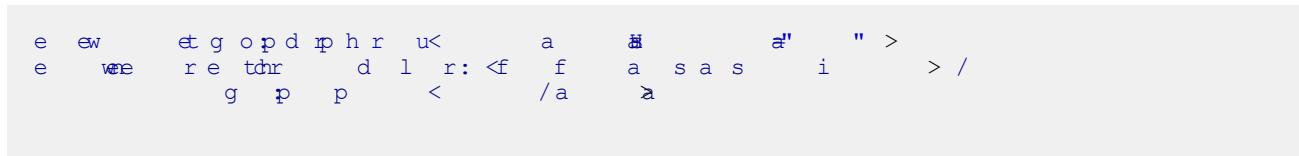
属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
feedZone	String	フィードを表示するゾーン。	はい	29.0	
componentId	String	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
display	Boolean	コンポーネントをページに表示するかどうかを指定するboolean値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global

chatteranswers:feedfilter

フィード条件を使用して、Chatter アンサーに表示されるフィードの並び替えと絞り込みを行うことができます。

次の例は、Chatter アンサーの feedfilter コンポーネントを表示します。



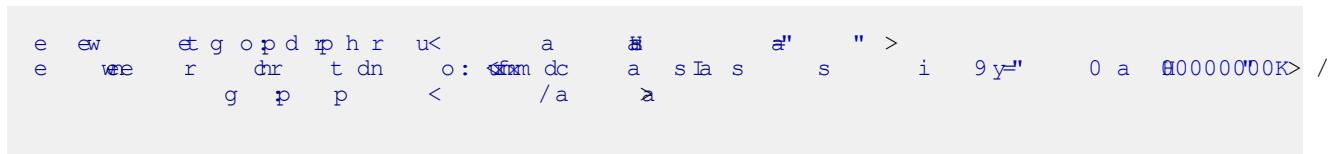
属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
filterOptions	String	Chatter アンサー検索条件に表示されるオプションには、「AllQuestions」、「UnansweredQuestions」、「UnsolvedQuestions」、「SolvedQuestions」、「MyQuestions」、「MostPopular」、「DatePosted」、「RecentActivity」を指定できます。		29.0	
componentId	String	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
display	Boolean	コンポーネントをページに表示するかどうかを指定するboolean値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global

chatteranswers:feeds

ユーザがゾーン内で質問と記事を参照して質問に対する返信を投稿できるようにする、Chatter アンサーフィードです。Chatter アンサーウェブ向けにカスタマイズした操作を作成するために、他の Chatter アンサー コンポーネントと一緒に使用します。

次の例は、Chatter アンサーの feeds コンポーネントを表示します。



属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
language	String	記事の取得に使用する言語。		29.0	
zone	String	フィードを表示するゾーン。	はい	29.0	
componentId	String	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
version	Double	JavaScript API バージョン。		29.0	
signed	Boolean	フィードのサインオンオプションを無効にするフラグ。		29.0	
display	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global
urlRewriting	Boolean	サイト URL 書き換え機能に基づいて URL を書き換えるフラグ。		29.0	

chatteranswers:forgotpassword

Chatter アンサーのパスワードを忘れた場合のページを表示します。このコンポーネントを使用するページには、バージョン 3 より前の Ext JS を含めることはできません。

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
d	String i	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。	14.0	global	
r d r d	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	14.0	global	
u r l r r	Boolean s R	サイト URL 書き換え機能に基づいて URL を書き換えるフラグ。	24.0		

chatteranswers:forgotpasswordconfirm

Chatter アンサーのパスワードの確認ページを表示します。このコンポーネントを使用するページには、バージョン 3 より前の Ext JS を含めることはできません。

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
d	String i	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。	14.0	global	
r d r d	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	14.0	global	
u r l r r	Boolean s R	サイト URL 書き換え機能に基づいて URL を書き換えるフラグ。	24.0		

chatteranswers:guestsignin

Chatter アンサーの [サインイン] および [サインアップ] ボタンです。これらのボタンは、ゲストユーザのみがアクセスできます。Chatter アンサーユーザ向けにカスタマイズした操作を作成するために、他の Chatter アンサー コンポーネントと一緒に使用します。

次の例は、Chatter アンサーの guestsignin コンポーネントを表示します。

```
e ex e ew et g o p d p h r u< a & " " >
e ett e wne rt dr n m g : & a s a s s $ i > /
e ex g p p < / a &
```

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
d	String i	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。	14.0		global
r d r d	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	14.0		global
u rl r	Boolean s R	サイト URL 書き換え機能に基づいて URL を書き換えるフラグ。	29.0		

chatteranswers:help

お客様に Chatter アンサーのヘルプページ (FAQ) を表示します。

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
d	String i	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。	14.0		global
r d r d	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	14.0		global

chatteranswers:login

Chatter アンサーのサインインページを表示します。このコンポーネントを使用するページには、バージョン 3 より前の Ext JS を含めることはできません。

属性

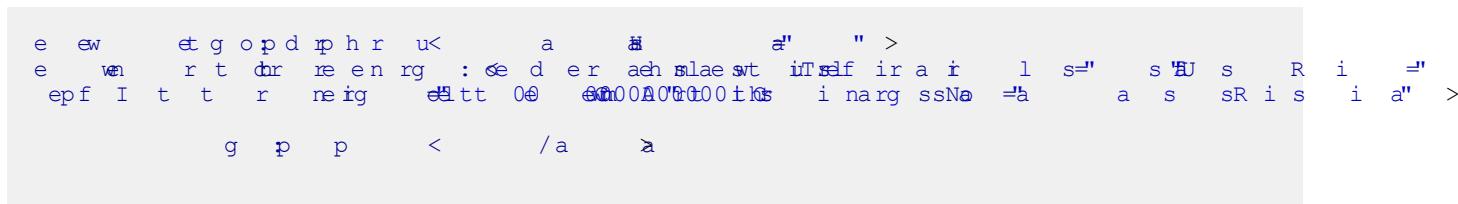
属性名	属性型	説明	必須項目	APIバージョン	アクセス
d	String i	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。	14.0		global

属性名	属性型	説明	必須項目	API バージョン	アクセス
display	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global
url_rewrite	Boolean	サイト URL 書き換え機能に基づいて URL を書き換えるフラグ。		24.0	

chatteranswers:registration

Chatter アンサーの登録ページを表示します。

次の例は、Chatter アンサーの registration コンポーネントを表示します。



属性

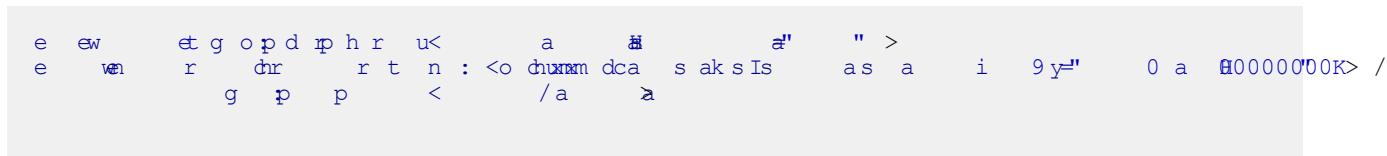
属性名	属性型	説明	必須項目	API バージョン	アクセス
display	Boolean	[契約条件] セクションを非表示にするフラグ。		24.0	
display_name	String	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
profile_id	id	このコンポーネントに Salesforce コミュニティからアクセスする場合は、セルフ登録したユーザのプロファイル ID。このプロファイルは、Salesforce コミュニティサイト登録でのみ使用され、スタンダードアロンの Force.com サイト登録では使用されません。		24.0	
registration_class	String	ChatterAnswers.AccountCreator Apex インターフェースを実装する Apex クラスの名前。使用されていない場合、Chatter アンサー登録では、生成される ChatterAnswers または ChatterAnswersRegistration のいずれかの Apex クラスが使用されます。		24.0	
display	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global

属性名	属性型	説明	必須項目	API バージョン	アクセス
url_rewrite	Boolean	サイト URL 書き換え機能に基づいて URL を書き換えるフラグ。		24.0	

chatteranswers:searchask

ユーザがゾーン内で質問と記事を検索して質問できるようにする、検索バーとボタンです。Chatter アンサーユーザー向けにカスタマイズした操作を作成するために、他の Chatter アンサー コンポーネントと一緒に使用します。

次の例は、Chatter アンサーの searchask コンポーネントを表示します。



属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
feed_id	String	フィードを表示するゾーン。	はい	29.0	
parent_id	String	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
signed_out	Boolean	フィードのサインオンオプションを無効にするフラグ。		29.0	
show_feed	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global
language	String	記事の取得に使用する言語。		29.0	
url_rewrite	Boolean	サイト URL 書き換え機能に基づいて URL を書き換えるフラグ。		29.0	

chatteranswers:singleitemfeed

1つのケースおよび質問の Chatter アンサーフィードを表示します。このコンポーネントを使用するページには、バージョン 3 より前の Ext JS を含めることはできません。

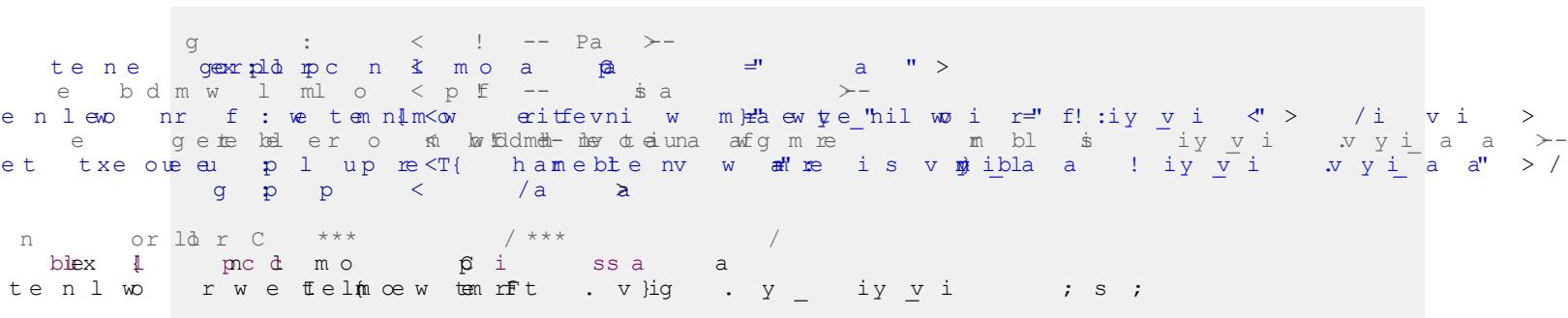
属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
d	I id	i フィードを表示するケースのエンティティ ID。	はい	24.0	
d	String i	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
r d r d	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global

flow:interview

このコンポーネントではページに Flow インタビューを埋め込みます。

例



属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
b u n o o	o c	String L ナビゲーションボタンが表示されるページブロックの領域。使用可能な値には、「top」、「bottom」、または「both」があります。指定されていない場合、この値はデフォルトの「both」に設定されます。		21.0	
b u o o l	String	コマンドボタンに適用されるスタイル(省略可能)。CSS クラスではなく、インラインスタイルでのみ使用できます。		21.0	
n o	s h c	ApexPagesPageReference フローの完了時にフローが移動する場所の特定に使用できる PageReference。		21.0	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
d	String	i	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。	14.0	global
r	Flow.Interview	i	FlowInterview の表現に使用できるオブジェクト。	21.0	
m	String	a	フローの一意の名前。	はい	21.0
r d r d	Boolean		コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	14.0	global
r r d r	Object		action メソッドの結果がクライアントに返されるときに再作成される 1 つ以上のコンポーネントの ID。この値には、単一の ID、ID のカンマ区切りのリスト、または ID のリストまたはコレクションの差し込み項目の式を使用できます。	21.0	
o l	h	p Boolean	は	ヘルプリンクを表示するかどうかを指定します。	21.0

ideas:detailOutputLink

アイデアを表示するページへのリンクです。注: このコンポーネントを使用するには、salesforce.com の担当者に問い合わせ、組織でアイデアの拡張標準コントローラを有効化するように依頼してください。

アイデア標準コントローラを使用する detailOutputLink コンポーネント

```

e   ex ro te e   l m m o d r p t E l i s o p e p m o e e ec u p y s s g c a V p a i s
e   e   l d h d t r n r d d v h a a i U L R .
ro   l m t e p t e f F e t e d a i , t h 0 0 0 0 0 0 3 i S g i d D b a , s : U L R s
e   h e n e r o p x e f c s c m / g s a i s a / a I / y R a l 0 0 0 0 0 0 0 3 5
e   e e e e l h r o e l b c V t S a t i p s a c u r o l s o t q T m i n f o a f i a i
t   e   dg l : g < ! -- Pa i a Pa >-
t   n t e g n d p r e p o r l a k C d a a a a = " " > a
x   t e t B e l g e p k p d n < k l o a c a i = " a S " >
e   t d t d d n e u : u t e < q y l s p a i k g @ L i = a i a ' P a
d { d e d t e l t e d b i t d a h d i l a l u i u i a < p i / s a i k @ L i >
br   br < < > / > /
x t e t { t x e o u u p e d p b d e t t x a o u u p > i l p . a T Y / a >
ex   B l g o p p c < k / a a >
x   t e t B e l g o p t e l p n t c o < m m k C a o a c i = " s S " >
e   t t d e { p r e l t a m o a l m m i t a s d = l a v a a ! L i s " i = " i s " >
ex   t n o B } m m d o c ! . a y
x t   t d p < / a L i a s a >
ex   B l g o p p c < k / a a >
g p p < / a a >

```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
d	String	i	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。	14.0	global
d d	I	String	i 表示するアイデアの ID。	はい	29.0
g	p	ApexPage	PageReference URL を出力リンクで使用する Visualforce ページ。このページは標準コントローラを使用する必要があります。	はい	29.0
d r m p	Integer	Na	アイデア詳細ページのコメントの表示対象のページ番号(ページあたり 50 個)。たとえば、コメントが 100 個ある場合は、pageNumber="2" では 51 ~ 100 番目のコメントが表示されます。	29.0	
g	p ff	Integer	@ 現在のページからの目的のページのオフセット。pageNumber が指定されている場合は、pageOffset 値は使用されません。pageNumber および pageOffset が両方とも設定されていない場合、結果として得られるリンクに指定ページが含まれないため、コントローラはデフォルトの最初のページに設定されます。	29.0	
r d r d	Boolean		コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	14.0	global
l	String	s y	detailOutputLink コンポーネントの表示に使用されるスタイル。主に、インライン CSS スタイルを追加するために使用されます。	29.0	
l l	C	String	s y detailOutputLink コンポーネントの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。	29.0	

ideas:listOutputLink

アイデアのリストを表示するページへのリンクです。注: このコンポーネントを使用するには、salesforce.com の担当者に問い合わせ、組織でアイデアの拡張標準コントローラを有効化するように依頼してください。

アイデア標準リストコントローラを使用する listOutputLink コンポーネント

```
te      lg : g < ! -- Pai s Pa >-
t n tegndp red poer ld se C dt are ead t a cad   " aV      S =ia      a " a
x      B l go p p c < k a a >
d t td t l ntaeu t era sp i te s iake lgL t i p g   " c      =ai s      'Pa R
d t td t l n u a < ps ia / s iaks OL i >
```

```

t t d t l ntu tu e ro <ps ia / s iaks OL
d t t d t l ntu u re <ps ia / s iaks OL
n o ex xt ex xt ex
    t t d t l ntu ea t roen <ps i / s iaks OL
        B l go p p c < k/a a >
        B l go p p c < k a a >
    t e d e pl tu d p < t aL dasd &" il i sa "v "la a" a>a
    txe du ue p lt up t d<Td } a l v &" il a a " >/
    t d p < /aL iasa >
        B l go p p c < k/a a >
        g p p < /a a >

```

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
ro g c	String	a アイデアリストの表示対象のカテゴリ。		29.0	
o umm dc	I	i アイデアを表示するゾーンの ID。communityID が設定されていない場合、ゾーンはユーザがアクセスできる有効なゾーン（デフォルト）に設定されます。ユーザに複数のゾーンへのアクセス権がある場合は、アルファベット順で先頭にくる名前のゾーンが使用されます。		29.0	
d g p	String i	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
g p	ApexPagesPageReference	URL を出力リンクで使用する Visualforce ページ。はい このページではセット指向の標準コントローラを使用する必要があります。	はい	29.0	
phi r m p	Integer	Na アイデアリストの表示対象のページ番号(ページあたり 20 個)。たとえば、アイデアが 100 個ある場合は、pageNumber="2" では 21 ~ 40 番目のアイデアが表示されます。		29.0	
g p ff	Integer	e 現在のページからの目的のページのオフセット。pageNumber が指定されている場合は、pageOffset 値は使用されません。pageNumber および pageOffset が両方とも設定されていない場合、結果として得られるリンクに指定ページが含まれないため、コントローラはデフォルトの最初のページに設定されます。		29.0	
r d r d	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global
ro	String s	アイデアリストでの並び替え順序。使用できる値には、「popular」、「recent」、「top」、および「comments」があります。		29.0	

属性名	属性型	説明	必須項目	API バージョン	アクセス
u	String	s sa アイデアリストの表示対象の状況。		29.0	
A r b u c	Boolean	k si y このコンポーネントがこのリンクを含むページで使用される communityId、sort、category、および status の値を再利用する必要があるかどうかを指定する boolean 値。		29.0	
l	String	s y listOutputLink コンポーネントの表示に使用されるスタイル。主に、インライン CSS スタイルを追加するために使用されます。		29.0	
l l C	String	s y listOutputLink コンポーネントの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。		29.0	

ideas:profileListOutputLink

ユーザのプロファイルを表示するページへのリンクです。注: このコンポーネントを使用するには、salesforce.com の担当者に問い合わせ、組織でアイデアの拡張標準コントローラを有効化するように依頼してください。

アイデア標準リストコントローラを使用する profileListOutputLink コンポーネント

e e	g ol : opf < ! -- Pa i Pa >-				
t n	t egnd p oer l d se C dt a re e at t a cad # " aV S =ia a " a				
x	B l go p p c < k a a >				
l e	d t r dl : n t p e u e t a ps ea L li ksc O a e ep s lt p n op R i s" c =a i PaR				
bu	d t p dl : n p R u i < s i /ps a L iks OL i >				
ex	d t t dl : n t p fu < e io ps a le i ksgOLol ip s op f i s" la =a i Pa> s a				
x	d t t dl : n t p fu < i /ps a L iks OL i >				
ex	B l go p p c < k/a a >				
xt	t e d e pl tu d p < t aL da s d a" iL i sa "v =ia a" a>a				
ext	t x e fu ue p lt up t d < Td } a l v a" il a a a " > /				
ex	B l go p p c < k/a a >				
	g p p < /a a >				

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
o umm dc	I String	i アイデアを表示するゾーンの ID。communityID が設定されていない場合、ゾーンはユーザがアクセス		29.0	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
d	String i	スできる有効なゾーン(デフォルト)に設定されます。ユーザに複数のゾーンへのアクセス権がある場合は、アルファベット順で先頭にくる名前のゾーンが使用されます。		14.0	global
g	ApexPagesPageReference p	URL を出力リンクで使用する Visualforce ページ。このページではセット指向の標準コントローラを使用する必要があります。	はい	29.0	
pr	Integer m	アイデアリストの表示対象のページ番号(ページあたり 20 個)。たとえば、アイデアが 100 個ある場合は、pageNumber="2" では 21 ~ 40 番目のアイデアが表示されます。		29.0	
g	Integer ff	現在のページからの目的のページのオフセット。pageNumber が指定されている場合は、pageOffset 値は使用されません。pageNumber および pageOffset が両方とも設定されていない場合、結果として得られるリンクに指定ページが含まれないため、コントローラはデフォルトの最初のページに設定されます。		29.0	
r d r d	Boolean d	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global
ro	String s	アイデアリストでの並び替え順序。使用できる値には、「ideas」、「votes」、および「recentReplies」があります。		29.0	
A r b	Boolean uc	y:このコンポーネントがこのリンクを含むページで使用されている userId、communityId および並び替えの値を再利用する必要があるかどうかを指定する boolean 値。		29.0	
l	String s	profileListOutputLink コンポーネントの表示に使用されるスタイル。主に、インライン CSS スタイルを追加するために使用されます。	y	29.0	
l l	String s	profileListOutputLink コンポーネントの表示に使用されるスタイルクラス。主に、外部 CSS スタイルシートを使用するときに適用される CSS スタイルを指定するために使用されます。	y	29.0	
ur d	I	プロファイルが表示されているユーザの ID。		29.0	

knowledge:articleCaseToolbar

ケースの詳細ページから記事を開く場合に使用される UI コンポーネントです。このコンポーネントでは現在のケース情報を表示し、ユーザがケースに記事を添付できるようにします。

このコンポーネントを使用する「FAQ」カスタム記事タイプテンプレートの例

```

t      n tegndprd por ld &C a e &d b ra al k Q="Ff v " & =a s'a >
wn te ded rg l k &C T ia sa a
{ n redr de n ere t eC g $rm &d! } clu I Pa. a a s . s a =
e { e de n c erI t eC g $rms} a=p d! c I Pa. a a s . s a "
e { re l de n c erI t C g &rm d=p ! Pa. a a si. " > /
at h } l bcl kT h F v _ ia < 1 < > / 
g p p < /a >

```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
c_I	String	現在の記事の ID。	はい	29.0	
s_a	String	現在のケースの ID。	はい	29.0	
i	String	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
c_C	Boolean	このコンポーネントに CSS を含める必要があるかどうかを指定する。デフォルトは true です。		29.0	
	Boolean	コンポーネントをページに表示するかどうかを指定する。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global

knowledge:articleList

記事の絞り込み済みリストのループです。このコンポーネントは同一ページで最大 4 回使用できます。各データカテゴリに指定できる条件は 1 つのみで、次のような標準項目のみにアクセスできます。

- ID (文字列): 記事の ID
- Title (文字列): 記事のタイトル
- Summary (文字列): 記事の概要
- urlName (文字列): 記事の URL 名
- articleTypeName (文字列): 記事タイプの開発者名
- articleTypeLabel (文字列): 記事タイプの表示ラベル
- lastModifiedDate (日付): 最終更新日
- firstPublishedDate (日付): 初回公開日
- lastPublishedDate (日付): 最終公開日

リンクのHTMLリストとして「phone」カテゴリの最もよく参照される上位10の記事を表示するknowledge:articleListの例。「phone」は「products」カテゴリグループに含まれます。

```

te      nou tu   p l   qu bl< o   a   c       Pa k   ya="   " >
e  ewn  lu     te   d t   rge l   k   r   rcl   iLa   ics v ia   ="a ia "
e  et e  to gec r donu   opc :h   p a i   s="   s   "
t      et Bro w   o m   d   s   V="y   si   "
e  ee   z   g   p   ia l S="   0
              >
r      e {   l   r   t   h an en   <de   &   awg! R F &   d i   . K   vi   i   ,
e  ewn  l}   l   c i a l i.   "   ! ia   i   <   <   >/&   >
e  lu     te   d t   rg l   k   c<   /   iLa   i s   >
te      nou u   p   p   <   /a>   Pa>

```

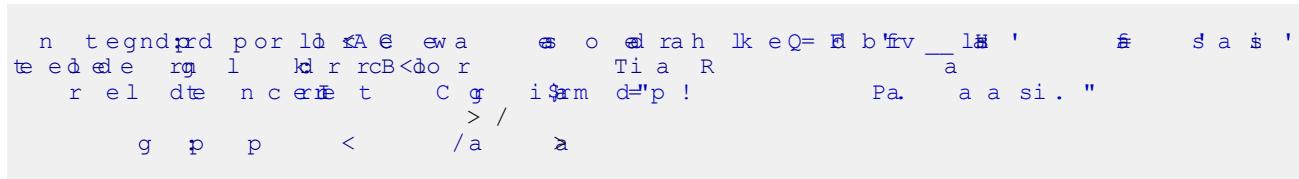
属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
list	c	String	記事タイプで絞り込みできる記事リスト。	29.0	
listName	c	String	articleListコンポーネントの本文に記事オブジェクトを表すために使用できる変数の名前。	29.0	
category	c	String	データカテゴリで絞り込みできる記事リスト。	29.0	
addToList	M	String	リストに記事を追加できるかどうかを指定するboolean変数名。	29.0	
componentId	i	String	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。	14.0	global
keyword	k	String	検索がnullでない場合の検索キーワード。keyword属性が指定されている場合、結果はキーワードの関連性で並び替えられ、sortBy属性は無視されます。	29.0	
language	g	String	記事の取得に使用する言語。	29.0	
pageNumber	p	Integer	現在のページ番号。	29.0	
pageSize	p	Integer	一度に表示される記事数。1ページに表示する記事の総数が200を超えることはできません。	29.0	
renderComponent	d	Boolean	コンポーネントをページに表示するかどうかを指定するboolean値。指定されていない場合、この値はデフォルトのtrueに設定されます。	14.0	global
sortOrder	s	String	記事リスト「mostViewed」、「lastUpdated」、および「title」に適用される並び替え値。keyword属性が指定されている場合、sortBy属性は無視されます。	29.0	

knowledge:articleRendererToolbar

記事のヘッダーツールバーを表示します。このツールバーには、投票に応じた数の星、Chatter フィード、言語選択リストおよびプロパティパネルがあります。このコンポーネントを使用するページには、バージョン 3 より前の Ext JS を含めることはできません。

カスタムレンダラツールのツールバーを表示する knowledge:articleRendererToolBar の例



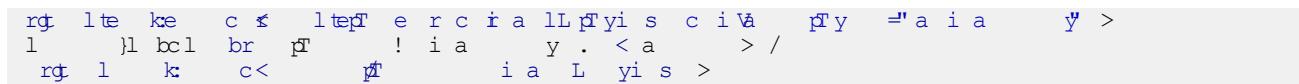
属性

属性名	属性型	説明	必須項目	API バー	アクセス
articleId	String	記事の ID。		29.0	
editable	Boolean	true の場合、投票コンポーネントは編集できる。 false の場合、投票コンポーネントは参照のみです。		29.0	
referenceId	String	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
useCSS	Boolean	このコンポーネントに CSS を含める必要があるかどうかを指定する。		29.0	
useLabel	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global
useChatter	Boolean	Chatter が有効化されており、記事のレンダラにフィードが必要な場合、これを true に設定する。		29.0	

knowledge:articleTypeList

選択可能なすべての記事タイプのループです。

選択可能なすべての記事タイプのリストを表示する単純な例



属性

	属性名	属性型	説明	必須項目	APIバージョン	アクセス
r l	rc	String	articleTypeList コンポーネントの本文に記事タイプはいのオブジェクトを表すために使用できる変数の名前。	はい	29.0	
d		String i	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
r d r d		Boolean	コンポーネントをページに表示するかどうかを指定するboolean値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global

knowledge:categoryList

カテゴリ階層のサブセットのループです。1ページに表示するカテゴリの総数が100を超えることはできません。

次の knowledge:categoryList の例では、「phone」カテゴリのすべての子孫のリストを表示しています。「phone」カテゴリは、「product」カテゴリグループに含まれます。

```

relt      mc ro g c < s      =a      a "ψ
wn e t à d tegt ro gkc e t< ro gne t      ro gac iysro ψ coai ryde' au      ap "pyc      a yG =" "
r eo e emog C o l h p      a =y      " v 1=" " →
t e h o e d l w p n erjtg & e mi v      rdg" b!l n ao yo "a > ! p a y < a i/ >
ewn e t à d t g ro gkc < /      La iys >
1      c      < s / >

```

属性

	属性名	属性型	説明	必須項目	APIバージョン	アクセス
ro	c	String	指定されている場合、コンポーネントはルート(最上位)カテゴリまでカテゴリ階層を列挙する。 rootCategory は、最上位カテゴリを指定するために使用できます。	はい	29.0	
ro g cou		String	個々のカテゴリが属するカテゴリグループ。	はい	29.0	
ro g rc		String	categoryList コンポーネントの本文に記事タイプのオブジェクトを表すために使用できる変数の名前。	はい	29.0	
d		String i	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
l l		Integer	rootCategoryと一緒に指定されている場合、コンポーネントはカテゴリ階層のこの指定の深さで停止する。 -1 は無制限を意味します。	v	29.0	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
rended	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	14.0		global
rootCategory	String	ancestorOf なしで指定されている場合、コンポーネントはこのカテゴリの子孫に対してループ処理を行う。	29.0		

liveAgent:clientChat

Live Agent のチャットウィンドウの主要な親要素です。Live Agent の追加のカスタマイズを行うには、この要素を作成する必要があります。

Live Agent が組織で有効になっている必要があります。このコンポーネントは Live Agent リリースで一度のみ使用できます。

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
id	String	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。	14.0		global
rended	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	14.0		global

liveAgent:clientChatAlertMessage

システムアラートメッセージ（「切断されました」など）を表示する Live Agent のチャットウィンドウ内の領域です。

各チャットウィンドウ内で使用する必要があります。各チャットウィンドウに作成できるアラートメッセージ領域は 1 つのみです。

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
label	String	すべてのエージェントが使用できなくなった場合に表示されるラベルを指定する文字列。デフォル	27.0		

属性名	属性型	説明	必須項目	API バージョン	アクセス
errorLabel	String	トのラベルは「利用できるエージェントがないため、チャット要求はキャンセルされました。」です。		27.0	
mb1	String	アラートを非表示にする場合に表示されるラベルを指定する文字列。デフォルトのラベルは「閉じる」です。		27.0	
oci	String	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
o1C	String	Cookie が無効になった場合に表示されるラベルを指定する文字列。デフォルトのラベルは「お使いのブラウザは、現在の Cookie を受け入れていません。チャットを要求するには Cookie が必要です。Cookie を有効にしてから、もう一度お試しください。」です。		27.0	
bol1h	String	Flash がインストールされていない場合に表示されるラベルを指定する文字列。デフォルトのラベルは「チャットには Flash Player または HTML5 互換 Web ブラウザが必要です。Flash Player をインストールするか、別の Web ブラウザを使用してください。」です。		27.0	
bol1h	String	チャットウィンドウの起動が不適な場合に表示されるラベルを指定する文字列。デフォルトのラベルは「[チャット] ウィンドウはボタンからのみ起動できます。直接アクセスすることはできません。」です。		27.0	
rdrd	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global

liveAgent:clientChatEndButton

訪問者がチャットセッションを終了するためにクリックする Live Agent のチャットウィンドウ内のボタンです。

また、:clientChatEndButton 内で使用する必要があります。

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
d	String i	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。	14.0		global
l b l	String a	ボタンに表示されるラベルを指定する文字列。デフォルトのラベルは「チャットを終了する」です。	24.0		
r d r d	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	14.0		global

liveAgent:clientChatInput

訪問者がエージェントへのメッセージを入力する Live Agent のチャットウィンドウ内のテキストボックスです。

また、gl :c で、内で使用する必要があります。各チャットウィンドウに作成できる入力ボックスは 1 つのみです。

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
d	String i	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。	14.0		global
r d r d	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	14.0		global

liveAgent:clientChatLog

訪問者にチャットのトランскriptを表示する Live Agent のチャットウィンドウ内の領域です。

また、gl :c で、内で使用する必要があります。各チャットウィンドウに作成できるチャットログは 1 つのみです。

属性

	属性名	属性型	説明	必須項目	APIバージョン	アクセス
n g	b lg	pT String	エージェントがメッセージを入力している場合に表示されるラベルを指定する文字列。デフォルトのラベルは「エージェントが入力しています。」です。		24.0	
nB d	b d E	String	エージェントがチャットを終了した場合に表示されるラベルを指定する文字列。デフォルトのラベルは「チャットはエージェントにより終了されました。」です。		24.0	
B dd	m b E	String V	訪問者がチャットを終了した場合に表示されるラベルを指定する文字列。デフォルトのラベルは「チャットを終了しました。」です。		24.0	
r m d	b l	fT String	チャットが新しいエージェントに転送された場合に表示されるラベルを指定する文字列。デフォルトのラベルは「{OperatorName} はチャットセッションの新しいエージェントです。」です。		24.0	
d		String i	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
r d r d		Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global
ro	b m	Stringv i s LN	訪問者が送信するメッセージの横に表示されるラベルを指定する文字列。デフォルトのラベルは「あなた」です。		24.0	

liveAgent:clientChatMessages

システム状況メッセージ（「Chat session has been disconnected (チャットセッションが切断されました)」など）を表示する Live Agent のチャットウィンドウ内の領域です。

この領域は各チャットウィンドウ内に 1 つあります。各チャットウィンドウに作成できるメッセージ領域は 1 つのみです。

属性

	属性名	属性型	説明	必須項目	APIバージョン	アクセス
d		String i	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global

属性名	属性型	説明	必須項目	API バージョン	アクセス
showInClient	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	14.0	global	

liveAgent:clientChatQueuePosition

プッシュ転送を使用するボタンで開始されるチャットセッションのキュー内の訪問者の位置を示すテキストラベルです(プル転送を使用するボタンでは、このコンポーネントは無効です)。

MA t 並 gl :c 並 内で使用する必要があります。

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
queuePosition	String	ページの他のコンポーネントがコンポーネントを	14.0	global	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
rendered	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	14.0	global	

liveAgent:clientChatSendButton

訪問者がエージェントにチャットメッセージを送信するためにクリックする Live Agent のチャットウィンドウ内のボタンです。

HTML 内で使用する必要があります。各チャットウィンドウに複数の送信ボタンを作成できます。

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
label	String	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。	14.0	global	
label	String	ボタンに表示されるラベルを指定する文字列。デフォルトのラベルは「送信」です。	24.0		
rendered	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	14.0	global	

liveAgent:clientChatStatusMessage

システム状況メッセージ（「You are being reconnected (再接続されます)」など）を表示する Live Agent のチャットウィンドウ内の領域です。

HTML 内で使用する必要があります。各チャットウィンドウに作成できる状況メッセージ領域は 1 つのみです。

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
label	String	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。	14.0	global	

属性名	属性型	説明	必須項目	API バージョン	アクセス
log_label	String	ネットワーク待ち時間または中断が発生した場合に表示されるラベルを指定する文字列。デフォルトのラベルは「エージェントから切断されました。接続の再確立を試行していますのでお待ちください...」です。		27.0	
rended	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global

messaging:attachment

添付ファイルを作成し、メールに追加します。

例

```

ne me te glem ltem < t Tis ic a qpi a pC a i i y=" " a
r l d to An ou Tpcc a y=" "
e bu tj r c f& Cane e t qps & ro&h d: &ao m T ! a . 'a
r t e o urop T mapp m c y=" s a. " >
tn m g & Bml indo E < ss i a i a >
l m h < >
b do < >
e tem n p } c mpD > a! pi i . < a , >/
A t d e pch e to { t< er le ftd a& s lni & } o s san Ta ! p a . < a . >/
n r c < >
t enou utt p l wwp < &kal E pr& v o &c // s sa . " >
t te ro ro & ntl d nrot emolF & g o l i rai o ifica c i Ssa .
n r c < / >
b do < / >
l m h < / >
tn m g & Bml indo EK ss i a i a >
tite met eng n :m &dr< ss i da a ma&u d &=C F Pi f =a y s sa . " >
l m h < >
b do < >
n ou{ &de le & c &ue r &ne d & o ians & Ta yc ! a . ap s sa sD a < & P >/
bl b rd r < a = " & >
r < >
e t t h r Gm tn r & qn > s & a < > & > & >
r t rh tlm tE < h u & a a < > & sa >
r r xe { rp e ptl ue r l ad &ov a = a & v C T & ! a . s sa >
r d r h f < < > a =
ct en e h{ l :prax o } & does 1 c// sa }sa bc r nc ! / i . " > ! . s Na
x d tn >r g d < > ! & i < >
x t edtn ot & c d mcd < > ! . a i < a >
xt t d t } u c d < > ! . s &sa >
r < >
t r p p < / a > a
bl b do < / >

```

```

l m h < / >
title m t gng :m d< s/s i a a a >
ne me tg gl m l: m < Tps/s i a i a a >

```

属性

	属性名	属性型	説明	必須項目	API バージョン	アクセス
1	mf	String	添付ファイルのファイル名を設定する。指定され ていない場合、ファイル名が自動的に生成されま す。		14.0	
d	refId	String	ページの他のコンポーネントが添付ファイルコン ポーネントを参照できるようにする識別子。		14.0	global
1	mailContentInline	Boolean	メールの添付ファイルのコンテンツの配置をイン ラインに設定する。		17.0	
rdr	displayType	String	添付ファイルの表示方法を指定する。有効な値は、 任意の MIME タイプまたはサブタイプです。デ フォルト値は「text」です。		14.0	
rdrd	showComponent	Boolean	コンポーネントをページに表示するかどうかを指 定する boolean 値。指定されていない場合、この値 はデフォルトの true に設定されます。		14.0	global

messaging:emailHeader

メールにカスタムヘッダーを追加します。ヘッダーの本文は 1000 文字までに制限されます。

例

```

ne me tg gl m l: m < t Tps ic a qpi a pC " a
r l d to An ou Tpcc a y=" "
t bu nj te ec g u oT mcs h=" si a " " a
r teo urop T mapp m c y=" s a . " >
ne em e g gl nm dter e < m uss idmer Hi a a =a H " a
B An e eot d ccr l Id: od I T ! a .
A E C M ET E D UN H OSD R
ne em g gl m d r < s/s i a Hi a a>
tn m g g Bml indo E < ss i a i a y
l m h < >
b do < >
e t e m n p } c m p D > a! pi i . < a , > /
t e e ou kpe d r h okh > l fn ap i s i < a > /
b do < / y
l m h < / >
tn m g g Bml indo E < s/s i a i a y
ne me tg gl m l: m < Tps/s i a i a a >

```

上述の例では次の HTML を表示します。

```

e       u   b   h   T   D 2   :a : , F 9  9 +0093   5 5   0000
e r Ao dm tnegr e u ro l F r o p Uio fcs cs s sa . >
nd ree @ l : l ro p o S fcc c - s sa . >
l t ee o u r q : T mapp mR c y - s a .
e@ no ed ml Tro e@ a efc c "ml i ars sao fmc <# i a s sa . >
g M I : ss2D a@J nl d7h164M@ 639m8 6558v0 6 a a a i a i a > S
t bu rj te ec : g u oT dcr S h si s a
n r MIME : V -s 1 . 0
o t t@ t lue rn:p r p -y i a /a v a;
b od rt r =by 2--- 21Pa 8 41366 A339 8 6558 0 6
te e u o C dm@ X: FAE SIX A@ M d@ RxxcdU NG EH:O@A@ oR I A E C@ O O M@ E T E D UN H
x C : j X E@S B K 009900000000
x r C : u u X UF-S sp 005 0000000
te d ree oe m l p ro X e S fmc s s a s sa .
el mb utt n qwven e X :h- la: sp@ sindi s@mc m b cu // spj sa . p / ay /sa .s
n Bn d C@ : B m@u F-S i@i@ 1 V 9 R i4 y 5
n re A lgrer l m b X: @-i 2 a i a: F09 9 C 0@P@E IME5 37 04 ( E F @ C] = F FF8739 5 0 0 7
n tn zM rE dm g o X - S-C : a O- i ai a -S 0

```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
i	String	ページの他のコンポーネントが emailHeader コンポーネントを参照できるようにする識別子。		14.0	global
a	String	ヘッダーの名前。注: この名前の先頭には X-SFDC-X- が付きます。	はい	14.0	
boolean	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global

messaging:emailTemplate

Visualforce メールテンプレートを定義します。すべてのメールテンプレートタグは、1 つの emailTemplate コンポーネントタグ内でラップされている必要があります。emailTemplate には、htmlEmailBody タグまたは plainTextEmailBody タグのいずれかを含める必要があります。詳細およびフォームコンポーネントは、子ノードとして使用することはできません。このコンポーネントは Visualforce メールテンプレート内でのみ使用できます。[設定] | [コミュニケーションテンプレート] | [メールテンプレート] を使用して、メールテンプレートを作成および管理できます。

例

```

re me tg gle m ltem m < t T@ic a q@ a p@C a i i y=" " a
e r l d to An ou T@cc a y=" "
bu jt @u c ou cc s c =Y a s 's s@ a . i a " >
e r ee o n p c T@l m o m y=" s sa a . i a " >
tn m g @ Bml mdo E< ss i a i a y=
l m h < >
b do < y
{ e e llt@ n n p } c @p > ! ip i . <a -- >/

```

	tr e t el pe owe & H > eni sl ifate mo ht cs u sfr do d } o w a Ta : ! p a .
x t te e d bl p m l dd c & ex{ ipa e ala a e r l i a l = ") v 5 = " a " v C T a ! a . s sa " >	
x e n lo u e x m p a u e e < } her a c e dvrel ha ! . s b N a G m " V a a a ! s a > /	
x e n lo u e x m p a t e e } b a e j c a e d r c t l h u a " bu ! j. S " v a a a ! s " > /	
x e n lo u x m p a t n e & e j c a d m e v d t a " h u r. ro a d l " v E a a a ! a s " i "	
x t te d bl p < /a a a a >	
b do < /y	
l m h < />	
tn m g & Bml indo E & s/s i a i a y	
t t e m t g n g n : m & d h < d s s i la a p maf d c s = " a " p a s s a . " >	
l m h < />	
b do < /y	
e t n h r a n t l { & e o & d t 3 & e s r s a n d } o y s s a n a t h ! a . < a 3 >	
x t te ed k l b p r d r < a l 2 a c x e g { a c a p e e t l " u e r s l i d a = ") v 5 = " a " v C T a ! a .	
x e n lo u e x m p a u e e < } her a c e dvrel ha ! . s b N a G m " V a a a ! s a > /	
x e n lo u e x m p a t e e } b a e j c a e d r c t l h u a " bu ! j. S " v a a a ! s " > /	
x e n lo u x m p a t n e & e j c a d m e v d t a " h u r. ro a d l " v E a a a ! a s " i "	
x t te d bl p < /a a a a >	
b do < /y	
l m h < />	
t t e m t g n g : m & < s/s i a a a >	
t t e m t g n g : m & < ms s i a c a a i a s s a v s " >	
ex e t r x e { r p e e p t l u & e r l ad & o v a = " a " v C T a ! a . s s a " >	
ex e { } & o c r m t & o b x t i g t n . & c N a & c x t , & m c . S } , & c . a i a , ! . s s a	
t t e m t g n g : m & < s/s i a a a >	
ne me tg gl m l : m < T & s i a i a a >	

翻訳テンプレートの例

ex e e th me Tn q ep ! b & h s n e b n a t i n b l d l e w a e a t o a t h r s h d a a h a y v a
er reed l k e l c e t l h e t T t m p t n e & t h o h b o & j s s a & m a h o d m c u g g l d s f a &
n n o h e c a l d l u g g a i s a v a k a a a y >-
ne me tg g l e m l t e m < t T & s i c a q i a p c a i i y " " a
r l d t o A n ou T p c c a y " "
e l e u g g m e n l c j u g g a = a b i i . a a "
e h u j e b d t l m j b u & Lc ! a . i a " -
r & o n p c m o l m o m y " s s a a . i a " >
tn m g & Bml indo E < s/s i a i a y
l m h < />
b do < /y
ee b & { & l n m r & g \$ t & g D & ! } a c i m p a _ i ! p i . & a -- >/
b l p & l n d o & & D & ! p a . i a _ y >/
b do < /y
l m h < />
tn m g & Bml indo E & s/s i a i a y
e ne me tg gl m l : m < T & s i a i a a >

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
	String i	ページの他のコンポーネントが emailTemplate コンポーネントを参照できるようにする識別子。		14.0	global

属性名	属性型	説明	必須項目	API バージョン	アクセス
g	String	メールテンプレートの表示に使用される言語。有効な値は、「en」、「en-US」などのSalesforce.comでサポートされている言語キーです。recipientTypeおよびrelatedToTypeの差し込み項目を受け入れます。		18.0	
c	String	メールを受信する Salesforce.com オブジェクト。		14.0	
o	String	テンプレートの差し込み項目データの取り出し元である Salesforce.com オブジェクト。有効なオブジェクトは、Visualforce がサポートするカスタムオブジェクトなど、標準コントローラを持つオブジェクトです。		14.0	
	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global
p	String	返信メールヘッダーを設定する。		14.0	
c	String	メールの件名行を設定する。最大 100 文字です。	はい	14.0	

messaging:htmlEmailBody

HTML バージョンのメール本文です。

例

```

ne me te glem item < t Tts ic a qpi a pC a i i y" "a
rl d to An ou Tflcc a y" "
e bu ty r c f@Camo e t qps e ro&h d: >ao m T ! a . "a
r t ee o urop T mapp m c y" s a. " >
tn m g d Bml indo E < ss i a i a >y
l m h < >
e t tx p <c s y y" ssy >
t bndo @ le mf exft : C } y :‡ 2a y p i ; $1 ;
t {
e wt bl
e b rd r x d h : p a i - ; 5
e b rd n x g c p : p s - i a ; 5
e et b rd r l d d : h s - y s a;
e b rd r loro c : - #FF 0000
e bn r o@ doro ck : a - #FFFFFF;
}

{ t d
e wt b rd r x d h : p i -1 ;
e n xdd g p : p ia a ;
e et b rd r l lod : s - y s i ;
e b rd r loro c : - # 000000
e bn r o@ doro ck : EEC@ - #FF ;
}

```

```

e     loro      c   :          # 000000
e     wt b rd r x d       h :     p     i -1 ;
e     n xdd g p :         p     ia    4;
e     et b rd r l lod :      s - y si ;
e     b rd r loro c :      -    # 000000
e     bn r oj doro ck :     a     -    #FFFFF ; 0
}
l           < s / y>
b do
<
y
e { e e tem n p } c mpD > a! pi i . < a , > /
e w B t oe dpe to { t er le ffd @ s lni al } o s san Ta ! p a . < a . > /
te e bl b rd r < a = " "
t r < >
te e t t h r m t n r & dh > s & & < > & & >
te t r t rch tlm tE < hu h a & a < > & sa & 
t r < >
e e ex t r xe { rp e ptl ue r l ad jov a = " a v C T a" ! a . s sa" >
t r < >
t e d r h f < < > a =
tt en e h l : p ro < o } ec does 1 c//sa }sa bcr nc. !/i. " > ! . s Na
t d < < >/a &
{ t x d tn r g d < > ! & i < >
{ t x t eltn o t c d m c d < > ! . a i < a >
{ t xt d t } u c d < > ! . & sa &
t r < >
e e ex t r p p < /a > a
te bl p < /a>
e e t n r c < >
e ext t enou utt p l wep < e hkal E pr a v o & c // s sa . " >
e e te ro ro d ntl d nrot emolF g o l i rai o ifca c i Ssa .
e ext t nou u p p < ka L i >
e e t n r c < / >
b do
l m h < /y
tn m g d Bml indo & s s i a i a & y
ne me tg gl m l: m < Tps s i a i a & y

```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
	String i	ページの他のコンポーネントが htmlEmailBody コンポーネントを参照できるようにする識別子。		14.0	global
		コンポーネントをページに表示するかどうかを指定するワード「腫脹」、「看鑫」、「赫化」、「野京」、「蜀ま」、「龍ズニ」、「×衰弱」、「毛」、「ネラ」、「榜」、「ヤ」、「咬」		14.0	global

```

e bu t̄ r c f̄ C a m o e t q̄ s e r c h d: b̄ o m T ! a . " a
r t̄ e o u r o p T mapp m c y" s a. " >
ne mtnx gl g :B l pmds E Tss i a i a i a >
e t e n n } c mp D a! i i . a ,
B t o e d e to { t er le ffd e s lni sl } o s san Ta !
a . a .
t r xe { r p e pt l u e r l ad j o v a = " a " v C T a ! a . s sa" >
{ ex b reGn ; bcr nC s ! . s Na
n ix g n } r : g d i ! d i
{ o x tle mnc o: jc d m c a i a ! . a i a
xt tu } : uc S sa ! . S sa
t r p p < /a >a
te ro ro d ntl d nrot emolF g o l i rai o ifmac i Ssa .
ne mtnx gl g :B l pmds E Tss i a i a i a >
ne me tg gl m l: m < Tss i a i a >a

```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
id	String	ページの他のコンポーネントが plainTextEmailBody コンポーネントを参照できるようにする識別子。	14.0		global
renderd	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。	14.0		global

site:googleAnalyticsTracking

サイトの使用状況を追跡および分析する目的で Google Analytics と Force.com サイトを統合するために使用する標準コンポーネントです。このコンポーネントは、追跡するページのサイトテンプレートか個別のページのいずれかに1つだけ追加してください。テンプレートとページの両方にコンポーネントを設定しないでください。注意: このコンポーネントは、Force.com サイトで使用されているページでのみ機能します。サイトが組織で有効化されていることと、[分析追跡コード] 項目に値が入力されている必要があります。追跡コードを取得するには、Google Analytics Web サイトにアクセスしてください。

例

```

tn egl n m d--G t nent o iapnt c et bpe o iothom d ap
e n r et cg l gd p m i $ a a ia . >-
tn Adg gl: n r < c $g c T k iay s ia > /

```

上述の例では次の HTML を表示します。

```

e tr tx c t p jpr < s ic py" v /saia " >
t r ttg Jo e t nnt o upa ss =o Htco osl8 == pc tt h wwwap. : h ") ? ps ss/. "
e mdeo ue mr c u t c pr Cci ep(cg Jo "a %sdg + g $ = " +'Ho s ansg ej" - aia y s .

```

```

x      t      jpr      t      c      r      pyE      C      'cv      / sapla      3 %'3 % s i/      3 %      ; ))
r      c      p      <      s i      >
r      c      p      <      s i      >
e      r      reg      t      e      p { c      gT      erg      kr      a      tn      a      Ta      ±      e      kn      _ra      _$}      cdag"      d      !      T      CS      .k      iay      s      i a      "      ; )
e      te      e      u      A      { b      midr      }      lh      i      in      s      s      ssN      ssf      Pa      (      !      i      S      .      iP      "      ; / )
}
e      te      e      u      A      b      midr      s      i      in      s      s      ss      )
e      e      trg      ew      A      p      l      d      o      r      k      r      u      a      a      ks      -L      i      Na      'a      "      ; )
e      e      trg      ew      A      p      l      d      o      r      k      r      u      a      a      ks      -L      i      s      a      (      s      a;      )
}
t      reger      ipw      c      T      g      ck      ak      a      .      _      av      Ea      ;( )
rr      c      ch      a      (      )
r      c      p      <      s i/      >

```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
id	String id	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
renderd	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global

site:previewAsAdmin

このコンポーネントは、管理者プレビュー モードのサイトにエラーメッセージの詳細を表示します。次のように、 apex:page の終了タグの直前にこのコンポーネントを追加することをお勧めします。注意: site:previewAsAdmin コンポーネントには、 apex:messages タグが含まれているため、エラーページの他の場所にそのタグを使用している場合、エラーメッセージが 2 回表示されます。

例

```

m      d      ndmmde      t      n      edo      em      rc      s      bop      ro      eh      mi      fo      t      g      i      g      p      g      py      $      a      a
r      A      A      d      p      m      <      $      v      i      s      i      >      /      g      i      g      p      g      py      $      a      a

```

上述の例では次の HTML を表示します。

```

d      dj      p      dj      :      <      s      i      a      =      i      _      10      "      $0
d      dj      p      dj      dj      <      sdj      i      a      =      i      _      2      10      "      $0      _      5      _      $

```

```

    ad      l b rd r loeo < et iv b:sd ry="el" load $9 #FFix d 00      s -hy:s i p      i f      ;
n xdd x  x g x p : p bn p r poia dorot5e nk 5t p 6 CCa f - : #FFEp ;      $1-      ;0
nt r m ng eg it h r:2mxlg p t an -1f 2m g ap0 ia 12p p;0 ia - ;5 >
    bl al dd c g ap< c ag cp ia = " 0 s i a= " >0
    b do r      <      < /y>
    d e trmg w nge r< t< ig g s =" ig fl/ s h/s A a i " i = " 4 0
    l n xdd ngx ipwtm: g ps dy=" ipa h d;5 ia ;0 i =" " 40 > / 
    dn r o lgi d dj < dj < > dj < > mig =": i io - 50 - 5 s3"
    l nx r em g t ln e t nptgo yed ia fl <5 >iu> Pa - s * >
    n r o g < s / >
    e te re te ket rf le < bt u dagit g o/s sv /   s ?l = "
    l n xdd g xpr m tg sp y=b du tapel e4; m r afa r L;5 > do      Mi s      a v iP < >/a
    d
    r b do <      < /y
    bl <      / a>
    d <      />
    p <      s / >a
    p <      s / >a

```

属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
	String i	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global

social:profileViewer

取引先(個人取引先を含む)、取引先責任者、またはリードの詳細ページにソーシャル取引先と取引先責任者のビューアを追加する UI コンポーネントです。ビューアにはレコード名、プロファイル写真、およびソーシャルネットワークアイコンが表示されるため、ユーザは自分のアカウントにサインインしてソーシャルデータを直接 Salesforce で参照できます。

ソーシャル取引先と取引先責任者が組織で有効化されている必要があります。このコンポーネントは、Account、Contact、および Lead オブジェクトでのみサポートされており、ページで一度だけ使用できます。このコンポーネントは、Force.com サイトの Visualforce ページでは使用できません。

この例では、取引先責任者のソーシャル取引先と取引先責任者のビューアを表示しています。

```

    x      n tegndprd p or h & C o a &C ca a = " " a>
    e e @ le wr dl&f : r p ft & s i o yla id ci c i y= ! ia. " > /
    ex      g p p < /a >

```

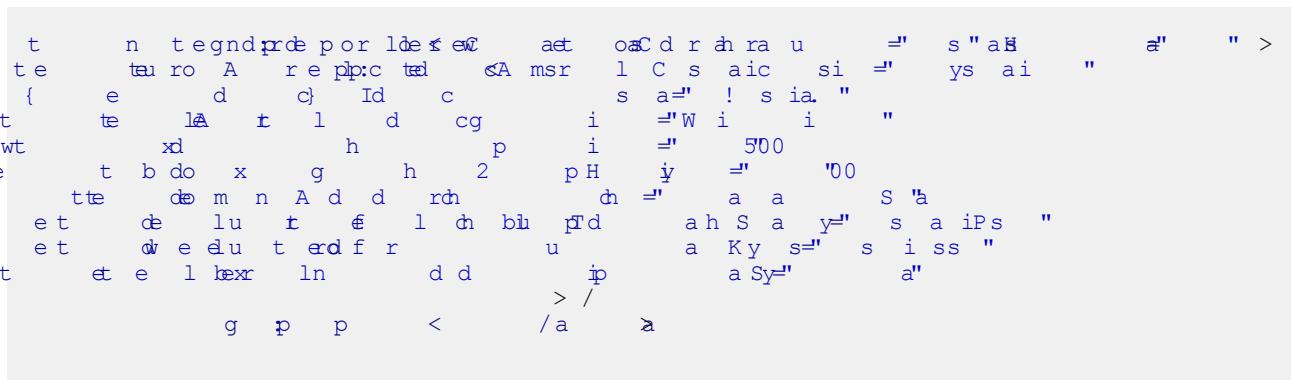
属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
d	I id	i ジーシャル取引先と取引先責任者のビューアを表示する、レコードのエンティティ ID。例: Contact.Id。	はい	24.0	
d	String i	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
r d r d	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global

support:caseArticles

ケース記事ツールを表示します。ツールは、ケースに現在添付されている記事や記事のキーワード検索を表示できます。このコンポーネントは、ケースフィールドとナレッジが有効化されている組織でのみ使用できます。このコンポーネントを使用するページには、バージョン 3 より前の Ext JS を含めることはできません。

次の例はケース記事ツールを表示します。



属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
r l	c pString	i a 検索を絞り込むために使用される記事タイプ。複数の記事タイプをカンマ区切りで定義できます。		25.0	
n o l	b h d b E ET Boolean	a a 記事をメールに添付できるかどうかを指定する boolean 値。		25.0	

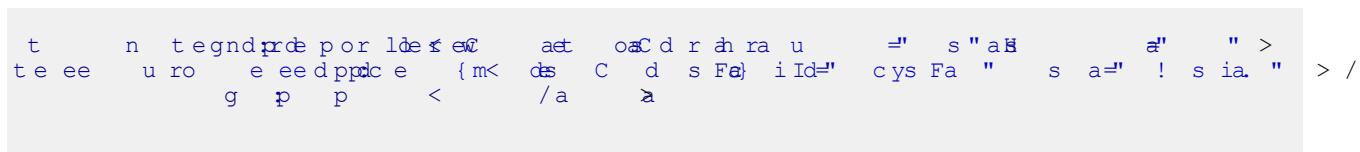
属性名	属性型	説明	必須項目	APIバージョン	アクセス
height	String	本文の高さ。ピクセル(px) 単位で数値を指定するか、「auto」を指定して表示されている記事リストに応じて高さが自動調整されるようにします。		25.0	
caseId	String	ケース記事を表示するレコードのケース ID。	はい	25.0	
category	String	検索を絞り込むために使用するデータカテゴリ。この値の形式は「CategoryGroup1:Category1」です。CategoryGroup1 と Category1 は、順に、カテゴリグループ名とカテゴリです。複数のカテゴリ条件をカンマ区切りで指定できますが、カテゴリグループあたり 1 つのみ指定できます。		25.0	
isDefault	Boolean	デフォルトのデータカテゴリの対応付けの事前絞り込みを考慮するかどうかを指定する boolean 値。		25.0	
defaultSearchType	String	defaultSearchType 属性が「keyword」である場合に使用されるキーワード。キーワードが指定されていない場合は、ケースの件名がデフォルトとして使用されます。		25.0	
firstPublished	String	記事検索フォームが初めて表示されるときのデフォルトのクエリを指定します。値は「keyword」、「mostViewed」、または「lastPublished」です。		25.0	
componentId	String	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
allowUrlSharing	Boolean	記事を URL で共有できるかどうかを指定する boolean 値。		25.0	
language	String	多言語のナレッジが有効化されている場合に、検索を絞り込むために使用される言語。		25.0	
logKeywords	Boolean	キーワード検索のログを記録するかどうかを指定する boolean 値。		25.0	
dom	String	ケース、記事検索フォームのいずれかまたは両方に添付されている記事を、コンポーネントで表示するかどうかを指定します。値は「attached」、「search」、「attachedAndSearch」、または「searchAndAttached」です。		25.0	
onSearchCompleted	String	記事検索が完了した後に呼び出される JavaScript。		25.0	
rendered	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global
renderer	Object	action メソッドの結果がクライアントに返されるときに再作成される 1 つ以上のコンポーネントの ID。この値には、単一の ID、ID のカンマ区切りのリスト		25.0	

属性名	属性型	説明	必須項目	API バージョン	アクセス
ト、またはIDのリストまたはコレクションの差し込み項目の式を使用できます。					
Br_u_o_d_m	String	s_a 検索ボタンの表示名。		25.0	
Br_t_r_ld_d_h	String	s_w Fa キーワード検索項目の幅(ピクセル(px)単位)。		25.0	
Br_n_u_o_b_mc	String	s_Fa i ウィジェットが現在検索モードである場合に、記事を検索するためにJavaScriptからコールできる関数の名前。		25.0	
A_d_d_h_c	Boolean	s_v 高度な検索へのリンクを表示するかどうかを指定する boolean 値。		25.0	
l_l_b_r_l	String	i コンポーネントのヘッダーに表示されるタイトル。		25.0	
d_h	String	i タイプのスタイル。「expanded」、「collapsed」、「fixed」、または「none」のいずれかです。		25.0	

support:caseFeed

ケースフィードコンポーネントには、パブリッシャー(メール、ポータル、活動の記録、内部メモ)、ケース活動フィード、フィードフィルタ、強調表示パネルなど、標準ケースフィードページのすべての要素が含まれます。このコンポーネントは、ケースフィードが有効化されている組織でのみ使用できます。

次の例はケースフィードコンポーネントを表示します。



属性

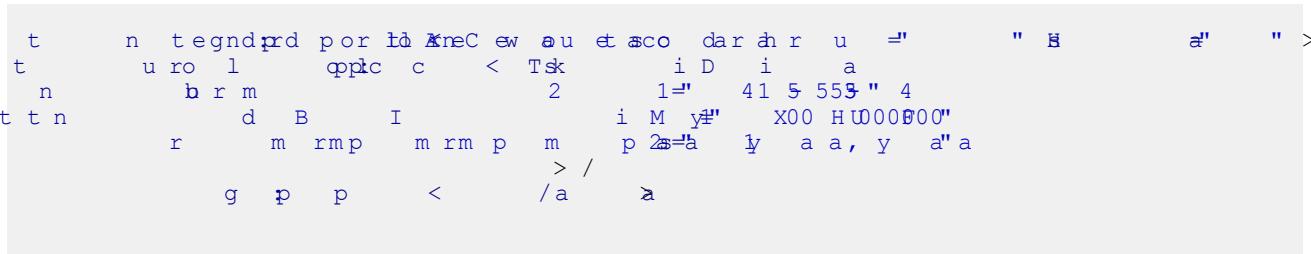
属性名	属性型	説明	必須項目	API バージョン	アクセス
c_I	id	s_a ケースフィードを表示するレコードのケース ID。	はい	26.0	
	String	i ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global

属性名	属性型	説明	必須項目	API バージョン	アクセス
show	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global

support:clickToDial

Open CTI または Salesforce CRM Call Center 用に有効化されたクリック-to-ダイヤルとして有効な電話番号を表示するコンポーネントです。この項目では、Salesforce とのコンピュータテレフォニーインテグレーション(CTI)の既存のクリック-to-ダイヤルコマンドが優先されます。

次の例に、クリック-to-ダイヤルコンポーネントを示します。



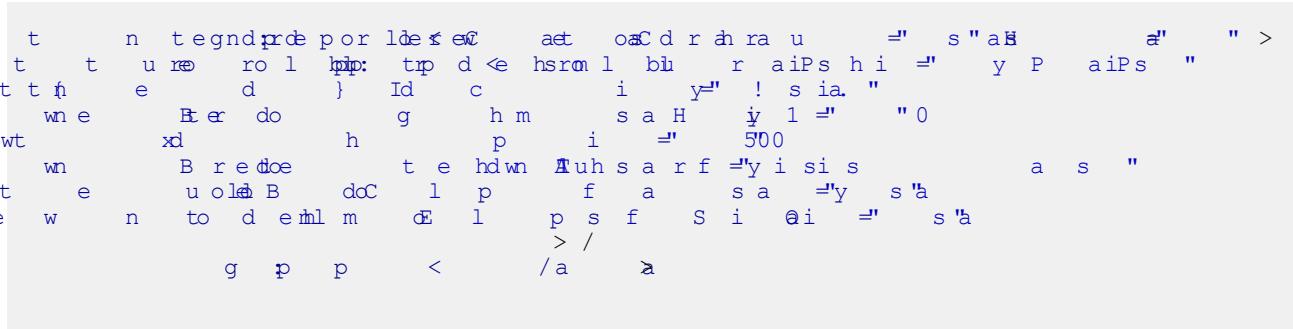
属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
recordId	String	クリック-to-ダイヤルの呼び出し元となるレコードのエンティティ ID。		28.0	
identifier	String	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
phoneNumber	String	クリック-to-ダイヤル機能を呼び出す電話番号。	はい	28.0	
parameters	String	ケースパラメータや取引先パラメータなど、クリック-to-ダイヤルの呼び出しに関連するパラメータ(省略可能)。		28.0	
show	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global

support:portalPublisher

support:portalPublisher では、ケースフィードを使用するサポートエージェントは、ポータルメッセージを作成し、投稿できます。このコンポーネントは、ケースフィードが有効化されている組織でのみ使用できます。

次の例はポータルパブリッシャーを表示します。



属性

属性名	属性型	説明	必須項目	API バージョン	アクセス
B r do	String	s a 回答本文のデフォルトのテキスト値。		25.0	
t B r do	g h String	s a 回答本文の高さ (em 単位)。		25.0	
u d l E	do C Boolean	p a 回答本文が空の場合に本文を折りたたんで上下の サイズを小さくするかどうかを指定する boolean 値。		25.0	
d	I id	i p ポータルパブリッシャーを表示するレコードの工 程ティティ ID。現在のバージョンでは、ケースレ コード ID のみがサポートされています。	はい	25.0	
d	String i	ページの他のコンポーネントがコンポーネントを 参照できるようにする識別子。		14.0	global
o bu	str u	String i s F回答がポータルに公開できなかった場合に呼び出 される JavaScript。		25.0	
o bu	m cc	String i s 回答がポータルに正しく公開された場合に呼び出 される JavaScript。		25.0	
r d r d	Boolean	コンポーネントをページに表示するかどうかを指 定する boolean 値。指定されていない場合、この値 はデフォルトの true に設定されます。		14.0	global
r d r	Object R	回答が正しく公開されたときに再作成される 1 つ 以上のコンポーネントの ID。この値には、単一の ID、ID のカンマ区切りのリスト、または ID のリ		25.0	

属性名	属性型	説明	必須項目	APIバージョン	アクセス
ストまたはコレクションの差し込み項目の式を使用できます。					
showEmail	Boolean	メール通知の送信オプションを表示するかどうかを指定する boolean 値。		25.0	
showSubmit	Boolean	送信ボタンを表示するかどうかを指定する boolean 値。		25.0	
buttonName	String	ポータルパブリッシャーの送信ボタンの名前。		25.0	
methodName	String	回答を公開するために JavaScript からコールできる関数の名前。		25.0	
pageTitle	String	ポータルパブリッシャーヘッダーに表示されるタイトル。		25.0	
width	String	ポータルパブリッシャーの幅(ピクセル(px)またはパーセント(%)単位)。		25.0	

topics:widget

属性

属性名	属性型	説明	必須項目	APIバージョン	アクセス
トピックページへのカスタム URL。指定された URL の最後に topicId が追加されます。					
customUrl	String	トピックページへのカスタム URL。指定された URL の最後に topicId が追加されます。		29.0	
entityId	String	フィードを表示するレコードのエンティティ ID。 例: Contact.Id。	はい	29.0	
referenceId	String	ページの他のコンポーネントがコンポーネントを参照できるようにする識別子。		14.0	global
rendered	Boolean	コンポーネントをページに表示するかどうかを指定する boolean 値。 指定されていない場合、この値はデフォルトの true に設定されます。		14.0	global

付録

付録

A

グローバル変数、関数、および式の演算子

Visualforce ページでは、数式と同じ式の言語を使用します。つまり、`m` 内のすべてが、現在コンテキストにあるレコードから得られる値にアクセスできる式として評価されます。

この付録では、Visualforce 式で使用できる変数、関数、および演算子の概要を説明します。

グローバル変数

ページで現在のユーザと組織に関する情報を参照するには、グローバル変数を使用します。

関数

レコードのデータの変換、計算の実行、Visualforce 属性の値の指定を行うには、関数を使用します。

式の演算子

式を結合して複合式を作成するには、演算子を使用します。

グローバル変数

ページで現在のユーザと組織に関する情報を参照するには、グローバル変数を使用します。

グローバル変数は、評価する Visualforce 式の構文 (`m` `$` など) を使用して参照する必要があります。

`$Action`

[取引先] タブのホームページの表示、取引先の新規作成、取引先の編集、取引先の削除など、標準 Salesforce アクションを参照する際に使用するグローバル差し込み項目の種別です。

`$Api`

API URL を参照する際に使用するグローバル差し込み項目の種別です。

`$Component`

Visualforce コンポーネントを参照する際に使用するグローバル差し込み項目の種別です。

\$ComponentLabel

メッセージに関連付けられている Visualforce ページで^p コンポーネントのラベルを参照するときに使用するグローバル差し込み項目です。

\$CurrentPage

現在の Visualforce ページまたはページ要求を参照するときに使用するグローバル差し込み項目の種別です。

\$Label

Visualforce ページのカスタム表示ラベルを参照する際に使用するグローバル差し込み項目の種別です。

\$Label.Site

Visualforce ページの標準サイト表示ラベルを参照する際に使用するグローバル差し込み項目の種別です。すべての標準表示ラベルと同様、テキストはユーザの言語および地域に基づいて表示されます。

\$ObjectType

標準オブジェクトまたはカスタムオブジェクト(取引先、ケース、商談など)およびその項目の値を参照する際に使用するグローバル差し込み項目の種別です。

\$Organization

組織プロファイルを参照する際に使用するグローバル差し込み項目の種別です。Organization 差し込み項目を使用して、組織の市区郡、Fax、ID、その他の詳細情報を参照します。

\$Page

Visualforce ページを参照する際に使用するグローバル差し込み項目の種別です。

\$Profile

現在のユーザを参照するときに使用するグローバル差し込み項目の種別です。Profile 差し込み項目を使用して、ライセンスの種類や名前など、ユーザのプロファイルに関する情報を参照します。

\$Resource

Visualforce ページで、既存の静的リソースを名前で参照するときに使用するグローバル差し込み項目の種別です。また、関数でリソース差し込み項目を使用して、静的リソースアーカイブ内の特定のファイルを参照することもできます。

\$SControl

既存のカスタム S コントロールを名前で参照する際に使用するグローバル差し込み項目の種別です。この種別の差し込み項目から、S コントロールが実行されるページへの URL が作成されます。

\$Setup

「階層」型のカスタム設定を参照する際に使用するグローバル差し込み項目の種別です。

\$Site

現在の Force.com サイトを参照するときに使用するグローバル差し込み項目の種別です。

\$System.OriginDateTime

リテラル値 1900-01-01 00:00:00 を表すグローバル差し込み項目です。

\$User

現在のユーザを参照するときに使用するグローバル差し込み項目の種別です。User 差し込み項目は、別名、役職、ID など、ユーザに関する情報を参照します。

\$User.UITheme および \$User.UIThemeDisplayed

これらのグローバル差し込み項目は、指定された Web ページ上でユーザに表示される Salesforce のデザインを識別します。

\$UserRole

現在のユーザのロールに関する情報を参照するときに使用するグローバル差し込み項目の種別です。Role 差し込み項目は、ロール名、説明、ID などの情報を参照します。

\$Action

[取引先] タブのホームページの表示、取引先の新規作成、取引先の編集、取引先の削除など、標準 Salesforce アクションを参照する際に使用するグローバル差し込み項目の種別です。

使用方法

オブジェクトとアクションを指定するには、ドット表記を使用します(例: `Account.create()`)。

例

次のマークアップは、新規アカウントを作成するためのリンクを追加します。

```
t w en&ou u p l up t < An kta Ahow i ou ! R F & ( i . N " )  
t nou u p p < ka L i >
```

次のマークアップは、添付ファイルをダウンロードするためのリンクを追加します。

```
x t n tegndprd ptoer ld & Cn a a a a a a a a  
x t nou u p p < ka L i  
x t l ut tne A o n w n ! R dF & ( i . a a . " )  
w n w t e t n } md dn a a i . " )  
x t w ol d o D Na !  
x t nou u p p < ka L i >  
g p p < /a >
```

\$Action グローバル変数の有効な値**関連リンク**

[\\$Action を使用した action メソッドへの動的参照](#)

\$Action グローバル変数の有効な値

次の表は、\$Action グローバル変数を使用して参照できるアクションとそのアクションを実行できるオブジェクトを示しています。すべてのオブジェクトでは、新規作成、コピー、表示、編集、リスト、削除などの基本アクションをサポートしています。\$Action グローバル変数は、多くの標準オブジェクトで使用できるアクションも参照します。組織で使用可能な値は、有効化している機能によって異なります。

値	説明	オブジェクト
Accept	レコードを引き受けます。	<ul style="list-style-type: none"> 広告グループ ケース

- ・ 行動
- ・ Google キャンペーン
- ・ キーワード
- ・ リード
- ・ 検索語句
- ・ SFGA バージョン
- ・ テキスト広告

Activate	契約を有効化します。	Contract
Add	価格表に商品を追加します。	Product2
AddCampaign	キャンペーンにメンバーを追加します。	Campaign
AddInfluence	商談の影響のあるキャンペーンリストにキャンペーンを追加します。	Opportunity
AddProduct	価格表に商品を追加します。	OpportunityLineItem
AddToCampaign	キャンペーンに取引先責任者またはリードを追加します。	<ul style="list-style-type: none"> ・ 取引先責任者 ・ リード
AddToOutlook	Microsoft Outlook に行動を追加します。	Event
AdvancedSetup	キャンペーンの高度な設定を起動します。	Campaign
AltavistaNews	www.teltel.com を起動します。	<ul style="list-style-type: none"> ・ 取引先 ・ / ・ リード
Cancel	行動をキャンセルします。	Event
CaseSelect	ソリューションにケースを指定します。	Solution
ChangeOwner	レコードの所有者を変更します。	<ul style="list-style-type: none"> ・ 取引先 ・ 広告グループ ・ キャンペーン ・ ケース ・ 取引先責任者 ・ 契約 ・ Google キャンペーン ・ キーワード ・ リード ・ 商談 ・ 検索語句 ・ SFGA バージョン ・ テキスト広告

ChangeStatus	ケースの状況を変更します。	<ul style="list-style-type: none"> ・ ケース ・ リード
ChoosePricebook	使用する価格表を選択します。	OpportunityLineItem
Clone	レコードをコピーします。	<ul style="list-style-type: none"> ・ 広告グループ ・ 納入商品 ・ キャンペーン ・ キャンペーンメンバー ・ ケース ・ 取引先責任者 ・ 契約 ・ 行動 ・ Google キャンペーン ・ キーワード ・ リード ・ 商談 ・ 商品 ・ 検索語句 ・ SFGA バージョン ・ テキスト広告 ・ カスタムオブジェクト
CloneAsChild	親ケースの詳細と関連付けたケース Case を作成します。	Case
CloseCase	ケースをクローズします。	Case
Convert	リードから得られる情報を使用して、新規の取引先、取引先責任者、商談を作成します。	Lead
ConvertLead	リードをキャンペーンメンバーに変換します。	キャンペーンメンバー
Create_Opportunity	キャンペーンメンバーに基づいて、 キャンペーンメンバー商談を作成します。	
Decline	行動を辞退します。	Event
Delete	レコードを削除します。	<ul style="list-style-type: none"> ・ 広告グループ ・ 納入商品 ・ キャンペーン ・ キャンペーンメンバー ・ ケース ・ 取引先責任者 ・ 契約 ・ 行動 ・ Google キャンペーン

		<ul style="list-style-type: none"> ・ キーワード ・ リード ・ 商談 ・ 商談商品 ・ 商品 ・ 検索語句 ・ SFCA バージョン ・ 解決方法 ・ ToDo ・ テキスト広告 ・ カスタムオブジェクト
DeleteSeries	定期的な行動または ToDo を削除します。	<ul style="list-style-type: none"> ・ 行動 ・ ToDo
DisableCustomerPortal	カスタマーportalユーザを無効化します。	Contact
DisableCustomerPortalAccount	カスタマーportalアカウントを無効化します。	Account
DisablePartnerPortal	パートナーportalユーザを無効化します。	Contact
DisablePartnerPortalAccount	パートナーportal取引先を無効化します。	Account
Download	添付ファイルをダウンロードします。	<ul style="list-style-type: none"> ・ 添付ファイル ・ ドキュメント
Edit	レコードを編集します。	<ul style="list-style-type: none"> ・ 広告グループ ・ 納入商品 ・ キャンペーン ・ キャンペーンメンバー ・ ケース ・ 取引先責任者 ・ 契約 ・ 行動 ・ Google キャンペーン ・ キーワード ・ リード ・ 商談 ・ 商談商品 ・ 商品 ・ 検索語句 ・ SFCA バージョン ・ 解決方法 ・ ToDo

			<ul style="list-style-type: none"> ・ テキスト広告 ・ カスタムオブジェクト
EditAllProduct	価格表にあるすべての商品を編集し OpportunityLineItem ます。		
EnableAsPartner	取引先をパートナー取引先として指定します。 Account		
EnablePartnerPortalUser	取引先責任者をパートナー порталь Contact ユーザとして有効化します。		
EnableSelfService	取引先責任者をセルフサービスユーザー Contact として有効化します。		
FindDup	重複するリードを表示します。 Lead		
FollowupEvent	フォローアップ行動を作成します。 Event		
FollowupTask	フォローアップ ToDo を作成します。 Event		
HooversProfile	Hoovers プロファイルを表示します。	<ul style="list-style-type: none"> ・ 取引先 ・ リード 	
IncludeOffline	Connect Offline に取引先レコードを含めます。 Account		
GoogleMaps	Google 地図上で住所を示します。	<ul style="list-style-type: none"> ・ 取引先 ・ 取引先責任者 ・ リード 	
GoogleNews	www.e...o...g...g...o...m...c... を表示します。	<ul style="list-style-type: none"> ・ 取引先 ・ 取引先責任者 ・ リード 	
GoogleSearch	www.e...o...g...g...o...m...c... を表示します。	<ul style="list-style-type: none"> ・ 取引先 ・ 取引先責任者 ・ リード 	
List	オブジェクトのレコードを表示します。	<ul style="list-style-type: none"> ・ 広告グループ ・ キャンペーン ・ ケース ・ 取引先責任者 ・ 契約 ・ Google キャンペーン ・ キーワード ・ リード ・ 商談 ・ 商品 ・ 検索語句 	

		<ul style="list-style-type: none"> • SFGA バージョン • 解決方法 • テキスト広告 • カスタムオブジェクト
LogCall	活動を記録します。	Activity
MailMerge	差し込み印刷を作成します。	Activity
ManageMembers	[メンバーの管理] ページを起動します。	Campaign
MassClose	複数ケースをクローズします。	Case
Merge	取引先責任者をマージします。	Contact
New	新規レコードを作成します。	<ul style="list-style-type: none"> • 活動 • 広告グループ • 納入商品 • キャンペーン • ケース • 取引先責任者 • 契約 • 行動 • Google キャンペーン • キーワード • リード • 商談 • 検索語句 • SFGA バージョン • 解決方法 • ToDo • テキスト広告 • カスタムオブジェクト
NewTask	ToDo を作成します。	Task
RequestUpdate	更新を要求します。	<ul style="list-style-type: none"> • 取引先責任者 • 活動
SelfServSelect	ユーザをセルフサービスユーザとして登録します。	Solution
SendEmail	メールを送信します。	Activity
SendGmail	Gmail で空のメールを開きます。	<ul style="list-style-type: none"> • 取引先責任者 • リード
Sort	価格表内の商品を並べ替えます。	OpportunityLineItem
Share	レコードを共有します。	<ul style="list-style-type: none"> • 取引先

		<ul style="list-style-type: none">・ 広告グループ・ キャンペーン・ ケース・ 取引先責任者・ 契約・ Google キャンペーン・ キーワード・ リード・ 商談・ 検索語句・ SFGA バージョン・ テキスト広告
Submit for Approval	承認を受けるレコードを送信します。	<ul style="list-style-type: none">・ 取引先・ 活動・ 広告グループ・ 納入商品・ キャンペーン・ キャンペーンメンバー・ ケース・ 取引先責任者・ 契約・ 行動・ Google キャンペーン・ キーワード・ リード・ 商談・ 商談商品・ 商品・ 検索語句・ SFGA バージョン・ 解決方法・ ToDo・ テキスト広告
Tab	オブジェクトのタブにアクセスします。	<ul style="list-style-type: none">・ 広告グループ・ キャンペーン・ ケース・ 取引先責任者・ 契約・ Google キャンペーン・ キーワード・ リード・ 商談

		<ul style="list-style-type: none"> ・ 商品 ・ 検索語句 ・ SFCA バージョン ・ 解決方法 ・ テキスト広告
View	レコードを参照します。	<ul style="list-style-type: none"> ・ 活動 ・ 広告グループ ・ 納入商品 ・ キャンペーン ・ キャンペーンメンバー ・ ケース ・ 取引先責任者 ・ 契約 ・ 行動 ・ Google キャンペーン ・ キーワード ・ リード ・ 商談 ・ 商談商品 ・ 商品 ・ 検索語句 ・ SFCA バージョン ・ 解決方法 ・ テキスト広告 ・ カスタムオブジェクト
ViewAllCampaignMembers	すべてのキャンペーンメンバーを表す。	Campaign
ViewCampaignInfluenceReport	[キャンペーンと影響を受ける商談] レポートを表示します。	Campaign
ViewPartnerPortalUser	すべてのパートナーポータルユーザを表示します。	Contact
ViewSelfService	すべてのセルフサービスユーザを表示します。	Contact
YahooMaps	Yahoo! 地図上で住所を示します。	<ul style="list-style-type: none"> ・ 取引先 ・ 取引先責任者 ・ リード
YahooWeather	天気情報を表示します。	Contact / /

\$Api

API URL を参照する際に使用するグローバル差し込み項目の種別です。

使用方法

Enterprise または Partner WSDL から API URL を指定したりセッション ID を返したりするには、ドット表記を使用します。

例

```
t A e e . r r   p r $r p ! i . xxx Enterprise WSDL SOAP エンドポイント。xxx は API のバージョンを示します。たとえば、r r   p r $r p ! i . 2 は Partner WSDL SOAP エンドポイントの式です。
t A e e . r r   p r $r ! i . xxx Partner WSDL SOAP エンドポイント。xxx は API のバージョンを示します。r r   p r $r ! i . 2. は Partner WSDL のエンドポイントの式です。
A n . } o p $ I :セッション.IDs D
```

\$Component

Visualforce コンポーネントを参照する際に使用するグローバル差し込み項目の種別です。

使用方法

Visualforce ページの各コンポーネントには、固有の `Id` 属性があります。ページが表示されるとき、この属性は、ドキュメントオブジェクトモデル (DOM) ID の生成に使用されます。JavaScript で `componentPath.to.Id` を使用して、ページ上の特定のコンポーネントを参照します。`Path.to.Id` は、参照するコンポーネントのコンポーネント階層指定子です。

例

次の JavaScript メソッドは、Visualforce ページ内の `Component` という名前のコンポーネントを参照します。

```
e n u t e o b{ fro f T i v S a( )
et atde o ut fm dg B e m d E t n n o f o m e C. mo gp $ l y u p( ! ' .s s v . ) a=
te t d m ro d g ro E M y T . i H L ;( )
```

次のページマークアップは、`Component` が参照する `Component` コンポーネントを表示します。

```
x t e t t x g p p < a a
x ou u t p d p mo gT l ue p t m i E # sg s "v a' s a / s a
```

コンポーネントがネストされている場合は、完全なコンポーネントパス指定子を使用する必要があります。たとえば、ページが次のような場合:

```
B g l go p d p c < a k a a i =
```

```

e   e   x t e t   t x   ou   u   p   p   < T   / a   >
e ex   e   t B e l g o   p   p o c   < c m   k/a   a   s   >
ex   e   t B   l g o   p   p o c   < c   k/a   a   s   >
      B   l g o   p   p c   <   k/a   a   >
      g   p   p   <   / a   a

```

関数内で次のようにコンポーネントを参照できます。

```

e   e   o   ut   mn   dg   B   m d E   I   .   y   (   )
e   te n   o   @te B C tl   eptes   t   t   check   task   c m .I   s   .   T   s   .   "

```

関連リンク

[JavaScript を使用したコンポーネントの参照](#)

[コンポーネントIDへのアクセスのベストプラクティス](#)

\$ComponentLabel

メッセージに関連付けられている Visualforce ページでコンポーネントのラベルを参照するときに使用するグローバル差し込み項目です。

使用方法

メッセージに関連付けられているコンポーネントのラベルを返します。

例

```

t   t   l   e   p   e   re   s   u   a }   m i a s s g   = " a s s t v   a "   !   ss   sa "   >
t   t x e   f u   au   p   l   up   t e n   m   o   a o e }   b   l   v   e p   l a "   ! : l e s s o r   d   :   a "   s   y "   " a   > /
t   t x e   f u   au   p   l   t u p   < t   a   l   v   l o r o   !   c s s   . :   i c " a   s   y "   " a   > /
t   d   l   p   <   / a   i a s a >

```

\$CurrentPage

現在の Visualforce ページまたはページ要求を参照するときに使用するグローバル差し込み項目の種別です。

使用方法

このグローバル変数を Visualforce ページで使用して現在のページ名 (`CurrentPage.Name`) または現在のページの URL (`CurrentPage.Url`) を参照します。ページ要求パラメータとその値を参照するには、`parameterName` を使用します。`parameterName` は、参照する要求パラメータです。

例

```

t   n   t e g n d p r d   p o r   id   x n c   a u   a c c   a   a   = "   "   >
x   t e t B e l g o   p   d p   c l l   < t e k r a   r } a   i   m H $ "   U   !   s   a   s N   a "   >
t   t e n t o b l   o   g e n   n   d u t   Y a c   m   o u   b r c c   !   a   .   a   a <   .   > /
ex   t   e   o   e n   n   o   n   r   o   c   Y   p   ' s a   i a   s   .
      B   l g o   p   p   c   <   k/a   a   >
x t   e   d t   l   b p t e j n e } t   < o u t r   a c   d   a   l   = "   !   a f   "   L   a   s   = "   s " a   > /
x e t   t   r   l   t   d   p   t   n   t   A   <   a   p   a   s   i s   = "   o   v   i   i   s   "   > /
t   bu   j   c   { ! $CurrentPage.parameters.relatedId }
      g   p   p   <   / a   a

```

\$Label

Visualforce ページのカスタム表示ラベルを参照する際に使用するグローバル差し込み項目の種別です。

使用方法

この式を Visualforce ページで使用し、カスタム表示ラベルにアクセスします。アプリケーションサーバでページをエンドユーザのブラウザに表示するよう構成している場合、返される値はコンテキストユーザの言語設定によって異なります。返される値は、優先順に次のいずれかとなります。

1. ローカル翻訳のテキスト
2. パッケージ翻訳のテキスト
3. マスタ表示ラベルのテキスト

例

e	e	e	e	g	p	p	<	a	à	a	ss	s	i	=	y	"		
n	r	e	u	t	g	h	<	o	a	l	=	"						
n	r	m	m	t	e	b	l	n	tr	u	l	h	y	!	p	ai	s	
e	u	r	m	m	t	e	b	l	n	tr	u	l	h	y	!	p	ai	s

\$Label.Site

Visualforce ページの標準サイト表示ラベルを参照する際に使用するグローバル差し込み項目の種別です。すべての標準表示ラベルと同様、テキストはユーザの言語および地域に基づいて表示されます。

使用方法

この式を Visualforce ページで使用し、標準サイト表示ラベルにアクセスします。アプリケーションサーバでページをエンドユーザのブラウザに表示するよう構成している場合、返される値はユーザの言語と地域によって異なります。

Salesforce には、次の表示ラベルがあります。

表示ラベル	メッセージ
authorization_required	認証が必要です
bandwidth_limit_exceeded	帯域幅の制限を超えていません
change_password	パスワードの変更
change_your_password	パスワードを変更する
click_forget_password	パスワードを忘れた場合は、[パスワードを忘れた場合]をクリックしてパスワードをリセットします。
communityNickname	ニックネーム
confirm_password	確認用パスワード
down_for_maintenance	<i>{0}</i> は、メンテナンスのため停止しています

表示ラベル	メッセージ
email	メール
email_us	メールを送信
enter_password	パスワードをお忘れですか? 以下にユーザ名を入力してください。
error	エラー: {0}
error2	エラー
file_not_found	ファイルが見つかりません
forgot_password	パスワードを忘れた場合
forgot_password_confirmation	パスワード忘れの確認
forgot_your_password_q	パスワードをお忘れですか?
get_in_touch	連絡を取る必要がある場合は、{1}してください。
go_to_login_page	ログインページに移動
img_path	/img/sites
in_maintenance	メンテナンスによる停止
limit_exceeded	制限数を超えるました
login	ログイン
login_button	ログイン
login_or_register_first	このページにアクセスするには、最初にログインまたは登録が必要です。
logout	ログアウト
new_password	新しいパスワード
new_user_q	新規ユーザですか?
old_password	現在のパスワード
page_not_found	ページが見つかりません
page_not_found_detail	ページが見つかりません: {0}
password	パスワード
passwords_dont_match	パスワードが一致しません。
powered_by	Powered by
register	登録
registration_confirmation	登録の確認
site_login	サイトログイン
site_under_construction	サイト構築中

表示ラベル	メッセージ
sorry_for_inconvenience	ご不便をお掛けして大変申し訳ございません。
sorry_for_inconvenience_back_shortly	ご不便をお掛けして大変申し訳ございません。すぐに復旧いたします。
stay_tuned	このまましばらくお待ちください。
submit	送信
temp_password_sent	仮のパスワードがメールで送信されました。
thank_you_for_registering	ご登録ありがとうございます。仮のパスワードがメールで送信されました。
under_construction	<i>{0}</i> は構築中です。
user_registration	新規ユーザの登録
username	ユーザ名
verify_new_password	新しいパスワードの確認

例

```

e n   e e e   g   p   p   <   a   à
e   r   e   tg   p   opMn   <   o   a   aßs   s   i   =y   "
e   u   remte   g   h   <   s   l   ="
e   u   remte   t   b l   ve   t   m$n   rd   =yp!p   ai   s .   ss a   s _ "
e   n   e e e   g   p   p   <   /a   à

```

\$ObjectType

標準オブジェクトまたはカスタムオブジェクト(取引先、ケース、商談など)およびその項目の値を参照する際に使用するグローバル差し込み項目の種別です。

使用方法

オブジェクトを指定するには、ドット表記を使用します(例: `{ } . c $!`)。! 〇 . y . s a と い う 構文を使用して、そのオブジェクトの項目を選択することもできます。

例

次の例では、取引先の 項目の表示ラベルを取得します。

```
b   tej An   ecou es_ !} O m l   y .   E   s NL   a.   a
```

動的参照を使用しても、`c $` を介してオブジェクトに関する情報を取得できます。たとえば、
b tej An ecou \$!} O m y . のようになります。' y

\$ObjectType で使用できるオブジェクトスキーマ詳細

グローバル変数を使用して、組織のオブジェクトに関するスキーマ情報にアクセスします。たとえば、オブジェクトの名前、ラベル、アクセシビリティにアクセスします。

\$ObjectType で使用できる項目スキーマ詳細

グローバル変数を使用すると、組織のオブジェクトに関するさまざまなスキーマ情報にアクセスできます。たとえば、オブジェクトの項目の名前、表示ラベル、データ型の参照に使用します。

関連リンク

\$ObjectType を使用したスキーマ詳細への動的参照

\$ObjectType で使用できるオブジェクトスキーマ詳細

グローバル変数を使用して、組織のオブジェクトに関するスキーマ情報にアクセスします。たとえば、オブジェクトの名前、ラベル、アクセシビリティにアクセスします。

Apex の DescribeResult を使用して入手できる情報は、オブジェクトを使用して入手できる詳細のサブセットです。次の表に、\$ObjectType で使用できる属性を示します。

名前	データ型	説明
Id	String	この属性は、単独では使用できません。 \$ObjectType の後に項目メンバ変数名を指定し、その後に項目属性を指定する必要があります。次に例を示します。 { e te b tej An ecous Id ! } final y . is NL a a
Label	String	この属性は、単独では使用できません。 \$ObjectType の後に項目セット名を指定し、反復コンポーネントで使用する必要があります。次に例を示します。 e e ex t r p p < a a e { e lte u bt terj e ac e \$ Id at t o \$ y mp p a r f v =! a >
RecordType	String	オブジェクトの 3 文字のプレフィックスコード。レコード ID はオブジェクト種別を示す 3 文字のコードが先頭に付けられます。たとえば、取引先には というプレフィックスが付けられ、商談には というプレフィックスが付けられます。 e te b j c \$ は 安定したプレフィックスを持つオブジェクトに値を返します。安定したプレフィックスまたは予測可能なプレフィックスを持たないオブジェクトデータ型については、項目は空白です。これらのコードに依存するページは、前方互換性を確保するために、このオブジェクト種別を決定する方法を使用できます。

名前	データ型	説明
label	String	オブジェクトの表示ラベル。多くの場合オブジェクト名と一致します。たとえば、医療分野の組織では、Account の表示ラベルを Patient に変更する可能性があります。この表示ラベルは、Salesforce ユーザインターフェースで使用されるものと一致します。
labelRule	String	オブジェクトの表示ラベル(複数形)。多くの場合オブジェクト名と一致します。たとえば、医療分野の組織では、Account の複数の表示ラベルを Patient に変更する可能性があります。この表示ラベルは、Salesforce ユーザインターフェースで使用されるものと一致します。
name	String	オブジェクトの名前。
blcc	Boolean	現在のユーザがこのオブジェクトを参照できる場合は s 、できない場合は f 。 s a
blc	Boolean	現在のユーザがオブジェクトを作成できる場合は s 、できない場合は f 。 s a
blmc	Boolean	項目がカスタムオブジェクトの場合は s 、標準オブジェクトの場合は f 。 s a
blrm	Boolean	現在のユーザがオブジェクトを削除できる場合は s 、できない場合は f 。 s a
bmblg	Boolean	現在のユーザがオブジェクトを同じ型の他のオブジェクトとマージできる場合は s 、できない場合は f 。 s a
bnql	Boolean	現在のユーザがオブジェクトをクエリできる場合は s 、できない場合は f 。 s a
brbl	Boolean	現在のユーザがオブジェクトを検索できる場合は s 、できない場合は f 。 s a
bdl	Boolean	現在のユーザがオブジェクトを復元できる場合は s 、できない場合は f 。 s a
dpbl	Boolean	現在のユーザがオブジェクトを更新できる場合は s 、できない場合は f 。 s a

関連リンク

[\\$ObjectType を使用したスキーマ詳細への動的参照](#)

\$ObjectType で使用できる項目スキーマ詳細

グローバル変数を使用すると、組織のオブジェクトに関するさまざまなスキーマ情報にアクセスできます。たとえば、オブジェクトの項目の名前、表示ラベル、データ型の参照に使用します。

を使用して入手できる情報は、Apex の DescribeResult である。オブジェクトを使用して入手できる詳細と似ていますが、そのサブセットです。次の表に、グローバル変数で使用できる属性を示します。

名前	データ型	説明
g_h	Long	可変長項目(バイナリ項目も含む)の最大サイズをバイトで指定。
croclum	aString	この項目に指定された数式。
Schema.sObjectField(string 文)	Schema.sObjectField	これが運動項目である場合は、制御項目。
urflum	String	数式が使用されていない場合に、この値に指定されるデフォルト値。
i_i	Integer	項目に指定された最大桁数。数値以外の項目の場合はゼロ。
p_T_i	String	項目レベルのヘルプの内容。詳細は、Salesforce オンラインヘルプの「項目レベルのヘルプの定義」を参照してください。
	aString	Salesforce ユーザインターフェースの項目の隣に表示されるテキストラベル。このラベルはローカライズが可能です。
h	Integer	文字列項目において、Unicode 文字での最大サイズを指定(バイトではないことに注意)。
mc	String	項目の名前。
	aString	Apex に使用される項目名。
L_cap_k	List<Schema.PicklistEntry>	項目の選択リストアイテムのリスト。項目が選択リストでない場合は空のリスト。
φ	Double	データ型 double の項目の場合は、小数点の右側と左側の両方をあわせた(ただし小数点自体は含まない)、格納可能な最大桁数。
o_f_c_T	List<Schema.sObjectType>	この項目の親オブジェクトのリスト。 g_r_u 属性が 1 の場合は、リストに複数のエントリがあります。そうでない場合は、エントリは 1 つのみです。
in_p	String	リレーションの名前。リレーションとリレーション名についての詳細は、『Force.com SOQL and SOSL Reference』の「リレーション名について」を参照してください。
rd_r_h_p	Integer	項目が子の場合は 1、子でない場合は 0。リレーションとリレーション名についての詳細は、『Force.com SOQL and SOSL Reference』の「リレーション名について」を参照してください。
c	String	データ型 double の項目の場合は、小数点の右側の桁数。小数点の右側に余分な桁がある場合は、切り捨てられます。

名前	データ型	説明
o p t	s Schema.SOAPType (string として)	項目のデータ型に応じた、SoapType enum 値の1つ。詳細は、『Force.com Apex コード開発者ガイド』の「Schema.SOAPType Enum 値」を参照してください。
b j l d c	s Schema.sObjectField (string として)	この項目への参照。
p y	S Schema.DisplayType (string として)	項目のデータ型に応じた、DisplayType enum 値の1つ。詳細は、『Force.com Apex コード開発者ガイド』の「Schema.DisplayType Enum 値」を参照してください。
bl cc	a Boolean	現在のユーザがこの項目を参照できる場合はtrue、できない場合はfalse。
u o b r m	a Boolean	項目がAuto Number 項目の場合はtrue、そうでない場合はfalse。
l lu d c c	a Boolean	項目がカスタム数式項目の場合はtrue、そうでない場合はfalse。
t d l c c	sD a Boolean	親オブジェクトの削除時に子オブジェクトが削除される場合はtrue、削除されない場合はfalse。
c	s a Boolean	項目が大文字と小文字を区別する場合はtrue、区別しない場合はfalse。
r bl c	Boolean	現在のユーザが項目を作成できる場合はtrue、できない場合はfalse。
u o mc	s Boolean	項目がカスタム項目の場合はtrue、標準オブジェクトの場合はfalse。
d e t l u d r f c	Boolean	作成時に項目がデフォルト値を受け取る場合はtrue、受け取らない場合はfalse。
d t d l p c k	a Boolean	選択リストが連動選択リストの場合はtrue、そうでない場合はfalse。
r l d I	Boolean	項目が外部 ID として使用されている場合はtrue、そうでない場合はfalse。
l r bl f i	Boolean	項目をステートメントの検索条件の一部として使用できる場合はtrue、そうでない場合はfalse。
r o q bl p	Boolean	項目が SOQL クエリの句に含まれる場合はtrue、含まれない場合はfalse。
l r on h d	F Boolean	項目が HTML 用に書式設定されており、HTML に表示されるように符号化する必要がある場合はtrue、そうでない場合はfalse。この属性に対してtrueである項目の例の1つ

名前	データ型	説明
d_o_o_u	Boolean	は、ハイパーアクションのカスタム数式項目です。ある1つの例は、IM_Eテキスト関数があるカスタム数式項目です。
ind	Boolean	p メソッドでレコードを指定するために項目を使用できる場合は 、使用できない場合は f 。 項目が名前項目の場合は 、そうでない場合は f 。このメソッドは、標準オブジェクトの名前項目 (Account オブジェクトの ccmm など) やカスタムオブジェクトの名前項目を識別するために使用します。Contact オブジェクトのように m と m 項目が代わりに使用される場合を除き、オブジェクトは名前項目を1つのみ持つことができます。
n_n_m_g	Boolean	項目が複数のデータ型のオブジェクトを親として持つことが可能な場合は、 。たとえば、ToDo は 取引先責任者 リード [h] (h [I]) 項目と 商談 取引先 [I] (h [I]) 項目の両方を持つことができ、いずれかのオブジェクトが特定 ToDo レコードの親になる可能性があるため、この属性に対して になります。それ以外の場合、この属性は f になります。
ll_b_l	Boolean	項目を空白にできる場合は 、できない場合は f 。 項目に項目の権限を指定できる場合は 、そうでない場合は f 。
m_m_b_l_p	Boolean	子オブジェクトから参照されるため親オブジェクトを削除できない場合は 、削除できる場合は f 。
er_t_r_d_l_c	Boolean	項目が制限つき選択リストの場合は 、そうでない場合は f 。
r_r_t_d_l_c_c	Boolean	項目上でクエリをソートできる場合は 、できない場合は f 。
ro_b_l	Boolean	項目の値を一意にする必要がある場合は 、そうでない場合は f 。
d_b_l_p	Boolean	次のいずれかの場合は 。 <ul style="list-style-type: none">現在のユーザが項目を編集できるカスタムオブジェクトでの主従関係項目の子レコードの親を他の親レコードに変更できる

名前	データ型	説明
re te a q r d M i Boolean et	e s s aR r u	上記以外の場合は f 。 詳細オブジェクトへの書き込みに親の参照・更新共有ではなく参照共有が必要な場合は、 s a

関連リンク

[\\$ObjectType を使用したスキーマ詳細への動的参照](#)

\$Organization

組織プロファイルを参照する際に使用するグローバル差し込み項目の種別です。Organization 差し込み項目を使用して、組織の市区郡、Fax、ID、その他の詳細情報を参照します。

使用方法

組織の情報にアクセスするには、ドット表記を使用します。例:

```
tn z ret n tg o } r $ ! o i ai a . s  
tn z ret n g o! $ ! o i ai a . s a
```

Organization 差し込み項目は、組織情報の一部として現在 Salesforce に保存されている値から値を取得します。

```
n z r n g no } $ ! は選択リスト値であるため、カスタム項目、入力規則、および Visualforce 式で  
A I IC などの選択リスト関数で使用する必要があります。
```

例

z r n g o \$ グローバル変数を使用してアクセスできる値は、次のとおりです。

```
tn z r n g o d $ I ! o i ai a . a  
tn z er n g jo $m ! o i ai a N a  
tn z r n g on } $o ! o i aiD a v i $  
tn z ret n tg o } r $ ! o i ai a . s  
tn z r tn g jo $ c ! o i ai a i y  
tn z ret n t g o! $ ! o i ai a . s a  
tn z r t t g o o o l $ do !co i ai a . s P a  
tn z r nt g n o } our $ C ! o i ai a . y  
tn z r nx g o $ ! o i ai a E a  
tn z re n g n o! o $ h ! o i ai a . P  
tn z ren g o A d $ g } o lm o p p ai a . G D s i a  
tn z r n g no } $ ! o ik a i u a i i S
```

\$Page

Visualforce ページを参照する際に使用するグローバル差し込み項目の種別です。

使用方法

この式を Visualforce ページで使用し、別の Visualforce ページにリンクします。

例

```

g p p < a à
d h h k<1 L j < 1 y
enfu u e p le t u p e < gkjr L ign w
t n etl hne To or hgsi s pia a! Pa. Pa >
nou u p p < ka L i a a.
g p p < /a à

```

\$Profile

現在のユーザを参照するときに使用するグローバル差し込み項目の種別です。Profile 差し込み項目を使用して、ライセンスの種類や名前など、ユーザのプロファイルに関する情報を参照します。

使用方法

組織の情報にアクセスするには、ドット表記を使用します。

e Visualforce では次の f 値は使用できません。

```

c p L i s y
U s y

```

例

```

r ol d f $ ! iP .
r ol } mf $ ! iP N a

```

\$Resource

Visualforce ページで、既存の静的リソースを名前で参照するときに使用するグローバル差し込み項目の種別です。また、関数でリソース差し込み項目を使用して、静的リソースアーカイブ内の特定のファイルを参照することもできます。

使用方法

} &u c \$を使用して、既存の静的リソースを参照します。形式は

\$nameOfResources です

e t e (例:) c &g I T) R s . s a

例

次の Visualforce コンポーネントは、静的リソースとしてアップロードされ、m g I T という名前が付けられた画像ファイルを参照します。

```

e enge p te &u < w áce &g alIT! R s h g s h 'a i = " " 50 = " " 50 /

```

z アーカイブ内(r または _ ファイルなど)のファイルを参照するには、関数を使用します。最初のパラメータには、そのアーカイブをアップロードしたときに指定した静的リソース名を、第2パラメータには、アーカイブ内の目的ファイルへのパスを指定します。たとえば、次のとおりです。例:

```

e nerd p e e t< &u á ac U$ L ! R F Q p R s Z . s i , =
e B m g u llw á g d ti p ' sa h / i g s h . h " 'j = " " 50 = " " 50 /

```

また、動的参照を使用して、静的リソースを参照することもできます。たとえば、`} } o o $ pp] の R s` ようになります（`t pp` プロパティまたは `o pp` メソッドがページのコントローラにある場合）。

関連リンク

[Visualforce ページのスタイル設定](#)

\$SControl

既存のカスタムSコントロールを名前で参照する際に使用するグローバル差し込み項目の種別です。この種別の差し込み項目から、Sコントロールが実行されるページへの URL が作成されます。

! 重要: Sコントロールは、Visualforce ページに置き換えられました。組織で以前に Sコントロールを使用していない場合は、作成できません。既存のSコントロールに影響はありません。今後も編集できます。

使用方法

名前で既存の Sコントロールにアクセスするには、ドット表記を使用します。

例

次の例は、Visualforce ページにある HelloWorld という名前の Sコントロールにリンクする方法を示します。

```
new user pp < a >
t am or loe lce & ! listHelloWorld > oc Hlou u p s p - < ka L i
g p p < /a >
```

ページにSコントロールを埋め込むだけなら、\$SControl 差し込み項目がなくても `or lo pc` タグを使用 `as` できます。例:

```
new user pp < a >
t er lo por lo c < mll or&ad
g p p < /a >
```

\$Setup

「階層」型のカスタム設定を参照する際に使用するグローバル差し込み項目の種別です。

使用方法

ドット表記を使用して階層カスタム設定およびその項目値にアクセスするために使用します。たと

えば、`erw ep$tpn flnc ho s . Cp _B c_s` です。

階層カスタム設定では、次の 3 つの異なるレベルの値を使用できます。

1. すべてのユーザのデフォルト値である組織
2. 組織の値を上書きするプロファイル
3. 組織およびプロファイルの値を上書きするユーザ

Salesforce では、実行中のユーザの現在のコンテキストに基づいて、このカスタム設定に適した値が自動的に判別されます。

このグローバル変数を使用する Visualforce ページでは、「リスト」型のカスタム設定は使用できません。リストカスタム設定には、Apex でアクセスできます。

例

次の例では、ユーザ設定に応じて入力項目に拡張ヘルプメッセージを条件付きで表示する方法を示しています。

```

x etn e { u ldp l < a >
xt te neuu p d t epl sath ao p g P a n s c s iH a i " " >
e {n er dr de A u erw epSt p m h !ho }S . Cp _B c_H S " " >
c e te tr E whg dt du raf n mo e i ket dud r so i mai fl h s , s a
t t le do o w of oe ro p fmu fk s a
xt te nou u p < /a Pa>
g p p < /a >

```

カスタム設定の組織レベルが **標準** に設定されていると、デフォルトで拡張ヘルプメッセージがユーザに表示されます。個々のユーザがヘルプメッセージの表示を希望しない場合は、そのユーザのカスタム設定を **カスタム** に設定して、組織(またはプロファイル)の値を上書きすることができます。

\$Site

現在の Force.com サイトを参照するときに使用するグローバル差し込み項目の種別です。

使用方法

現在の Force.com サイトの情報にアクセスするには、ドット表記を使用します。次のサイト項目のみを使用できます。

差し込み項目	説明
\$Site.Name	現在のサイトの API 名を返します。
\$Site.Domain	組織の Force.com ドメイン名を返します。
\$Site.CustomWebAddress	現在のサイトの [カスタム アドレス] 項目の値を返します。
\$Site.OriginalUrl	このページがサイトに指定されたエラーページである場合は、元の URL を返し、そうでない場合は null を返します。
\$Site.CurrentSiteUrl	現在の要求に対するサイト URL の値(例: http://www.salesforce.com や http://www.salesforcepx.com/fmc/c/s/pf/ など)を返します。
\$Site.LoginEnabled	現在のサイトがログインが有効なポータルと関連付けられている場合は true を返し、そうでない場合は false を返します。
\$Site.RegistrationEnabled	現在のサイトがセルフ登録対応のカスタマーポータルと関連付けられている場合は true を返し、そうでない場合は false を返します。
\$Site.IsPasswordExpired	認証ユーザの場合、現在ログインしているユーザのパスワードの有効期限が切れている場合、true を返します。認証されていないユーザの場合は、false を返します。
\$Site.AdminEmailAddress	現在のサイトの [サイトの管理者] 項目の値を返します。

差し込み項目	説明
\$Site.Prefix	現在のサイトの URL パスのプレフィックスを返しますたとえば、サイト URL が <code>nco_rfmc_rcet_pn_y</code> である場合、 <code>pn</code> がパスのプレフィックスです。プレフィックスが定義されていない場合、または、ページにカスタム Web アドレスを使用してアクセスした場合には、Null を返します。
\$Site.Template	現在のサイトに関連付けられたテンプレートを返します。テンプレートが指定されていない場合、デフォルトテンプレートを返します。
\$Site.ErrorMessage	現在のページがサイトに指定されたエラーページで、エラーがある場合は、現在のページのエラーメッセージを返し、そうでない場合は空の文字列を返します。
\$Site.ErrorDescription	現在のページがサイトに指定されたエラーページであり、エラーがある場合は、現在のページのエラーの説明を返し、そうでない場合は空の文字列を返します。
\$Site.AnalyticsTrackingCode	サイトに関連付けられている追跡コード。このコードは、Google Analyticsなどのサービスによって、サイトのページリクエストデータを追跡するために使用されます。

例

次の例では、`$_Tp` 差し込み項目の使用方法を示しています。

```

gl pt AlpoJ m <nt o acer exi m="ef o dir hi la i fi a" H a" s'a
net ee dtrd et l r hu s a a Sys s=" " >
e t e t e t e h l mT rt od etpl -auts elt ro av i fsh ya s& y $ >-
x t e t ne em po e tel pm et t pa } l smi $ Tp =a ! i S . 'a >
xe end p bmdo < a i =a " y
x ro m p f < a >
e n eno dmm x lt uBe < b boIc ria v k <@th - a S 'a
n w olm dc{oc ome o k o xi e="ph} heJn r$ciua=a=1 ! ' Pa fiP s ; ' s a"
br < > /
br < > /
e et n r c < >
e xtet txe ou u p l up t et e a e on ph y h a'dY ia i a s a & a .
n e t uhrat fu ffk < / >
e et n r c < / > /
br < / > /
br < / > /
xe n d ro m p f < /a i >
x t a om po p < /a > $ i >
g p p < /a >

```

\$System.OriginDateTime

リテラル値 1900-01-01 00:00:00 を表すグローバル差し込み項目です。

使用方法

このグローバル変数は、日付/時間オフセット計算を実行する場合や、日付/時間項目にリテラル値を割り当てる場合に使用します。

例

次の例では、1900年1月1日以降の経過した日数を計算します。

```
e t @ r m gW $ N @ ( ) - sT@Di ia
```

\$User

現在のユーザを参照するときに使用するグローバル差し込み項目の種別です。User 差し込み項目は、別名、役職、ID など、ユーザに関する情報を参照します。

使用方法

現在のユーザの情報にアクセスするには、ドット表記を使用します。例:

```
e A I @t le T e ! $r { } ss Ti s a i s ) F s a )
```

例

次の例は、現在のユーザの会社名と、現在のユーザの状況を表示します (Boolean 値を返します)。

```
tn t or n h g u C o < g p p > a a ai < a b > Pa
e e e w n exhu A T g ip si s y
{ e t n rre n h p n o T m c < m o p > f h a y a i s
e e ur e m : o } m C a s pi s U ! s p . Na < a > /
e et ur I p h c < s s v a ? rA }
g p p < / a a >
```

\$User.UITheme および \$User.UIThemeDisplayed

これらのグローバル差し込み項目は、指定された Web ページ上でユーザに表示される Salesforce のデザインを識別します。

これら 2 つの変数は、\$User.UITheme がユーザに表示すべきデザインを返し、\$User.UIThemeDisplayed が実際に表示されるデザインを返すという点で異なります。たとえば、ユーザは[新しいユーザインターフェースのテーマ]のデザインを表示するよう設定された権限を持っている場合がありますが、そのデザインをサポートしていないブラウザを使用していると、Internet Explorer などで \$User.UITheme が異なる値を返す場合があります。

使用方法

これらの変数を使用して、Salesforce Web ページをユーザに表示するときに使用される CSS を特定します。どちらの変数も、次の値のいずれかを返します。

— 古い Salesforce テーマ

— Spring '10 以前に使用されていた Salesforce テーマ

— Salesforce カスタマーポータルのテーマ

b ro • — Salesforce AppExchange のテーマ
 m h T— Spring '10 で導入された、現在の Salesforce テーマ

例

次の例は、ユーザのテーマを基にさまざまなレイアウトを表示する方法を示したものです。

```
x      g p p < a a
t      teB l goe pt l p e <fn kma Cd r a e i re'e y m I h$ TmU h! Us 2T . == ' " >
ex     et e te h id h m h /si s ...
          B l go p p c < k/a a >
          ...
x      g p p < /a a
t      teB l goe pt l p e <fn kma Cd r a e i re'e y m I h$ TmU h! Us T . == ' 3 " >
ex     et e ten h h m h i/si s ...
          B l go p p c < k/a a >
          ...
g p p < /a a
```

\$UserRole

現在のユーザのロールに関する情報を参照するときに使用するグローバル差し込み項目の種別です。Role 差し込み項目は、ロール名、説明、ID などの情報を参照します。

使用方法

現在のユーザのロールに関する情報にアクセスするには、ドット表記を使用します。

Visualforce では次の 値は使用できません。

```
A      te Amro nccou rcc s a ssF o
oA     • teC Amro nccou rcc a ssF o
ten    ro A  teppAmro nccou rcc i y ssF o
rol    • p P a y
```

例

```
t re lo do B $d d U f s IRL . s ai i y
```

関数

レコードのデータの変換、計算の実行、Visualforce 属性の値の指定を行うには、関数を使用します。

関数は、評価する Visualforce 式で使用する必要があります。Visualforce ページでは、次の関数を使用できます。

日付および時間関数



メモ: API バージョン 20.0 未満の Visualforce ページでは、数式の date/time データ型が正しく評価されない場合があります。誤って date 型として解釈される可能性があります。

関数	説明	使用
DATE	入力した年、月、および日の値から日付値を返します。Salesforce では、桁の年、month に 2 桁の月、day に数式項目内の DATE 関数の値が、う 2 桁の日を指定します。 るうでない年の 2 月 29 日などのよう無効な日付である場合に、詳細ページにエラーを表示します。	E(yeaE,month,day)。year に 4
DATEVALUE	日付/時間またはテキストの式に対して日付型の値を返します。	E DATE expression 。expression に、日付/時間かテキスト値、差し込み項目、または式を指定します。
DATETIMEVALUE	年、月、日、および GMT 時刻値を返します。	E DATETIME expression 。expression に、日付/時間かテキスト値、差し込み項目、または式を指定します。
DAY	月の日付を、1 から 31 までの数値形式で返します。	date 。 date を、日付項目や TODAY() などの値で指定します。
MONTH	数値形式で指定された日付の中の、1(1 月) から 12(12 月) までの数値を返します。	M date 。 date に、返される月を含む日付を表す項目または式を指定します。
NOW	現在の日付/時間を返します。 関数は、現在の日付と時刻を GMT タイムゾーンで返します。た とえば、の場合: W N O ()	W N O ()
	{ e t do et nd d T m : sa ' a á i s } n o luJ M 2 : 2: M 21 1T 1 1 6 G 08	
	これは、次の出力を生成します。	
	e t do et nd d T m : sa ' a á i s n o luJ M 2 : 2: M 21 1T 1 1 6 G 08	
	ヒント	
	<ul style="list-style-type: none"> 括弧を削除しないでください。 括弧は空白のままにしておきます。値を入れる必要はありません。 数値および 関数と一緒に計算演算子または減算演算子を使用すると、異なる日付と時間が返されます。たとえば、は、現在から 5 日先の日付と時間を計算します。 日付時間項目を使用する場合は、TODAY を使用してください。 	N O W + N O () 5

関数	説明	使用
TODAY	<p>現在の日付を日付データ型で返します。</p> <p>A 関数は、現在の日付を返します。たとえば、次のマークアップの場合:</p> <pre>e t { do d : T A } : Tsa ' ias D ! O Y ()</pre> <p>これは、次の出力を作成します。</p> <pre>e t do d n : T o luJ M 2 sa ' ias 1 : : M 2 T 00 00 0G 08</pre> <p>ヒント</p> <ul style="list-style-type: none"> • 	T D O Y ()

関数	説明	使用
PRIORVALUE	項目の前の値を返します。	A I E field V R MR (

論理関数

関数	説明	使用
AND	値がすべて真である場合はTRUEを返し、1つ以上の値が偽である場合はFALSEを返します。 次のマークアップでは、価格と数量が1未満の場合は、「Small」という単語が表示されます。納入商品の価格または数量が1より大きい場合、この項目は空白になります。	lg l o dg cD 2N (を 返し、1つ以上の値が偽である場合 使用し、logical1,logical2,... は FALSE を返します。 a,i
	{ e A r I c ! E N i <P t t n u Q i &1 y ,) n l l m } l h " S " a ,)	
	Visualforce マークアップでは、単語 の代わりに を使用できます。 たとえば、 c D N i <P A e A r Q c i &1 y ,) i <P t t n u e 是Q c i &1 y ,) i <P t t n u と同じです。 i &1 y ,)	&&
	• value_if_true 式と value_if_false 式のデータ型は 同じにしてください。	
CASE	与えられた式を一連の値と照合します。式が値と一致する場合は、対応する結果を返します。どの値とも一致しない場合は、u を返します。	c E expression value1 S(result1 value2 result2 else_result . expression には、 指定された値と比較する項目または 値を指定します。value と result の各 値は、result を返すために等しくする 必要があります。else_result に は、式がどの値とも一致しない場合 に返される値を指定します。
IF	式が真か偽かを判断します。真の場合は与えられた値を返し、偽の場合には別の値を返します。 次のマークアップは、商談の 項目が true に設定されて	I logical_test F (value_if_true value_if_false . logical_test に、評価する式を指定します。 value_if_true に、式が真の場合に 返す値を指定します。

関数	説明	使用
e	Visualforce マークアップでは、単語 の代わりに <code><P></code> を使用できます。 たとえば、 <code>c i a l y</code> は <code>c i a l y</code> と同じです。	<code>.sN , s —))</code> <code>(<P> ,)</code> <code>((<P>))</code>

算術関数

関数	説明	使用
ABS	数値の絶対値を計算します。数値の絶対値とは、正または負の記号がない数値のことです。	<code>ABS(number)</code> 。それぞれの <code>number</code> に、差し込み項目、式、または削除する記号を持つその他の数値を指定します。
CEILING	数値を、もっとも近い整数に切り上げます。	<code>CEIL(number)</code> 。 <code>number</code> を切り上げられた値に置き換えます。
EXP	e を指定した指数まで掛け合わせた値を返します。	<code>EXP(number)</code> 。 <code>number</code> に、数値項目または <code>5</code> などの値を指定します。
FLOOR	もっとも近い整数に切り捨てられた数値を返します。	<code>FLOOR(number)</code> 。 <code>number</code> に、数値項目または <code>5.245</code> などの値を指定します。
LN	指定した数の自然対数を返します。自然対数は、定数 e の値 <code>2.71828182845904</code> に基づきます。	<code>LN(number)</code> に、自然対数を求める項目または式を指定します。
LOG	数値の基數 <code>10</code> の対数を返します。	<code>LOG(number)</code> に、基數 <code>10</code> の対数の計算元項目または式を指定します。
MAX	数値のリストの中で最大の数値を返します。	<code>MAX(number1, number2, ...)</code> を使用し、 <code>number</code> に最も高い数を取得する項目または式を指定します。
MIN	数値のリストの中で最小の数値を返します。	<code>MIN(number1, number2, ...)</code> を使用し、 <code>number</code> に項目または式を指定します（これらの項目または式から、最も低い数が取得されます）。
MOD	数値を指定した除数で除算した後の剰余を返します。	<code>MOD(number, divisor)</code> 。 <code>number</code> を除算する項目または式で置き換え、 <code>divisor</code> には、約数として使用する数値を指定します。

関数	説明	使用
ROUND	指定した数値にもっとも近い数値を返します。新しい数値は、指定した桁数で制限します。	<i>number num_digits</i> 。 <i>number</i> に、値を丸める項目または式を指定します。 <i>num_digits</i> に、値を丸めるときに考慮する小数点以下の桁数を指定します。
SQRT	指定された数値の正の平方根を返します。	<i>number</i> 。 <i>number</i> に、 ^R 平方根にする項目または式を指定します。

テキスト関数

関数	説明	使用
BEGINS	テキストが特定の文字列で始まるかどうかを判断し、該当する場合は TRUE を返します。偽である場合は 文字または項目を指定します。 次のマークアップは、商談の 項目が文字列「Closed」で開始する場合、true を返します。 標準フェーズ名「Closed Won」(商談 成立)と「Closed Lost」(不成立)の場合、どちらも true が返されます。	<i>text compare_text</i> 。 <i>text</i> , <i>compare_text</i> に、比較する
BR	テキスト文字列に改行を挿入します。	^R ()
CASESAFEID	15 文字の ID を大文字と小文字を区別しない 18 文字の ID に変換します。	<i>id</i> 。 <i>id</i> にオブジェクトの ID を指定します。
CONTAINS	テキストの 2 つの引数を比較し、最初の引数に 2 番目の引数が含まれる場合には TRUE を返します。含まれない場合は FALSE を返します。	<i>text compare_text</i> 。 <i>text</i> に、 <i>compare_text</i> の値を含む
	この例は、 スタムテキスト項目の内容を確認	テキストを指定します。

関数	説明	使用
	<p>し、「part」という言葉の含まれる商品に対して「Parts」を返します。それ以外の場合は、「Service」と表示します。</p> <pre>{ t n rt t on e or cu rdou pp c !F (ip§ a(i y p y _ e e t r r p " " a ",) Ea , r } c " v s ")</pre> <p>この関数は、大文字と小文字を区別するため、<code>compare_text</code> の値には必ず正しい大文字を使用してください。</p>	
FIND	テキスト文字列中での指定した文字列の位置を数値で返します。	I search_text text N (start_num] 。search_text に検索対象となる文字列を指定します。 text には、検索対象となる項目または式を指定します。start_num には、左から右に向かって検索を開始する文字数を指定します。
GETSESSIONID	ユーザのセッションIDを返します。	E E IT I G SD SS ()
HTMLENCODE	大なり記号(>)などのHTMLで予約されている文字を などのHTMLエンティティ文字に置き換えて、HTMLで使用するテキスト文字列や差し込み項目値を符号化します。	I M E CT E text H o ftext に差し込み項目または予約文字を含むテキスト文字列を指定します。
ISPICKVAL	選択リスト項目の値が指定したテキストリテラルと等しいかどうかを判断します。	I IC picklist_field IS P K (text_literal 。picklist_field には選択リストの差し込み項目名を指定し、text_literal には引用符で囲んだ選択リストの値を指定します。text_literal を差し込み項目、または関数の結果にすることはできません。
JSENCODE	バックスラッシュ(\)などのエスケープ文字をアポストロフィー(')などの安全でないJavaScript 文字の前に挿入して、JavaScript で使用するテキスト文字列や差し込み項目値を符号化します。	I E C E text 。text に差し込み項目または安全でないJavaScript 文字を含むテキスト文字列を指定します。
JSINHTMLENCODE	エスケープ文字を安全でない JavaScript 文字の前に挿入し、HTML で予約されている文字を HTML エンティティ文字に置き換えて、	I } M E CT E text H o ftext に、差し込み項目または安全でないJavaScript 文字を含むテキスト文字列を指定します。

関数	説明	使用
LEFT	HTML タグ内の JavaScript で使用するテキスト文字列や差し込み項目値を符号化します。	E <code>text num_chars</code> 。E <code>text</code> に、返す項目または式を指定します。 <code>num_chars</code> に、返す文字列の先頭からの文字数を指定します。
LEN	テキスト文字列の先頭から、指定した数の文字を返します。	E <code>text</code> 。E <code>text</code> に、返される長さを持つ項目または式を指定します。
{ textAn n { e t }	指定したテキスト文字列の文字数を返します。 L は、取引先名の文字数を返します。E は、文字と空白をカウントします。 } E h T p は8を返します。	. a) L N S)
LOWER	指定したテキスト文字列内のすべての英字を小文字に変換します。英字でない文字は、この関数の影響を受けません。地域が適用されている場合は地域ルールが適用されます。	E <code>text [locale]</code> を使用し、 <code>text</code> に小文字に変換する項目またはテキストを指定して、 <code>locale</code> には任せません。意で2文字の ISO 言語コードまたは5文字の地域コードを指定します。
LPAD	テキスト文字列の左側に指定した文字を挿入します。	<code>text padded_length P (pad_string)</code> 。次の変数を指定します。 • <code>text</code> には、その左側に文字を挿入する項目または式を指定します。 • <code>padded_length</code> には、返されるテキストの合計文字数を指定します。 • <code>pad_string</code> には、挿入する必要のある文字を指定します。 <code>pad_string</code> は省略可能であり、デフォルトでは空白になります。 <code>text</code> の値が <code>pad_string</code> よりも長い場合には、 <code>text</code> が <code>padded_length</code> のサイズまで切り取られます。
MID	テキスト文字列中の途中で指定した開始位置から、指定した数の文字を返します。	MI <code>text start_num (num_chars</code> 。E <code>text</code> に、文字列を返すときに使用する項目または式を指定します。 <code>start_num</code> に、開始位置として使用する文字の、左から数えた文字数を指定します。 <code>num_chars</code> に、返す合計文字数を指定します。

関数	説明	使用
RIGHT	テキスト文字列の末尾から、指定した数の文字を返します。	I text num_chars 。 text に、返す項目または式を指定します。 num_chars に、返す文字列の最後尾からの文字数を指定します。
RPAD	テキスト文字列の右側に指定した文字を挿入します。	text padded_length P (pad_string)。次の変数を指定します。 <ul style="list-style-type: none"> text には、その後に文字を挿入する項目または式を指定します。 pad_length には、返されるテキスト文字列の合計文字数を指定します。 pad_string には、挿入する必要のある文字を指定します。pad_string は省略可能であり、デフォルトでは空白になります。 <p>text の値が pad_string よりも長い場合には、text が padded_length のサイズまで切り取られます。</p>
SUBSTITUTE	テキスト文字列中の元のテキストを新規のテキストで置き換えます。	I Etext old_text S new_text 。 text に、値の代入をする項目か値を指定します。 old_text に、置換対象となるテキストを指定します。 new_text に old_text を置換するテキストを指定します。
TEXT	数式を使用するすべての場所で、パーセント、数値、日付、日付/時間、または通貨の各項目のデータ型をテキストに変換します。また、入力規則、数式項目、および項目自動更新で、選択リスト値をテキストに変換します。	E value 。 value に、テキスト形式に変換する項目または式を指定します。この関数では、小数点(ピリオド)またはマイナス記号(ダッシュ)以外の特殊文字を使用しないでください。
TRIM	テキスト文字列の先頭と末尾から、スペースとタブを削除します。	I M text 。 text に、タブとスペースを削除する項目または式を指定します。
UPPER	指定したテキスト文字列内のすべての英字を大文字に変換します。英字でない文字は、この関数の影響を受けません。地域が適用されている場合は地域ルールが適用されます。	E text locale]。必要に応じて、 text を大文字に変換する項目または数式と置き換え、 locale は、省略可能な 2 文字の ISO 言語コードまたは 5 文字の地域コードに置き換えます。

関数	説明	使用
URLENCODE	<i>RFC 3986, Uniform Resource Identifier (URI): Generic Syntax</i> での定義に従つて、URLでは不正な空白スペースなどの文字を、これらの文字を表すコードに置き換えて、URLで使用するテキスト文字列や差し込み項目を符号化します。たとえば、空白スペースは <code>_20</code> に置き換えられ、感嘆符は <code>_21</code> に置き換えられます。	<code>URLENCODE([text])</code> [text]には差し込み項目または符号化するテキスト文字列を指定します。
VALUE	テキスト文字列を数値に変換します。	<code>VALUE([text])</code> [text]には数値に変換する項目または式を指定します。

高度な関数

関数	説明	使用
GETRECORDIDS	リストビューまたは関連リストなどのリスト内で選択したレコードのレコード ID の形式で、配列された文字列を返します。	<code>GETRECORDIDS([object_type])</code> [object_type]に、取り出したいレコードのカスタムオブジェクトまたは標準オブジェクトへの参照を指定します。
INCLUDE	Sコントロールスニペットの内容を返します。複数の Sコントロールで共通のコードを再利用する場合に、この関数を使用します。	<code>INCLUDE([source], [inputs])</code> [source]に、参照するSコントロールスニペットを指定します。[inputs]に、スニペットに渡す情報を指定します。
LINKTO	リンク (href およびアンカータグ) の形式で、カスタムSコントロールまたは Salesforce ページへの相対 URL を返します。	<code>LINKTO([label], [target], [id], [inputs], [上書きなし])</code> [label]に、リンクのテキストを指定します。[target]に URL を指定します。[id]に、レコードへの参照を指定します。入力は省略可能で、リンクには追加のパラメータを指定できます。上書きなし引数も省略可能で、デフォルトは「False」です。これは、\$Action.Account.New など標準の Salesforce ページへの対象に適用されます。別の箇所で上書きを指定したかどうかに関係なく標準の Salesforce ページを表示したい場合は、上書きなしを「True」に置き換えます。
REGEX	テキスト項目を正規表現と比較し、一致する場合に TRUE を返します。該当しない場合は FALSE を返します。正規表現とは、特定の構文規則	<code>REGEX([text], [regex_text])</code> [text]にはテキスト項目を、[regex_text]には一致させる正規表現を指定します。

関数	説明	使用
REQUIRESCRIPT	に従う文字列の形式を記述するための文字列です。	J E I E C I url Q U R r l に R 必要なスクリプトへのリンクを指定します。R S R
URLFOR	Visualforce ページにある静的リソースアーカイブ内のアクション、Sコントロール、Visualforce ページ、またはファイルの相対 URL を返します。 これは、 静的リソースアーカイブ (.zip ファイルや .jar ファイルなど)に含まれるファイルへの参照を返すために使用できます。	target id L i n p u e s R 上書きなし] 。target に URL またはアクション、Sコントロール、静的リソースマージ変数を指定します。id に、レコードへの参照を指定します。inputs に、省略可能な追加のパラメータを指定します。上書きなし引数も省略可能で、デフォルトは「False」です。これは、\$Action.Account.New など標準の Salesforce ページへの対象に適用されます。別の箇所で上書きを指定したがどうかに関係なく標準の Salesforce ページを表示したい場合は、上書きなしを「True」に置き換えます。
VLOOKUP	Excel の VLOOKUP() 関数と同様に、カスタムオブジェクトに関連する値を検索し、その値を返します。	field_to_return O O K (P field_on_lookup_object lookup_value。field_to_return には返す値が含まれている項目を、field_on_lookup_object には一致させる値が含まれている関連オブジェクト上の項目を、また lookup_value には一致させる値を指定します。

関数の符号化

演算子	説明	使途
-	2つの値の差を計算します。 $u - v$ 。それぞれの u と v に、差し込み項目、式、または他の数値を指定します。	$l - u$
*	その値を乗算します。 $e * l$ 。 u と v に、差し込み項目、式、または他の数値を指定します。	$l * u$
/	その値を除算します。 e / l 。 u と v に、差し込み項目、式、または他の数値を指定します。	l / u
^	指定した数の累乗まで数値を乗算します。 $number^integer$ 。 $number$ に、差し込み項目、式、または別の数値を指定します。 $integer$ には、整数を含む差し込み項目、式、または任意の整数を指定します。	
()	開き括弧と閉じ括弧で囲まれた式を最初に計算するように指定します。それらの $expression$ に、差し込む他の式は、標準の演算子の優先順位で計算されます。	$(expression1) expression2\dots$ 。 $expression$ に、差し込み項目、式、または他の数値を指定します。

論理演算子



メモ: 相対比較式に 値を含めることはできません。含めると例外が発生します。具体的には、次の演算子の前後どちら側にも 値を使用できません。

- (より小さい) <
- (以下) < =
- (より大きい) >
- (以上) >=

演算子	説明	使途
および	= == 2つの値が等しいかどうかを評価します。	$expression1=expression2$ または $expression1 == expression2$ 。それらの $expression$ に、差し込み項目、式、または他の数値を指定します。
および	< > != 2つの値が異なるかどうかを評価します。 $e < e$ と $e > e$ に、差し込み項目、式、または他の数値を指定します。	$expression1 <> expression2$ または $expression1 != expression2$ 。 $expression$ に、差し込み項目、式、または他の数値を指定します。
<	値がこの記号に続く値よりも小さいかどうかを評価します。 $l < u$ 。 $value1 < value2$ 。それらの $value$ に、差し込み項目、式、または他の数値を指定します。	

演算子	説明	使途
>	値がこの記号に続く値よりも大きいかどうかを評価します。	$value1 > value2$ 。それぞれの $value$ に、差し込み項目、式、または他の数値を指定します。
< =	値がこの記号に続く値以下かどうかを評価します。	$value1 \leq value2$ 。それぞれの $value$ に、差し込み項目、式、または他の数値を指定します。
>=	値がこの記号に続く値以上かどうかを評価します。	$value1 \geq value2$ 。それぞれの $value$ に、差し込み項目、式、または他の数値を指定します。
&&	2つの値、または式が両方とも true であるかどうかを評価します。この演算子は、論理関数 AND の代わりに使用します。	$logical1 \&\& logical2$ 。 $logical1$ と $logical2$ に、評価する値または式を指定します。
	複数の値または式のうち、少なくとも 1 つが true であるかどうかを評価します。この演算子は、論理関数 OR の代わりに使用します。	$logical1 \ \ logical2$ 。 $logical1$ と $logical2$ に、評価する値または式を指定します。

テキスト演算子

演算子	説明	使途
&	複数の文字列を連結します。	$string1&string2$ 。それぞれの $string$ に、差し込み項目、式、または他の数値を指定します。

付録

B

Apex および Visualforce 開発のセキュリティのヒント

セキュリティについて

Apex および Visualforce ページの強力な組み合わせにより、Force.com 開発者は、Salesforce にカスタム機能およびビジネスロジックを提供したり、Force.com プラットフォーム内部で実行するまったく新しいスタンダードアロン製品を作成することができます。ただし、プログラミング言語と同様、開発者はセキュリティ関連の不備について認識する必要があります。

Salesforce.com は、複数のセキュリティ防御を Force.com プラットフォーム自体に統合しました。ただし、不注意な開発者は多くの場合に組み込み防御を回避し、アプリケーションと顧客をセキュリティ上のリスクにさらしている場合があります。開発者が Force.com プラットフォーム上で犯す多くのコーディングエラーは、一般的な Web アプリケーションのセキュリティ脆弱性と類似しています。一部のコーディングエラーは Apex 固有のものです。

AppExchange のアプリケーションを認証するには、開発者はここで説明するセキュリティ上の弱点について学習および理解する必要があります。詳細は、<http://wiki.developerforce.com/page/Security> にある Developer Force の Force.com セキュリティリソースのページを参照してください。

クロスサイトスクリプト (XSS)

クロスサイトスクリプト (XSS) の攻撃は、悪意のある HTML またはクライアント側のスクリプトが Web アプリケーションに提供される、幅広い範囲の攻撃となります。Web アプリケーションには、Web アプリケーションのユーザに対する悪意のあるスクリプトが含まれています。ユーザは、知らぬ間に攻撃の被害者となります。攻撃者は、Web アプリケーションに対する被害者の信頼を利用し、攻撃の媒体として Web アプリケーションを使用しています。データを適切に検証することなく動的 Web ページを表示する多くのアプリケーションは攻撃されやすいといえます。Web サイトに対する攻撃は、あるユーザからの入力が別のユーザに表示されることを目的としている場合は特に単純です。可能性として、掲示板、ユーザコメントスタイルの Web サイト、ニュース、またはメールアーカイブなどがあります。

たとえば、次のスクリプトがスクリプトコンポーネント、行動、または Visualforce ページを使用する Force.com ページに使用されているとします。

```
f n erre re code m erfe & & g v > $rma up tr! ' mr par cao sp.s f a as il v > a = t e C g $rm up rt' m r pa aca s .sp & a s i/ >
```

このスクリプトブロックは、ユーザが入力した `m` `p` の値をページに挿入します。これで攻撃者は `m` `p` に次の値を入力することができます。

```
n d o u t h m o c w w o t e c l h ; : ' i p e n o . d i b g e c o o k t B / e n d a o c a u . k m 2 i o d i - r c f o a 2 k ' f 2 . i v;
```

この場合、現在のページのすべての Cookie が `c_g` `c` スクリプトに対する要求のクエリ文字列として `r` `o` `c_m` `c` に送信されます。この時点で、攻撃者は被害者のセッション Cookie を持つてあり、彼らが 被害者になりますとして Web アプリケーションに頗ソ轟女乍ナ 驚ミべき癪 す。こ d む马" 嘴 おほヨ c して03zK03L

タマイズできるようになっているためです。意図的にページに追加されるコマンドに対し XSS フィルタを指定しても意味はありません。

カスタム JavaScript

独自の JavaScript を作成した場合、Force.com プラットフォームにはユーザを保護する方法がありません。たとえば JavaScript で使用している場合、次のコードは XSS の攻撃に対して脆弱です。

```
r c p < s i >
t e o o f o r c v d n a = i a s a
t e n d o u m r c o o f .i ( ; )
t r c p < s i / >
```

<apex:includeScript>

このタグによって指定された値は、使用する JavaScript への URL です。攻撃者がパラメータに任意のデータを入力できる場合(下記の例参照)、被害者に別の Web サイトの JavaScript ファイルを使用するよう指示することができる可能性があります。

```
t el du rp c c < Visualforce コンポーネントを使用して、ページにカスタムスクリプトを追加できます。こうした場合、内容が安全で、ユーザが提供したデータが含まれていないことを慎重に確認してください。たとえば、次のスニペットはスクリプトの値としてユーザ提供の入力が含まれているため、特に脆弱です。
```

Visualforce ページのエスケープされない出力と式

属性を false に設定するコンポーネントを使用する場合、または Visualforce コンポーネント外の式を含める場合は、出力がフィルタ処理されないため、セキュリティの検証が必要です。これは、数式を使用する場合は特に重要です。

数式は関数コールとして使用したり、プラットフォームオブジェクト、ユーザの環境、システム環境、要求の環境に関する情報を含めることができます。式が生成する出力が表示されるときにエスケープされないことを認識することが重要です。数式はサーバに表示されるため、JavaScript またはその他のクライアント側の技術を使用してクライアントの表示データをエスケープすることはできません。これにより、数式が非システムデータ(恶意のあるまたは編集可能なデータ)を参照し、式自体が関数にラップされていない場合、表示中に出力をエスケープするという危険な状況を誘発する場合があります。

一般的な脆弱性は、ユーザ入力をページに表示する場合に発生します。次に例を示します。

```
n tegndprd por id &n C a u &cc a a &" " >
ro m p f < a >
x tte eoBemmm &n ot rt d r< e ou ae t lt au pd &" p I "v U &" a " > /
x et n txe { ue p&t x up &n ld& T v &" ! y & " > /
ro m p f < /a >
c te nou t tht d pu & a p Pa &" " >
e exuo o e m ldiff e v tTxe ou au ep j&t &p ei< &en &de & v l &c ! p yf E " s =a
t te nou u &p p < /a >
g p p < /a >
```

エスケープされない id によっても、クロスサイトスクリプトの脆弱性が誘発されます。たとえば、

```
t r x l r c p < s i > a (ss ' ')
```

を入力し、[更新] をクリックすると、JavaScript が実行されます。この場合、アラートダイアログが表示されますが、悪意のある使用が設定されている場合があります。

安全でないと考えられる文字列をエスケープするために使用できる関数があります。

HTMLENCODE

HTMLENCODE 関数は、大なり記号 (>) などの HTML で予約されている文字を、> などの HTML エンティティ文字に置き換え、HTML で使用するテキスト文字列や差し込み項目値を符号化します。

JSENCODE

新しいJSENCODE 関数は、バックスラッシュ (\) のようなエスケープ文字を、アポストロフィー ('') のような安全でないJavaScript 文字の前に挿入して、JavaScript で使用するテキスト文字列や差し込み項目値を符号化します。

JSINHTMLENCODE

新しいJSINHTMLENCODE 関数は、エスケープ文字を、安全でないJavaScript 文字の前に挿入し、HTML で予約されている文字を HTML エンティティ文字に置き換えて、HTML タグ内の JavaScript で使用するテキスト文字列や差し込み項目値を符号化します。

URLENCODE

新しいURLENCODE 関数は、空白などの URL では不正な文字を *RFC 3986, Uniform Resource Identifier (URI): Generic Syntax* で定義されているようにこうした文字を表すコードに置き換え、URL で使用するテキスト文字列や差し込み項目を符号化します。たとえば、感嘆符は %21 に置き換えられます。

前述の例を保護するために `URLENCODE` を使用するには、

`t txe ou u p l up e <T ex a e M Em Dv Ed e &H Tl DN o (py f> ") s =a s'a > /`

`t t ユーザが rc p < を入力し、[更新] をクリックしても、JavaScript は実行されません。代わりに文字列が符号化され、ページには m self vt rT x l rac p > i<s と表示されます。 (ss')`

`タグの代入およびデータの使用によって、エスケープされた文字およびエスケープが必要な文字が異なります。たとえば、次のような文の場合`

`tr re rc te pn erne & es d tgv > $rman! & p e! tr rPac a ap $J L R " ; si v > a =`

リンクで使用されるため、URL では HTML エスケープ文字の & 代わりに &22; を使用して二重引用符をエスケープする必要があります。そうでない場合、次のような要求

`e en lem pd mepat l r mc ha . t ooB x 22 rf 22 UB L?R = % 232 a (ss' 3%) B B`

では、次のようにになります。

`tr re rc t p xo l r< & i tv > ar = " " ; c a p(ss' ; > // s i/ >`

JavaScript が実行し、アラートが表示されます。

この場合、JavaScript が実行されないようにするために、`URLENCODE` 関数を使用します。

`r rc p < n ser > Ctag = " $rman DN Sp0t (r Pa ca a $JpL R > ;) s i/ >`

また、数式タグを使用して、プラットフォームオブジェクトデータを追加することもできます。データがユーザの組織から直接取得されますが、データをエスケープしてユーザが他のユーザ(権限レベルがより高いユーザ)のコンテキストでコードを実行できなくなります。これらの種類の攻撃は同じ組織内のユーザによって実行され、組織のユーザロールを弱体化し、データ監査の完全性を提言させてしまいます。また、多くの組織には、外部ソースからインポートされたデータがありますが、悪意のあるコンテンツの除外が行われない場合があります。

クロスサイトリクエストフォージェリ (CSRF)

クロスサイトリクエストフォージェリ (CSRF) の攻撃を受ける脆弱性は、プログラムエラーよりも保護対策の欠如です。単純な例を示して CSRF について説明します。攻撃者が `www.com` に Web ページを持っているとします。この Web ページは、サイトへの通信量を増大させる重要なサービスや情報を提供するページなどです。攻撃者のページには、次のような HTML タグがあります。

```
www.com
< i
  o cm c
  >
  c h o u b p tge e n r pce o r// tpo ur etela m y etia i ra d/n e kld ci? mak ap a a a .
  g hd h h i 1 = i 1 > /
```

つまり、攻撃者のページには、あなたの Web サイトでアクションを実行する URL が含まれています。ユーザが攻撃者の Web ページにアクセスしたときに、まだあなたの Web ページにログインしている場合、URL が取得され、アクションが実行されます。ユーザはあなたの Web ページで認証されているため、この攻撃は成功します。これは非常に単純な例で、攻撃者の手口はより巧妙になっており、コールバック要求を生成するスクリプトを使用したり、あなたの AJAX メソッドに対して CSRF 攻撃を行うこともあります。

詳細および従来の防御方法については、次の記事を参照してください。

- http://www.owasp.org/index.php/Cross-Site_Request_Forgery
- <http://www.cgisecurity.com/articles/csrf-faq.shtml>
- <http://shiflett.org/articles/cross-site-request-forgeries>

Force.com プラットフォーム内では、この攻撃を回避する対 CSRF トークンが実装されています。すべてのページにランダムな文字列が非表示項目として含まれています。次のページが読み込まれると、アプリケーションはこの文字列の正当性を確認し、値が予測値と一致しない限り、コマンドを実行しません。この機能によって、すべての標準コントローラおよびメソッドの使用時に攻撃から保護されます。

開発者は、リスクを意識せずに組み込み防御策をスキップしてしまう場合があります。たとえば、オブジェクト ID を入力パラメータとして SOQL コールで使用するカスタムコントローラがあるとします。次のコードスニペットについて考えます。

```
t en gorpld p d { l n t } aex " y ss" g p a b il i <" /a >
bl { l p d m C i ss a y ss a
bt n d pc i v i i ( )
e ek dA e It e negrntpe e t gie st rm Ba . gd Pa ( . ) Pa a s ( . ) i ( ; )
c An e e ou tbo coej l d t An ou= Ms c[ di E R O W ] H R i = i ;
d l bo j ;
n r ru ;
```

この場合、開発者は独自の action メソッドを作成して、意識せずに CSRF 対策コントロールをスキップしています。パラメータはコードで読み込まれ、使用されます。CSRF 対策トークンが読み込まれたり、検証されたりすることはありません。攻撃者の Web ページでは、CSRF 攻撃を使用してユーザをこのページに移動させ、パラメータとして攻撃者が望む値を指定する可能性があります。

このような状況に対する組み込み防御策がないため、開発者は前例の変数のようなユーザ指定のパラメータに基づいてアクションを実行するページの書き込みに対し、注意する必要があります。回避策の1つは、アクションを実行する前に中間の確認ページを挿入して、ユーザが本当にそのページをコールしようとしているのかどうかを確認することです。その他の対策として、組織のアイドルセッションのタイムアウトを短くすること、ユーザがあるサイトで認証されたままブラウザを使用して別のサイトに移動しないように、アクティブなセッションからログアウトすることを推奨すること、などが考えられます。

SOQL インジェクション

他のプログラミング言語では、上記の弱点を SQL インジェクションといいます。Apex では SQL を使用しませんが、独自のデータベースクエリ言語 SOQL を使用します。SOQL は、SQL より単純で、機能が制限されています。そのため、SOQL インジェクションのリスクは SQL と比較して大幅に低くなりますが、攻撃は従来の SQL インジェクションとほぼ同じです。集計時は、SQL/SOQL インジェクションではユーザが提供した入力を取得し、これらの値を動的 SOQL クエリに使用します。入力が検証されない場合、SOQL ステートメントを事实上変更する SOQL コマンドを指定し、アプリケーションにトリックを仕掛けて意図しないコマンドを実行するようにします。

SQL インジェクション攻撃の詳細は、以下を参照してください。

- http://www.owasp.org/index.php/SQL_injection
- http://www.owasp.org/index.php/Blind_SQL_Injection
- http://www.owasp.org/index.php/Guide_to_SQL_Injection
- <http://www.google.com/search?q=mysql+injection>

Apex での SOQL インジェクションの脆弱性

以下に SOQL に対して脆弱な Apex コードおよび Visualforce の単純な例を示します。

```

x t e n      g o r p l d e p a   < o r l d a c a   Q = L   O S      "   >
x       r o m p f   <   a   >
x t e t t x e o u e u t p e l u p   & T   E a m   v   a"!   N   " a   > /
x e t n t x e { u e p h u p}   T   m   á   v   a"!   " a   > /
x t t   e Bndrem p o l tu{ < n e n a o } a c n w q   Q   a"!   " " y   i a =! !   y   > /
x       r o m p f   <   / a   >
x       g   p   p   <   / a   >

b l u   t e n p c c o r l d r C   i   Q s L a   O S
t   e b l n n r   p c g   m   i   i s   a
c   t   e   n g n   r u }   m   ; a
c   e   n   e   n l   u   s   ; a
}

e   e   e   b l e   e   n { g r c   n f q   c   i   R a   y( )
n   t   r   n r g   q   g   t   d t n   E l S C   d   T y i S   C = L   c   S   E   F R O W   H   a   R   '
e   e   e   t   e   l   d e In le   d   f \ e l   m   a'   = \   m a k   N   a   +   %   '   a   %   '   ;   '
e   e   t   n t e   l a u   b   t   n   q r   m   R y s   D g =   a s a   a   y(   y i s   ;   )
c   n   n   r u   l h   ;   ;


```

これは単純な例ですが、ロジックについて説明しています。コードは、削除されていない取引先責任者の検索を行うためのものです。ユーザは という入力値を指定します。値はユーザが指定する任意の値で、検証されません。SOQL クエリは動的に構築され、`n q` メソッドで実行されます。ユーザが正当な値を指定すると、ステートメントは次のように期待どおり実行されます。

```

e   u l   d n l   u   p p   B U   m   b o / s   s   i   v   a   a
m   r   g   Q   //   g   i
t   d t n   E   E C   e l e e r   M   C   L e   S   E   d E E R   Q e W   d H   a f   R   D   B   $   b o   =   s   a k   N   a   %   '   %

```

ただし、次のようにユーザが予期しない値を入力したかのようになります。

ul qd dntu treq U e mst s : i v ma la E s % (S) NI a K

この場合、クエリ文字列は次のようにになります。

t d tn E EC e E f M C L e ls E e H R Q A W e H t a f t R D s e I H s D a N N L a K E S g N

結果には削除されていない取引先責任者だけでなく、すべての取引先責任者が表示されます。SOQL インジェクションにより、脆弱なクエリの対象となるロジックを変更することができます。

SQL インジェクションの防御策

SOQL インジェクションの攻撃を回避するには、動的 SOQL クエリを使用しないようにします。代わりに、静的クエリとバインド変数を使用します。上記の脆弱な例は、静的 SOQL を使用して次のように書き直すことができます。

bl̥ te ꝑ pc c or lð r C i QsLa OS
 t ðl̥n n r p cg m i is a
 e t e ngn ru} m ; a
 e n e n l u s æ a
 }
 e e e e bie e n {grc n fq c i Ra y()
 e n e r g q nm mi S Ny æ + % ' a , ;'
 e t n q lu t dtñ E EC d RTysM C=L c [S E E FR O W H a R
 e e e t e l de In le q fel ñ \$ n q = : ms ak ñ a Ny a;
 } n ru lh ;

動的 SOQL を使用する必要がある場合、`l_cq_w_o_p` メソッドを使用して、ユーザ指定の入力を削除します。このメソッドは、ユーザから渡される文字列のすべての単一引用符にエスケープ文字(\)を追加します。このメソッドにより、すべての単一引用符を、データベースコマンドではなく、囲まれた文字列として処理します。

データアクセスコントロール

Force.com プラットフォームは、データ共有ルールを広範囲に使用します。各オブジェクトには権限があり、ユーザーが読み取り、作成、編集、削除できる共有設定がある場合があります。これらの設定は、すべての標準コントローラを使用する場合に強制されます。

Apex クラスを使用する場合、組み込みユーザ権限、および項目レベルのセキュリティ制限は実行時に重視されません。デフォルトの動作として、Apex クラスに組織内のすべてのデータを読み込み更新する機能があります。これらのルールは強制されないため、Apex を使用する開発者は、ユーザ権限、項目レベルのセキュリティ、または組織のデフォルト設定によって通常は非表示となる機密データが不注意で公開されないようにする必要があります。これは特に、Visualforce ページで当てはまります。たとえば、次の Apex 構似コードについて考えます。

```
    blit 1 te ipa co oncl d r C i ss a s
    e blu d r pdc i v i a )
t tn t ta c c ct dt n E ECeo a T MeC@L c [S fil El FR O W : H ] a R N av a;
}
```

この場合、現在ログインしているユーザにこれらのレコードを表示する権限がない場合でも、すべての取引先責任者レコードが検索されます。解決策として、クラスを宣言する場合、次のように修飾キーワードの `h
r
s` を使用します。

```
bu n     prc t   h1 phe n u co iomcià r sc i a     ss a s  
n     r     h   h     . . .
```

キー`h`を使用すると、プラットフォームはすべてのレコードに完全アクセス権限を付与するのではなく、現在ログインしているユーザのセキュリティ共有権限を使用します。

付録

C

Visualforce コントローラで使用する Apex クラス

この付録では、カスタム Visualforce コントローラおよびコントローラ拡張を構築する場合に使用できる、システムが提供する Apex クラスについて説明します。

カスタムコントローラおよび拡張機能についての詳細は、「[カスタムコントローラおよびコントローラ拡張](#)」(ページ 76)を参照してください。

Apex についての詳細は、『[Force.com Apex コード開発者ガイド](#)』を参照してください。

[ApexPages クラス](#)

現在のページの参照、および現在のページに関連付けられたメッセージの追加や確認をするために、[ApexPages](#) を使用します。

[Action クラス](#)

An [ApexPages](#) を使用して、Visualforce カスタムコントローラまたはコントローラ拡張で使用できる action メソッドを作成できます。

[Cookie クラス](#)

Apex クラスにより、Apex を使用して Force.com サイトの Cookie にアクセスできます。

[IdeaStandardController クラス](#)

オブジェクトは、[Idea](#) で提供される機能のほか、アイデア固有の機能を提供します。

[IdeaStandardSetController クラス](#)

オブジェクトは、[KnowledgeArticle](#) で提供される機能のほか、アイデア固有の機能を提供します。

[KnowledgeArticleVersionStandardController クラス](#)

オブジェクトは、[KnowledgeArticleVersion](#) で提供される機能のほか、記事固有の機能を提供します。

[Message クラス](#)

標準コントローラ使用時にエンドユーザがページを保存すると発生する入力規則エラーが含まれます。

PageReference クラス

PageReference は、ページのインスタンス化への参照です。多数の属性の 1 つである PageReferences は URL、一連のクエリパラメータ名および値で構成されます。

SelectOption クラス

オブジェクトは Visualforce レコード、リスト、またはソード構成要素に指定可能な値のいずれかを指定します。

StandardController クラス

標準コントローラの拡張を定義する場合は、StandardController を使用します。

StandardSetController クラス

オブジェクトを使用して、Salesforce が提供する、プリビルドされた Visualforce リストコントローラと同様のリストコントローラ、またはその拡張としてリストコントローラを作成することができます。

ApexPages クラス

現在のページの参照、および現在のページに関連付けられたメッセージの追加や確認をするために、ApexPages を使用します。

名前空間

System

使用方法

ApexPages は PageReference クラスおよび Message クラスの名前空間としても使用されます。

ApexPages インスタンスマетод

ApexPages インスタンスマетод

ApexPages のインスタンスマетодを次に示します。

addMessage(sObject)

現在のページのコンテキストにメッセージを追加します。

addMessages(Exception)

発生した例外に基づいて、現在のページのコンテキストにメッセージのリストを追加します。

getMessages()

現在のコンテキストに関連付けられたメッセージのリストを返します。

hasMessages()

現在のコンテキストに関連付けられたメッセージが存在する場合は true、存在しない場合は false を返します。

hasMessages(ApexPages.Severity)

指定された重要度のメッセージが存在する場合は `true`、存在しない場合は `false` を返します。

s a

addMessage(sObject)

現在のページのコンテキストにメッセージを追加します。

署名

```
public void addMessage(sObject obj, String message, ApexPages.Severity severity)
```

```
Ba . ss a ()
```

パラメータ

`msg`

型: [ApexPages.Message](#)

戻り値

型: Void

addMessages(Exception)

発生した例外に基づいて、現在のページのコンテキストにメッセージのリストを追加します。

署名

```
public void addMessages(List<Exception> exceptions)
```

パラメータ

`ex`

型: Exception

戻り値

型: Void

getMessages()

現在のコンテキストに関連付けられたメッセージのリストを返します。

署名

```
public List<Message> getMessages()
```

パラメータ

なし

戻り値

型: [ApexPages.Message\[\]](#)

hasMessages()

現在のコンテキストに関連付けられたメッセージが存在する場合は `f` 、存在しない場合は `f` を返します。

署名

```
blB n eðo pc h g M i as ss sa ( )
```

パラメータ

なし

戻り値

型: Boolean

hasMessages(ApexPages.Severity)

指定された重要度のメッセージが存在する場合は `f` 、存在しない場合は `f` を返します。

s a

署名

```
blB n eðo ep c ex eAðh g M te i g rp as ss sa ( Bav. Si )
```

パラメータ

`sev`

型: `ApexPages.Severity`

戻り値

型: Boolean

Action クラス

Action クラスを使用して、Visualforce カスタムコントローラまたはコントローラ拡張で使用できる action メソッドを作成できます。

名前空間

ApexPages

使用方法

たとえば、カスタム保存を実行するコントローラ拡張に

メソッドを作成できます。

インスタンス化

次のコードのスニペットは、`savee`アクションを使用する新しい `ApexPages` オブジェクトをインスタンス化する方法について説明しています。

```
t e An g p An eon eak A ct { BaAni g v p iai c = Ba . i ( ' a ; ' )
```

例

次の例では、ユーザが新しい取引先を更新または作成し、[保存]ボタンをクリックした場合、更新された取引先または作成された取引先に加えてメッセージがシステムデバッグログに書き込まれます。この例では、取引先の標準コントローラを拡張します。

コントローラ拡張は、次のとおりです。

```
e bh{ l n p c c g pC i ss a a
e e e bie e ne grte t { d rf c o i c Ra T p R i
} e ne rt ru g 2 p Pa S;
} /...
/...
```

上記のコントローラ拡張を使用するページの Visualforce マークアップは、次のとおりです。

```
t n n o om p < a >
tt e t e n bt u p ei mn et & coa x ia tT Akja pov c" y" Ba . i " ... > /
x tit eBndemn p o l utern ro a or ldma(C v a t a"co a o )o co i "I Ta =" !kia v
t n n o om p < /a >
te n e gorpld pa < g a pC a =" a " >
t an { me oCeo qte t< Id rT y o kia v T p R i s " > /
g p p < /a >
```

デバッグログについての詳細は、Salesforce オンラインヘルプの「デバッグログの表示」を参照してください。

Action インスタンスメソッド

- c のインスタンスメソッドを次に示します。

`getExpression()`

アクションが呼び出されたときに評価される式を返します。

`invoke()`

アクションを呼び出します。

`getExpression()`

アクションが呼び出されたときに評価される式を返します。

署名

```
e bh{ e xc p c g gr o E pi i S ss ( )
```

パラメータ

なし

戻り値

型: String

invoke()

アクションを呼び出します。

署名

```
public String invoke()
```

パラメータ

なし

戻り値

型: System.PageReference

Cookie クラス

このクラスにより、Apex を使用して Force.com サイトの Cookie にアクセスできます。

名前空間

System

使用方法

PageReference クラスのメソッドを使用して、ページに Cookie を添付します。

重要:

- Apex の Cookie 名と値セットは URL 符号化されています。つまり、@などの文字は % 記号および 16進数表現に置き換えられます。
- メソッドは Cookie 名にプレフィックス「p」を追加します。
- Cookie の値を に設定すると、期限切れの属性の設定ではなく、空の文字列値の Cookie を送信します。
- Cookie の作成後は、Cookie のプロパティを変更することはできません。
- 機密情報を Cookie に格納する場合は注意してください。Cookie の値に関係なくページはキャッシュされます。動的なコンテンツを生成するために Cookie の値を使用する場合は、ページキャッシュを無効にする必要があります。詳細は、Salesforce オンラインヘルプの「Force.com サイトページのキャッシュ」を参照してください。

クラスを使用する場合は、次の制限に留意してください。

- クラスには、Salesforce.com API バージョン 19 以降を使用して保存されている Apex を使用することでのみアクセスできます。
- Force.com ドメインごとに設定できる Cookie の最大数はブラウザにより異なります。新しいブラウザは古いブラウザより高い制限が設定されています。
- Cookie は名前および属性を含め 4K 未満である必要があります。

サイトの詳細は、Salesforce オンラインヘルプの「Force.com サイトの概要」を参照してください。

例

次の例では、Cookie クラスを作成します。このクラスは Visualforce ページ（下記マークアップを参照）を使用して、ユーザにページが表示されるたびにカウンタが更新されます。ページへのアクセス回数が Cookie に保存されます。

```

public class Counter {
    public Integer count {get; set;}
}

<apex:page>
<script>
    var cookie = document.cookie;
    var count = cookie.match(/(^|;)count=([0-9]+)(;|$)/);
    if(count) {
        count = parseInt(count[2]);
    } else {
        count = 0;
    }
    count++;
    document.cookie = "count=" + count;
    document.getElementById("counter").innerHTML = count;
</script>
<div id="counter"></div>

```

次は、上記の Controller クラスを使用する Visualforce ページです。アクションでは、上記のコントローラでメソッドをコールします。

```

<apex:page controller="CounterController">
<apex:form>
    <apex:commandbutton value="Count Me!" action="increment" />
    <apex:outputtext value="You have been counted " />
    <apex:outputtext value="times." />
</apex:form>

```

Cookie インスタンスマソッド

Cookie インスタンスマソッド

○○ C のインスタンスマソッドを次に示します。

`getDomain()`

要求を行うサーバの名前を返します。

`getMaxAge()`

Cookie の有効期間を示す秒単位の数字が返されます。 を設定すると、セッション Cookie が発行されます。
を設定すると、Cookie は削除されます。

`getName()`

Cookie の名前を返します。 にはできません。

`getPath()`

Cookie の取得元のパスを返します。 または空白にすると、場所はルートまたは「/」に設定されます。

`getValue()`

セッション ID など、Cookie で取得されるデータを返します。

`isSecure()`

Cookie が HTTPS でのみアクセス可能な場合、 を返します。それ以外の場合は、 f を返します。

s a

`getDomain()`

要求を行うサーバの名前を返します。

署名

```
public String getDomain();
```

パラメータ

なし

戻り値

型: String

`getMaxAge()`

Cookie の有効期間を示す秒単位の数字が返されます。 を設定すると、セッション Cookie が発行されます。
を設定すると、Cookie は削除されます。

署名

```
public Integer getMaxAge();
```

パラメータ

なし

戻り値

型: Integer

getName()

Cookie の名前を返します。 にはできません。

署名**パラメータ**

なし

戻り値

型: String

getPath()

Cookie の取得元のパスを返します。 または空白にすると、場所はルートまたは「/」に設定されます。

署名**パラメータ**

なし

戻り値

型: String

getValue()

セッション ID など、Cookie で取得されるデータを返します。

署名**パラメータ**

なし

戻り値

型: String

isSecure()

Cookie が HTTPS でのみアクセス可能な場合、**true** を返します。それ以外の場合は、**false** を返します。

署名

```
public boolean isSecure()
```

パラメータ

なし

戻り値

型: Boolean

IdeaStandardController クラス

IdeaStandardController オブジェクトは、IdeaStandardController の特定のインスタンスでコールされ、実行されます。

名前空間

ApexPages

使用方法

IdeaStandardController オブジェクトのメソッドは、IdeaStandardController の特定のインスタンスでコールされ、実行されます。

 メソッドは、現在限定リースプログラムでのみ使用できます。組織でのこれらのクラスの有効化についての詳細は、株式会社セールスフォース・ドットコムの担当者までお問い合わせください。

このクラスに記載されたメソッドのほか、クラスは、クラスに関連付けられたすべてのメソッドを継承します。

インスタンス化

IdeaStandardController オブジェクトはインスタンス化できません。アイデアの標準コントローラを使用する場合は、カスタム拡張コントローラのコンストラクタを介してインスタンスを取得できます。

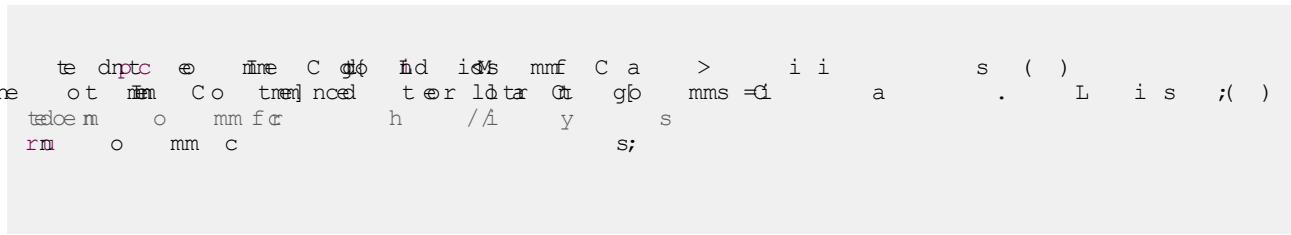
例

次の例では、IdeaStandardController オブジェクトをカスタムリストコントローラのコンストラクタで使用する方法を示します。この例では、コメントリストデータを Visualforce ページに表示する前に操作するためのフレームワークを示します。

```
public class IdeaStandardController {
    private List comments;
    private String commentText;

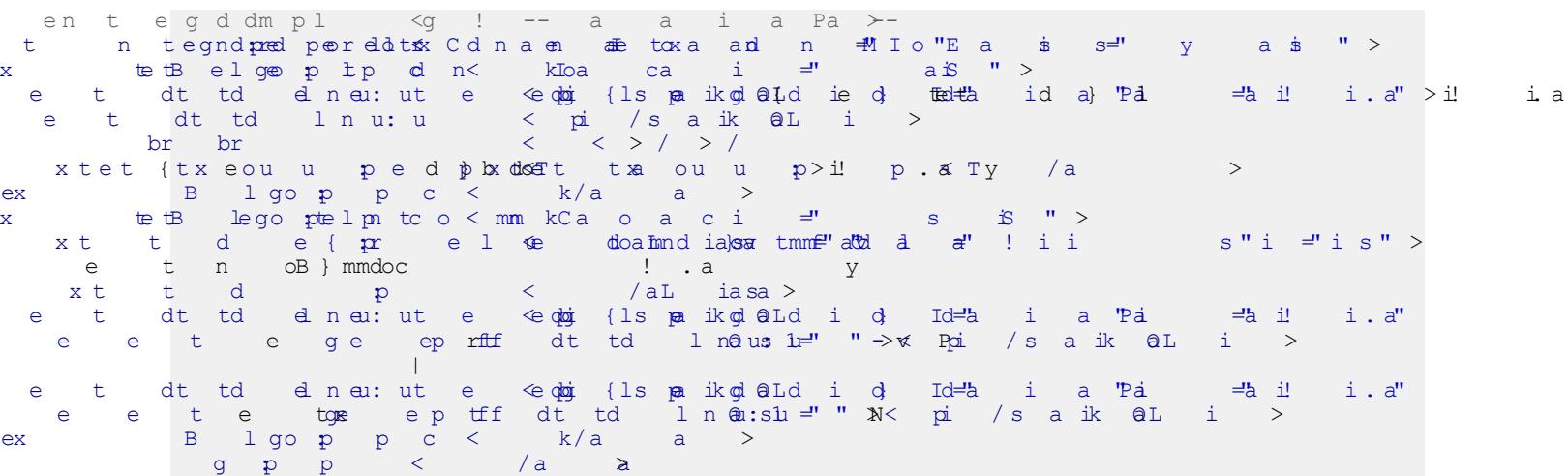
    public IdeaStandardController() {
        comments = new List();
        commentText = '';
    }

    public void addComment() {
        Comment comment = new Comment();
        comment.text = commentText;
        comments.add(comment);
        commentText = '';
    }
}
```



次のVisualforce マークアップは、上記の IdeaStandardController の例をページ内で使用する方法を示します。この例が機能するためには、ページ名を `IdeaStandardController` にする必要があります。

 メモ: Visualforce ページにアイデアとコメントを表示するには、次の例でコメントを表示する特定のアイデアの ID(例: 1) を指定する必要があります。



IdeaStandardController インスタンスマソッド

IdeaStandardController インスタンスマソッド

のインスタンスマソッドを次に示します。

`getCommentList()`

現在のページの参照のみコメントのリストを返します。

`getCommentList()`

現在のページの参照のみコメントのリストを返します。

署名

パラメータ

なし

戻り値

型: IdeaComment[]

このメソッドは、次のコメントプロパティを返します。

```

d   .   i
n   oB  mndoc      y
t   r   d . c       D   a   a
r   B d . d       I       a   y
r   B t d n e o   mmm c       m c   a   yk   i   Ny   a

```

IdeaStandardSetController クラス

ndt t ed md or lā r C t e n t オブジェクトは、 or lā r C
固有の機能を提供します。

で提供される機能のほか、アイデア

名前空間

ApexPages

使用方法

 メソッド or lā r C t クラスおよび or lā r C クラスは、現在限定リースプログラムでのみ使用できます。組織でのこれらのクラスの有効化についての詳細は、株式会社セールスフォース・ドットコムの担当者までお問い合わせください。

 上記のメソッドのほか、md or lā r C n t t クラスは a sr lā r C に関連付けられたメソッドを継承します。

 メソッド: or lā r C から継承したメソッドを使用して、 I メソッドによって返 sa されたアイデアのリストを変更することはできません。

インスタンス化

IdeaStandardSetController オブジェクトはインスタンス化できません。アイデアの標準リストコントローラを使用する場合は、カスタム拡張コントローラのコンストラクタを介してインスタンスを取得できます。

例: プロファイルページの表示

次の例では、 IdeaStandardSetController オブジェクトのカスタムリストコントローラのコンストラクタでの使用方法を示します。

```

    le ep dix & nM In if E ss a y aP s
    er n ex al p t fe ngt pt ed mde ae ortlā t eGnBa . or a lā r aC S i as ;
    ble e dix p cM In e x & f E ti e ngt pt ed md s stor h d r C Bar .ld r caS a a S
    t t edn e or d k r aC t e ingt pt ed md teer h d r C Bar .ld r caS a a S ) ;
} te ble t ed p e { glo d d M L fi < s > a i i s a )
e de e dt I t ejn e et or lā t C dg[ s = a I aS . L i sa ;( )
e e edo m d r f h / i iy s a
} e n r ru d i s; a
}

```

次の Visualforce マークアップは、上記の IdeaStandardSetController の例とコンポーネントによって、最新の回答、登録されたアイデア、ユーザに関する投票の一覧を表示するプロファイルページがどのように表示されるかを示します。この例では特定のユーザ ID を識別しないため、ページには現在ログインしているユーザのプロファイルページが自動的に表示されます。この例が機能するためには、ページ名を 'profile' にする必要があります。

```

<apex:page>
    <apex:form>
        <apex:sectionHeader title="アイデアとコメント" />
        <apex:repeat value="{!ideas}" var="idea">
            <apex:panelGrid columns="2" style="width: 100%; border-collapse: collapse; margin-bottom: 10px;" />
            <apex:outputText value="アイデア:<br/>{!idea.title}" />
            <apex:outputText value="説明:<br/>{!idea.description}" />
            <apex:outputText value="状況:<br/>{!idea.status}" />
            <apex:outputText value="作成者:<br/>{!idea.creator}" />
            <apex:outputText value="最終更新:<br/>{!idea.lastUpdated}" />
            <apex:outputText value="評議:<br/>{!idea.votes}" />
            <apex:outputText value="評議:<br/>{!idea.comments}" />
        </apex:repeat>
    </apex:form>
</apex:page>

```

前の例では、特定のアイデアの詳細ページを表示する次の Visualforce マークアップにリンクします。この例が機能するためには、ページ名を 'profile' にする必要があります。

```

<apex:page>
    <apex:form>
        <apex:repeat value="{!ideas}" var="idea">
            <apex:panelGrid columns="2" style="width: 100%; border-collapse: collapse; margin-bottom: 10px;" />
            <apex:outputText value="アイデア:<br/>{!idea.title}" />
            <apex:outputText value="説明:<br/>{!idea.description}" />
            <apex:outputText value="状況:<br/>{!idea.status}" />
            <apex:outputText value="作成者:<br/>{!idea.creator}" />
            <apex:outputText value="最終更新:<br/>{!idea.lastUpdated}" />
            <apex:outputText value="評議:<br/>{!idea.votes}" />
            <apex:outputText value="評議:<br/>{!idea.comments}" />
            <apex:outputText value="詳細:<br/><a href='/idealist/{!idea.id}'>詳細></a>" />
        </apex:repeat>
    </apex:form>
</apex:page>

```

例: 上位のアイデアとコメント、最近のアイデアとコメント、最も人気のあるアイデアとコメントのリストを表示

次の例では、IdeaStandardSetController オブジェクトのカスタムリストコントローラのコンストラクタでの使用方法を示します。



メモ: この例でアイデアが返されるためには、少なくとも 1 つのアイデアを作成する必要があります。

```

public class IdeaStandardSetController {
    public IdeaStandardSetController() {
        // コンストラクタ
    }
}

```

```

    te ble t ed p e { glo d d M L fi<s > a i i s a( )
    e de dt I te jne et or l d t C dia[ s =á I as . L i sa ;( )
    e e do m d r f h /i iy s a
    e n r ru d i s; a
}

```

次の Visualforce マークアップは、上記の IdeaStandardSetController 例をコンポーネントと共に使用して、最近、上位、最も人気あるアイデアとコメントをどのように表示するかを示します。この例が機能するためには、ページ名を `g` にする必要があります。

```

    n te g d lm p g< ! -- a ai s Pa >-
    n tegndpxl por ddtx Cd n a a a deatad n #I " oa E $ s=" Ly i sa $ "
    re ad t r c d r v S ="a as " a
    B lgo p p c < ka a >
    t tdt l ntue u t eo pi te siaksOLte in p s =" d "c Iai s "PaR
    t tdt l nu u <pi /siaksOL i >
    t tdt l ntutu e ro pi siaksOL pi g osd =" " I#I =ai s "Pa>
    t tdt l nu u <pi /siaksOL i >
    t tdt l ntu u mo pi alu es iaks oip ei p sg o#ur d " ap =ai s "Pa> P a s a
    t tdt l nu u <pi /siaksOL i >
    t tdt l ntu en t roen pi a siaksOLti en ps g =n o stomm Gai s "PaR
    t tdt l nu u <pi /siaksOL i >
    B lgo p p c < ka a >
    B lgo p p c < ka a >
    xt t ed epleu do<m ded aL t fia d d a! ! i i s "va =ia a" a>a
    e t dt td el no: he d cdpi des daeik dae w i ig =ap ibg aia a" =ai
    e t test d ed e lt dt td l mou: ia a a < pi /sa ik a i >
    ex t d p < /aL iasa >
    B lgo p p c < ka a >
    g p p < /a a

```

前の例では、`l nou: lu < pi` コンポーネントは、特定のアイデアの詳細ページを表示する次の Visualforce マークアップにリンクします。このページの名前は `g` にする必要があります。

```

    n ew g d lm p g< ! -- a vai Pa >-
    n tegndpxl por l d s Cd a a a a = " >a
    te B el go p ip d n < kloa ca i = " a as " >
    e t dt td el n ue u e < pi{ se pa dik {OL eid test vaid } "Pa =a il i . a" >il
    e t dt td l n u: u < pi /sa ik OL i >
    br br < < > / > /
    xt et { txe ou u pe d }px doct txa ou u p>il p . a Ty /a >
    B lgo p p c < ka a >
    g p p < /a a

```

IdeaStandardSetController インスタンスマソッド

IdeaStandardSetController インスタンスマソッド

のインスタンスマソッドを次に示します。

`getIdeaList()`

現在のページセットの参照のみアイデアのリストを返します。

getIdeaList()

現在のページセットの参照のみアイデアのリストを返します。

署名

e **b**lt d t **p c dg**] I i a[L i sa ()

パラメータ

なし

戻り値

型: Idea[]

使用方法

ポートを使用して、アイデアリストや詳細ページのほか、プロファイルページを表示できます(下記の例を参照)。次に、このメソッドで返されるプロパティのリストを示します。

KnowledgeArticleVersionStandardController クラス

ほか、記事固有の機能を提供します。

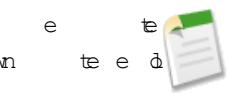
で提供される機能の

名前空間

ApexPages

使用方法

e 上記のメソッドのほか、`rgln` `rn` `toe` `nd` `rd` `Kr` `ld` `Vr` `ic` `n` `t` クラスは `ar` `da` `r` `C` 関連付けられたすべてのメソッドを継承します。



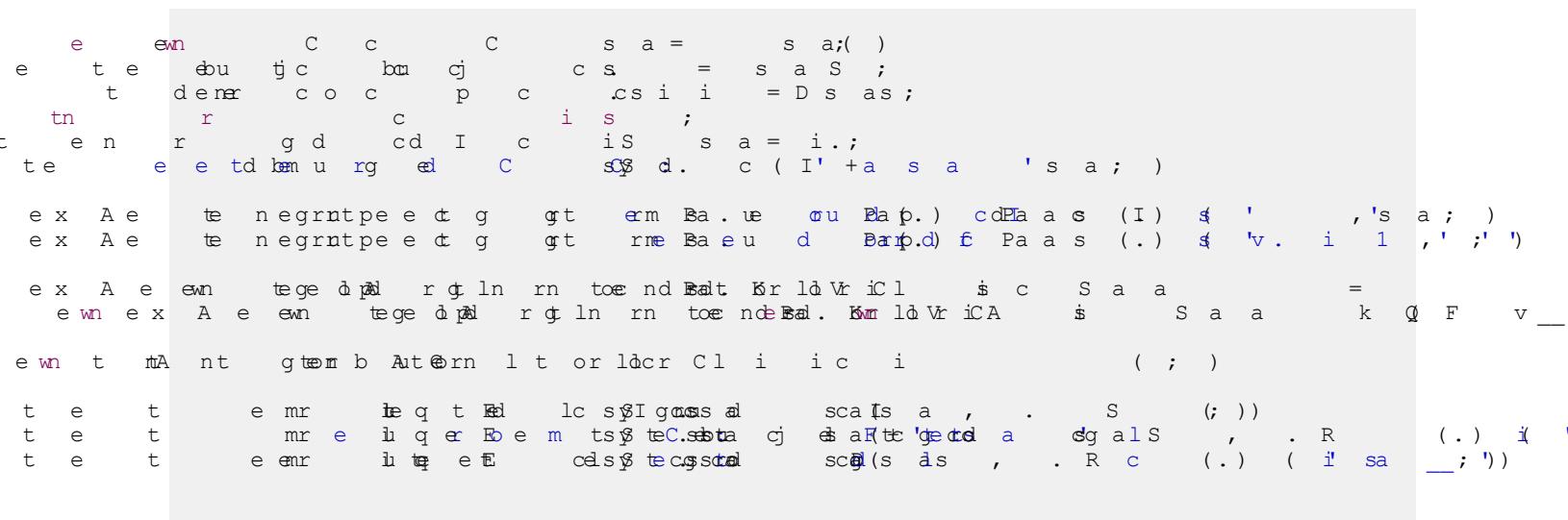
**tメモ: ただし、 、 および i メソッドは、継承されても
rg ln rn toe nd rd Kr ld Wr ic \$ クラスには使用できません。**

例

次の例では、`r g ln rn tœ nd rd Br lð Vr ic` オブジェクトを使用してカスタム拡張コントローラを作成する方法を示します。この例では、カスタマーサポートエージェントが、ケースをクローズするときに作成するドラフト記事で自動入力された項目を表示できるようにするというクラスを作成します。

前提条件:

- 「」という記事タイプを作成します。手順は、Salesforce オンラインヘルプの「記事タイプの定義」を参照してください。
 - 【詳細】というテキストカスタム項目を作成します。手順は、Salesforce オンラインヘルプの「カスタム項目の記事タイプへの追加」を参照してください。
 - 「場所」というカテゴリグループを作成して、「」というカテゴリに割り当てます。手順は、Salesforce オンラインヘルプの「カテゴリグループの作成と編集」および「カテゴリグループへのデータカテゴリの追加」を参照してください。
 - 「トピック」というカテゴリグループを作成して、「メンテナンス」というカテゴリに割り当てます。



前の例で説明した目的で(ケースで登録された記事の変更)カスタム拡張コントローラを作成した場合、クラスを作成した後に次の手順を実行します。

1. Salesforce 組織にログインし、[設定] で [カスタマイズ] > [ナレッジ] > [設定] をクリックします。
2. [編集] をクリックします。
3. [カスタマイズを使用] 項目にクラスを割り当てます。この操作により、新しいクラスに指定された記事タイプは、クローズケースに割り当てられた記事タイプに関連付けられます。
4. [保存] をクリックします。

[KnowledgeArticleVersionStandardController インスタンスメソッド](#)

KnowledgeArticleVersionStandardController インスタンスメソッド

別のオブジェクトから新しい記事を作成するときに、ソースオブジェクトトレコードの ID を返します。

`getSourceld()`

別のオブジェクトから新しい記事を作成するときに、ソースオブジェクトトレコードの ID を返します。

`setDataCategory(String, String)`

新しい記事を作成するときに、指定したデータカテゴリグループのデフォルトのデータカテゴリを指定します。

`getSourceld()`

別のオブジェクトから新しい記事を作成するときに、ソースオブジェクトトレコードの ID を返します。

署名

パラメータ

なし

戻り値

型: String

setDataCategory(String, String)

新しい記事を作成するときに、指定したデータカテゴリグループのデフォルトのデータカテゴリを指定します。

署名

```
blu t e td      p c t      e   mo g  n    i    g t rDsgecdu ar a a     ygipSro g c a     yG      , i S      a     y )
```

パラメータ

categoryGroup

型: String

category

型: String

戻り値

型: Void

Message クラス

標準コントローラ使用時にエンドユーザがページを保存すると発生する入力規則エラーが含まれます。

名前空間

ApexPages

使用方法

標準コントローラを使用している場合、エンドユーザがページを保存したときに発生するすべての入力規則エラー（標準およびカスタム）が自動的にページのエラーコレクションに追加されます。コンポーネントがバインドされた項目にエラーが発生すると、そのコンポーネントのエラーコレクションにメッセージが追加されます。そのページのエラーコレクションにすべてのメッセージが追加されます。詳細は、『Visualforce開発者ガイド』の「[入力規則と標準コントローラ](#)」を参照してください。

アプリケーションでカスタムコントローラや拡張を使用する場合は、エラーを収集するためのクラスを使用する必要があります。

インスタンス化

カスタムコントローラまたはコントローラ拡張では、次のいずれかの方法でメッセージをインスタンス化できます。

• `getSeverity()` `severity` enum の値を返します。

ここで、`severity` はメッセージの重要度を指定する enum で、`summary` はメッセージを要約するために使用する String です。次に例を示します。

`String severity = Message.getSeverity();`

• `getSummary()` `summary` String を返します。

ここで、`severity` はメッセージの重要度を指定する enum、`summary` はメッセージを要約するために使用する String、`detail` はエラーに関する詳細情報を示す String です。

ApexPages.Severity Enum

enum 値を使用してメッセージの重要度を指定します。有効な値は次のとおりです。

- Critical

- Error

- Information

- Notice

- Warning

すべての enum は、`toString()` などの標準メソッドにアクセスできます。

Message インスタンスマソッド

このインスタンスマソッドを次に示します。

`getComponentLabel()`

関連するコンポーネントのラベルを返します。表示ラベルが定義されていない場合、メソッドはを返します。

`getDetail()`

メッセージの作成に使用する詳細パラメータの値を返します。詳細 string が指定されていない場合、このメソッドはを返します。

`getSeverity()`

メッセージの作成に使用する重要度の enum を返します。

`getSummary()`

メッセージの作成に使用する要約の String を返します。

`getComponentLabel()`

関連するコンポーネントのラベルを返します。表示ラベルが定義されていない場合、メソッドはを返します。

署名

```
e blnt er tpcg go om l ip i S L a ( )
```

パラメータ

なし

戻り値

型: String

getDetail()

メッセージの作成に使用する詳細パラメータの値を返します。詳細 string が指定されていない場合、このメソッドは null を返します。

署名

```
e knt tr p cg g l i i S D i a ( )
```

パラメータ

なし

戻り値

型: String

getSeverity()

メッセージの作成に使用する重要度の enum を返します。

署名

```
x kAe te p deg r pt g ri Bav. Si y v Si y( )
```

パラメータ

なし

戻り値

型: ApexPages.Severity

getSummary()

メッセージの作成に使用する要約の String を返します。

署名

```
e blnt r p cg g u r mm i i S S ay( )
```

パラメータ

なし

戻り値

型: String

PageReference クラス

PageReference は、ページのインスタンス化への参照です。多数の属性の 1 つである PageReferences は URL、一連のクエリパラメータ名および値で構成されます。

名前空間

System

PageReference オブジェクトは次の目的で使用します。

- ・ ページのクエリ文字列パラメータおよび値を表示または設定する
- ・ ユーザを action メソッドの結果として異なるページにナビゲートする

インスタンス化

カスタムコントローラまたはコントローラ拡張では、次のいずれかの方法で、PageReference を参照またはインスタンス化できます。

```
g     existingPageName Pa.
```

組織すでに保存している Visualforce ページの PageReference を参照します。このプラットフォームはこのようにページを参照することで、コントローラまたはコントローラ拡張が指定されたページの有無に依存することを認識し、コントローラまたは拡張が存在する間はページが削除されないようにします。

Force.com プラットフォームでホストされる任意のページに PageReference を作成します。たとえば 'partialURL' を recordId に設定すると、`ApexSalesforceInstance` の partialURL にある Visualforce ページを参照します。同様に、`partialURL` を recordID に設定すると、指定したレコードの詳細ページを参照します。

この構文は、PageReference はコンパイル時ではなく実行時に構成されるため、`existingPageName` のページ以外の Visualforce ページの参照にはお推めしません。実行時の参照は、参照整合性システムには使用できません。したがって、プラットフォームはこのコントローラまたはコントローラ拡張機能が指定されたページの有無に依存することを認識しないため、ユーザによるページの削除を防ぐためにエラーメッセージを表示しません。

外部 URL の PageReference を作成します。次に例を示します。

ApexPages メソッドを使用して、現在のページの PageReference オブジェクトをインスタンス化することができます。次に例を示します。

要求ヘッダー

次の表に、要求時に設定される一部のヘッダーを示します。

ヘッダー	説明
Host	要求 URL で要求されるホスト名です。このヘッダーは、常に Force.com サイト要求および「私のドメイン」要求に設定されます。また、HTTP/1.1 ではなく、HTTP/1.0 を使用する場合、その他の要求ではこのヘッダーは省略可能です。
Referer	現在の要求の URL に含まれるか、リンクされた URL です。このヘッダーは省略可能です。
User-Agent	この要求を開始したプログラム (Web ブラウザなど) の名前、バージョン、拡張子のサポートです。このヘッダーは省略可能で、ほとんどのブラウザで別の値に書きできます。そのため、信頼できるヘッダーではありません。
CipherSuite	このヘッダーが存在し、空白以外の値である場合、要求には HTTPS が使用されています。それ以外の場合、要求には HTTP が使用されています。空白以外の値の内容はこの API で定義するものではなく、予告なく変更される場合があります。
X-Salesforce-SIP	要求の要求元 IP アドレスです。このヘッダーは、Salesforce のデータセンター外で開始された HTTP 要求と HTTPS 要求に常に設定されます。
	 メモ: 要求が Content Delivery Network (CDN) またはプロキシサーバーを通過する場合、要求元 IP アドレスは変更されて元のクライアント IP アドレスとは同じではなくになっている可能性があります。
X-Salesforce-Forwarded-To	この要求を処理している Salesforce インスタンスの完全修飾ドメイン名です。このヘッダーは、Salesforce のデータセンター外で開始された HTTP 要求と HTTPS 要求に常に設定されます。

例: クエリ文字列パラメータの取得

次の例では、PageReference オブジェクトを使用して、現在の URL のクエリ文字列パラメータを取得する方法を示します。この例では、`getqueryString()` メソッドは クエリ文字列パラメータを参照します。

```
bluetel = pc or lde Mac i ss a y
etbiAit = &cuAn cg ou cc i ( )
n r r ue d E EtC An T ou LM cfS &R O
e ex d A e E Etel n ewgrntph e dtR ge =gt rm Ba . gd Pa( .I ) Pa a s ( . ) ( ' ; ' )
```

次のページマークアップは、上記のコントローラから `getQueryString()` メソッドをコールします。

```
ten gortpla mp c er lde Mac a = " y " >
te tBe l & o p le a r < t k a g e tar i rg = " Rrmiv @ y is Pa a s' >
e w@ttn tegn n dñ tY ac m a o i cc ! a . a a .
ex B l go p p c < k/a a >
g p p < /a a >
```



メモ:

この例が正しく機能するためには、Visualforce ページを URL の有効な取引先レコードに関連付ける必要があります。たとえば、**I** が取引先 ID の場合、次の URL を使用します。

h e : p **Salesforce instance** // pM dg t / a E/y sE Rail 00 000B003 5

cc メソッドは、埋め込み SOQL クエリを使用して、ページの URL の パラメータで指定した取引
先を返します。 tにアクセスするためには、 eメソッドは次のように g p 名前空間を使用しま
す。

rn ます、 c g e メソッドが現在のページの f ce インスタンスを返します。 f c
は、クエリ文字列パラメータなど、Visualforce ページへの参照を返します。

`method` メソッドを使用して、`a` 指定されたクエリ文字列パラメータの名前と値の対応付けを返します。

次に、**を指定する** メソッドのコールにより、**パラメータ**自体の値を返します。

例: action メソッドの結果として新しいページに移動

カスタムコントローラまたはコントローラ拡張の action メソッドはいずれも、メソッドの結果として PageReference オブジェクトを返すことができます。PageReference の `redirect` 属性を `true` に設定すると、PageReference が指定した URL に移動します。

次の例では、`メソッドでこの移動を実装する方法を示します。この例では、メソッドで返されたa
PageReference によって、ユーザは新たに保存した取引先レコードの詳細ページに移動させます。`

e bI l t ep n m d or lâr Ci ss a y S
An t on cou cc a ;

etblAt t ou cg ou cc i ()
t n nou fce ln w n out iAac (aou == cc) a = ; ()
n rt rn ou cc a ;

e e e e bI en { grc f c i Ra vs a ()
et A dd t ent dush occ d b h // a asa a.
tn t rn ou cc i s a ;
et ne t dte tr eho d e tle h //wgo t spn f hou icca a a . (a v i)
e e e n tegr e e wf ex gAct Ra teg ndprd aor th in C w ou Bacc S a a
te e et e t et gcod r r u c a Pa R i (;)
n ter ru gcc a Pa

次のページマークアップは、上記のコントローラから メソッドをコールします。^aユーザが[保存]をクリックすると、新たに作成した取引先の詳細ページに移動します。

```
x ro m p f < /a /a >
g p p < /a /a >
```

PageReference インスタンスマソッド

PageReference インスタンスマソッド

このセクションでは、PageReference のインスタンスマソッドを次に示します。

`getAnchor()`

ページの URL で参照されるアンカーの名前を返します。これは、URL のハッシュタグ (#) より後の部分です。

`getContent()`

Web ブラウザでユーザに表示されるページの出力を返します。

`getContentAsPDF()`

ページの URL で参照されるコンポーネントの `Content` 属性に関係なくページを PDF として返します。

`getCookies()`

Cookie 名と Cookie オブジェクトの対応付けを返します。キーは Cookie 名の String で、値にはその名前を持つ Cookie オブジェクトのリストが含まれます。

`getHeaders()`

要求ヘッダーの対応付けを返します。キー文字列にはヘッダー名が含まれ、値文字列にはヘッダーの値が含まれます。

`getParameters()`

ページ URL に含まれるクエリ文字列パラメータの対応付けを返します。キー文字列にはパラメータの名前が含まれ、値文字列にはパラメータの値が含まれます。

`getRedirect()`

PageReference オブジェクトの `Redirect` 属性の現在の値を返します。

`getUrl()`

クエリ文字列パラメータやアンカーを含め、URL が本来定義されている場合は、PageReference に関する相対 URL を返します。

`setAnchor(String)`

URL のアンカー参照を指定された文字列に設定します。

`setCookies(Cookie[])`

Cookie オブジェクトのリストを作成します。このクラスと組み合わせて使用します。

`setRedirect(Boolean)`

PageReference オブジェクトの `Redirect` 属性の値を設定します。これに設定した場合、クライアント側のリダイレクトでリダイレクトが実行されます。

`getAnchor()`

ページの URL で参照されるアンカーの名前を返します。これは、URL のハッシュタグ (#) より後の部分です。

署名

```
public String getContent() { }
```

パラメータ

なし

戻り値

型: String

getContent()

Web ブラウザでユーザに表示されるページの出力を返します。

署名

```
public Blob getContent() { }
```

パラメータ

なし

戻り値

型: Blob

使用方法

返される Blob の内容は、ページの表示方法によって異なります。ページを PDF で表示すると、PDF が返されます。ページを PDF で表示しない場合、HTML が返されます。文字列として返される HTML の内容にアクセスするには、Blob メソッドを使用します。



メモ: テストメソッドで `getContent()` を使用すると、PDF として表示される Visualforce ページを併用して空白の PDF が生成されます。

このメソッドは、次のものには使用できません。

- トリガ
- スケジュール済みの Apex
- 一括処理ジョブ
- Test メソッド
- Apex メールサービス

Visualforce ページにエラーがある場合、`getContent()` が発生します。

getContentAsPDF()

例: `PageReference pr = new PageReference('pageName');` 属性に関係なくページを PDF として返します。

署名

```
public Blob getContentAsPDF() { }
```

パラメータ

なし

戻り値

型: Blob

使用方法

このメソッドは、次のものには使用できません。

- トリガ
- スケジュール済みの Apex
- 一括処理ジョブ
- Test メソッド
- Apex メールサービス

getCookies()

Cookie 名と Cookie オブジェクトの対応付けを返します。キーは Cookie 名の String で、値にはその名前を持つ Cookie オブジェクトのリストが含まれます。

署名

```
t blu   en t r p e M ge p t eno i & goaji S k ,Cs $ . ik >[ i s ( )
```

パラメータ

なし

戻り値

型: Map<String, System.Cookie[]>

使用方法

○○ C クラスと組み合わせて使用します。 C メソッドによって設定された「 p 」プレフィック
ス付きの Cookie のみを返します。

getHeaders()

要求ヘッダーの対応付けを返します。キー文字列にはヘッダーナイムが含まれ、値文字列にはヘッダーの値が含まれます。

署名

```
t blu   er p@tM g p g gdir < a i S , i S H & ( )
```

パラメータ

なし

戻り値

型: Map<String, String>

使用方法

この対応付けを変更して、PageReference オブジェクトの範囲内に保持できます。たとえば、次のように指定できます。

```
    e integrat ed c u Ra p H . a ( .D ) ( 9 9 9 9 / / ; )
```

要求ヘッダーの説明については、「[要求ヘッダー](#)」を参照してください。

getParameters()

ページ URL に含まれるクエリ文字列パラメータの対応付けを返します。キー文字列にはパラメータの名前が含まれ、値文字列にはパラメータの値が含まれます。

署名

```
    t blu er p atM ege pt g g i km aiS , iS > Pa a s ( )
```

パラメータ

なし

戻り値

型: Map<String, String>

使用方法

この対応付けを変更して、PageReference オブジェクトの範囲内に保持できます。たとえば、次のように指定できます。

```
    e nt gre t ftc rm u Ral pm PaI a s ( . ) i ( 'D , ' y ; )
```

パラメータキーでは、大文字と小文字は区別されません。次に例を示します。

```
    t mr ss a ( A e te negrntpe e ct ge gt rm Ba . g rm Pa ( .m ) m Pa a s ( . ) ( y Pa a a == ) A e te negrntpe e ct ge gt rm Ba . g rm Pa ( .m ) m Pa a s ( . ) ( y a a a ; )
```

getRedirect()

PageReference オブジェクトの属性の現在の値を返します。

署名

```
    e blB tn e do t pc gl r ic a R i ( )
```

パラメータ

なし

戻り値

型: Boolean

使用方法

PageReference オブジェクトの URL が ローカル URL または ドメイン外の Web サイトに設定されている場合、
属性が `l` または `f` のどちらに設定されているかに関係なく、常にリダイレクトされます。

`getUrl()`

クエリ文字列パラメータやアンカーを含め、URL が本来定義されている場合は、PageReference に関連付けられた相対 URL を返します。

署名

```
public String getUrl();
```

パラメータ

なし

戻り値

型: String

`setAnchor(String)`

URL のアンカー参照を指定された文字列に設定します。

署名

```
public void setAnchor(String anchor);
```

パラメータ

`anchor`

型: String

戻り値

型: System.PageReference

例

たとえば、`HttpRequest request = apexPages.currentPage();` です。

`setCookies(Cookie[])`

Cookie オブジェクトのリストを作成します。クラスと組み合わせて使用します。

署名

```
public void setCookies(Cookie[] cookies);
```

パラメータ

cookies

型: System.Cookie[]

戻り値

型: Void

使用方法



重要:

- Apex の Cookie 名と値セットは URL 符号化されています。つまり、@などの文字は % 記号および 16 進数表現に置き換えられます。
- Cookie の値を null に設定すると、期限切れの属性の設定ではなく、空の文字列値の Cookie を送信します。
- Cookie の作成後は、Cookie のプロパティを変更することはできません。
- 機密情報を Cookie に格納する場合は注意してください。Cookie の値に関係なくページはキャッシュされます。動的なコンテンツを生成するために Cookie の値を使用する場合は、ページキャッシュを無効にする必要があります。詳細は、Salesforce オンラインヘルプの「Force.com サイトページのキャッシュ」を参照してください。

setRedirect(Boolean)

PageReference オブジェクトの *etc* 属性の値を設定します。true に設定した場合、クライアント側のリダイレクトでリダイレクトが実行されます。

署名

```
public void setRedirect(boolean etc) { }
```

パラメータ

redirect

型: Boolean

戻り値

型: System.PageReference

使用方法

この種類のリダイレクトは HTTP GET 要求を実行し、POST を使用してビューステートを更新します。false に設定した場合、リダイレクトはサーバ側の転送で実行されます。これは参照先ページが同じコントローラを使用し、参照元ページで使用される拡張の適切なサブセットを含む場合にのみビューステートを維持します。

PageReference オブジェクトの URL が http://www.salesforce.com 以外の Web サイト、または別のコントローラまたはコントローラ拡張を使用するページに設定されている場合、*etc* 属性が true または false のどちらに設定されているかに関係なく、常にリダイレクトされます。

SelectOption クラス

このオブジェクトは Visualforce ブロックの、`value`、`label`、または `disabled` 属性に指定可能な値のいずれかを指定します。

名前空間

System

このオブジェクトは、エンドユーザに表示されるラベルと、オプションが選択された場合にコントローラに返される値で構成されます。このオブジェクトは無効な状態で表示することもできます。その場合、ユーザはオプションとして選択することはできませんが、表示はされます。

インスタンス化

カスタムコントローラまたはコントローラ拡張では、次のいずれかの方法で、SelectOption をインスタンス化できます。

`new SelectOption(String value, String label, Boolean isDisabled)`

`value` は、ユーザがオプションを選択した場合にコントローラに返される String です。`label` は、オプション選択肢としてユーザに表示される String です。`isDisabled` は Boolean で、これを `true` に設定すると、ユーザはオプションを選択できませんが、表示することができます。

`new SelectOption(String value, String label)`

`value` は、ユーザがオプションを選択した場合にコントローラに返される String です。`label` は、オプションの選択肢としてユーザに表示される String です。`isDisabled` の値は指定されないため、ユーザはオプションの表示と選択を行えます。

例

次の例では、SelectOptions オブジェクトのリストを使用して Visualforce ページのコンポーネントに指定可能な値を提供する方法を示します。次のカスタムコントローラでは、メソッドは使用可能な SelectOption オブジェクトのリストを定義して返します。

```
public class MyController {
    public List getOptions() {
        List options = new List();
        options.add(new SelectOption('A', 'Apple'));
        options.add(new SelectOption('B', 'Banana'));
        options.add(new SelectOption('C', 'Cherry'));
        options.add(new SelectOption('D', 'Date'));
        options.add(new SelectOption('E', 'Elderberry'));
        options.add(new SelectOption('F', 'Fig'));
        options.add(new SelectOption('G', 'Grape'));
        options.add(new SelectOption('H', 'Honeydew'));
        options.add(new SelectOption('I', 'Iceberg'));
        options.add(new SelectOption('J', 'Jalapeno'));
        options.add(new SelectOption('K', 'Kumquat'));
        options.add(new SelectOption('L', 'Lemon'));
        options.add(new SelectOption('M', 'Mango'));
        options.add(new SelectOption('N', 'Nectarine'));
        options.add(new SelectOption('O', 'Orange'));
        options.add(new SelectOption('P', 'Peach'));
        options.add(new SelectOption('Q', 'Quince'));
        options.add(new SelectOption('R', 'Rambutan'));
        options.add(new SelectOption('S', 'Strawberry'));
        options.add(new SelectOption('T', 'Tangerine'));
        options.add(new SelectOption('U', 'Ugli fruit'));
        options.add(new SelectOption('V', 'Vanilla'));
        options.add(new SelectOption('W', 'Watermelon'));
        options.add(new SelectOption('X', 'Xigua'));
        options.add(new SelectOption('Y', 'Yuzu'));
        options.add(new SelectOption('Z', 'Zucchini'));
        return options;
    }
}
```

次のページマークアップで、`<apex:selectList>` タグは上記のコントローラの `getOptions()` メソッドを使用して、使用可能な値のリストを取得します。`apex:selectItem` は、`as` 属性の子であるため、オプションはチェックボックスとして表示されます。

```

    <apex:selectList value="selectedCategory" multiple="true">
        <apex:selectItem value="Category A" label="Category A" checked="checked"/>
        <apex:selectItem value="Category B" label="Category B"/>
        <apex:selectItem value="Category C" label="Category C"/>
        <apex:selectItem value="Category D" label="Category D"/>
        <apex:selectItem value="Category E" label="Category E"/>
    </apex:selectList>

```

SelectOption インスタンスマソッド

SelectOption インスタンスマソッド

`options` のインスタンスマソッドを次に示します。

`getDisabled()`

SelectOption オブジェクトの `disabled` 属性の現在の値を返します。

`getEscapeItem()`

SelectOption オブジェクトの `escapeItem` 属性の現在の値を返します。

`getLabel()`

ユーザに表示されるオプションのラベルを返します。

`getValue()`

ユーザがオプションを選択した場合にコントローラに返されるオプション値を返します。

`setDisabled(Boolean)`

SelectOption オブジェクトの `disabled` 属性の値を設定します。

`setEscapeItem(Boolean)`

SelectOption オブジェクトの `escapeItem` 属性の値を設定します。

`setLabel(String)`

ユーザに表示されるオプションラベルの値を設定します。

`setValue(String)`

ユーザがオプションを選択した場合にコントローラに返されるオプション値の値を設定します。

`getDisabled()`

SelectOption オブジェクトの 属性の現在の値を返します。

署名

```
e b i B tn e d o p c g b l d i D a i s a ( )
```

パラメータ

なし

戻り値

型: Boolean

使用方法

例　　bl d r u を に設定した場合、オプションは表示されますが、選択できません。l を f を Di si s に設定した場合、オプションは表示され、選択できます。

`getEscapeItem()`

SelectOption オブジェクトの E c 属性の現在の値を返します。

署名

```
e b i B tn e d o t p c g E c m i P p a s a ( )
```

パラメータ

なし

戻り値

型: Boolean

使用方法

例　　m d r e c u を に設定した場合、重要な HTML および XML 文字はこのコンポーネントによって生成された HTML 出力でエスケープされます。d が f に設定されている場合、項目は書き込まれたとおりに表示されます。

`getLabel()`

ユーザに表示されるオプションのラベルを返します。

署名

```
e b l i t e r p c g f l i S L a ( )
```

パラメータ

なし

戻り値

型: String

getValue()

ユーザがオプションを選択した場合にコントローラに返されるオプション値を返します。

署名**パラメータ**

なし

戻り値

型: String

setDisabled(Boolean)SelectOption オブジェクトの 属性の値を設定します。^a**署名****パラメータ***isDisabled*

型: Boolean

戻り値

型: Void

使用方法例　　`bl d r u` を `l` に設定した場合、オプションは表示されますが、選択できません。
 `l` を `f` に設定した場合、オプションは表示され、選択できます。**setEscapeItem(Boolean)**SelectOption オブジェクトの 属性の値を設定します。^a**署名**

パラメータ

itemsEscaped

型: Boolean

戻り値

型: Void

使用方法

`m d rE c u` を `に設定した場合、重要な HTML および XML 文字はこのコンポーネントによって生成された HTML 出力でエスケープされます。` `d E c` が `f` に設定されている場合、 `項目は書き込まれたとおりに表示されます。`

setLabel(String)

ユーザに表示されるオプションラベルの値を設定します。

置首

d p e b n l r l y g l i i L s a (i S a)

パラメータ

label

型: String

戾り値

型: Void

setValue(String)

ユーザがオプションを選択した場合にコントローラに返されるオプション値の値を設定します。

署名

d p c d n u r Mg ui Vl s a (i S v a)

パラメータ

value

型: String

戾り値

型: Void

StandardController クラス

標準コントローラの拡張を定義する場合は、StandardController を使用します。

名前空間

ApexPages

使用方法

StandardController オブジェクトは、salesforce.com が提供する、プリビルドされた Visualforce コントローラを参照します。StandardController オブジェクトを参照する必要があるのは、標準コントローラの拡張を定義する場合のみです。StandardController は、拡張クラスコンストラクタの単一引数のデータ型です。

インスタンス化

次の方法で、StandardController をインスタンス化することができます。

```
t n t e n d p r d e o r l a w m e x A t B a c n s a e g a n d p r d e o s t l a r C b j B a . S c a a s ( ) ; )
```

例

次の例では、StandardController オブジェクトの標準コントローラ拡張のコンストラクタでの使用方法を示します。

```
bl t e l n e p t c o f l a r n C o E s s a y $  
r n t A h t p o u f c c v i c a a  
e t x n n t n h o G m u n t z A t e e $ r s h b m p i m i i a s v a s  
e t r b l e t e b e t u c e g t v G i n d a t a m o a r o y h m f l r d R s a a  
c t e n o r l a r c // .  
t e b l i n e t x o r p l a r n e x A d E i n t e y n d p r d t o r t l a s m { C d ( o r l a r C s a a s )  
t t A m t e o m e d e c o r l a i r s C . g = r d ( c s ) . R ; ( )  
t e b l e t { p a g r g g i s G i ( )  
e n r r u t l d o n t m c H + ' d t e c a . + a + ' a . ; ' )  
}
```

次の Visualforce マークアップは、上記のコントローラ拡張をページ内で使用する方法を示します。

```
t n t e g n d p r d p o r e d a n t C a m n a c c t e a a e t x o r l a r n C " o E s = " y $ >  
t n r } g g g p ! i < > /  
x e t n e { u l d p l t a p n n < o u a c c m v v a ! p a . " a < > / > /  
x x t n e B n d m n p o l t u < a o } a v a v s a i a = " k " a > /  
x ro m p f < / a >  
g p p < / a >
```

StandardController インスタンスマソッド

StandardController インスタンスマソッド

のインスタンスマソッドを次に示します。

`addFields(List<String>)`

Visualforce ページが読み込まれると、Visualforce マークアップで参照される項目に基づいて、ページにアクセスできる項目が表示されます。このメソッドは、コントローラがそれらの項目にも明示的にアクセスできるように、`mf` に指定された各項目に参照を追加します。

cancel()

キャンセルページの PageReference を返します。

delete()

レコードを削除し、削除ページの PageReference を返します。

edit()

標準編集ページの PageReference を返します。

getId()

Visualforce ページ URL の クエリ文字列パラメータの値に基づいて、現在コンテキストにあるレコードの ID を返します。

getRecord()

Visualforce ページ URL の クエリ文字列パラメータの値に基づいて、現在コンテキストにあるレコードを返します。

reset()

新たに参照された項目へのアクセス権限を再取得するようにコントローラを強制します。このメソッドがコールされる前にレコードに加えられた変更は、すべて破棄されます。

save()

変更を保存し、更新された PageReference を返します。

view()

標準詳細ページの PageReference オブジェクトを返します。

addFields(List<String>)

Visualforce ページが読み込まれると、Visualforce マークアップで参照される項目に基づいて、ページにアクセスできる項目が表示されます。このメソッドは、コントローラがそれらの項目にも明示的にアクセスできるように、`mf` に指定された各項目に参照を追加します。

署名

```
public void addFields(List<String> fieldNames) { }
```

パラメータ***fieldNames***

型: List<String>

戻り値

型: Void

使用方法

このメソッドは、レコードが読み込まれる前にコールする必要があります。通常、コントローラのコンストラクタによってコールされます。このメソッドがコンストラクタ外でコールされる場合、`StandardController.addFields` をコールする前に `StandardController.reset` メソッドを使用する必要があります。

ld mf の文字列には、AccountId などの API 項目名か、foo__r.myField__c などの項目への明示的なリレーションを使用できます。

このメソッドは、動的な Visualforce バインドで使用されるコントローラのみに使用できます。

cancel()

キャンセルページの PageReference を返します。

署名

```
bole e e e p com grn fl i c cs g . Ra a ( )
```

パラメータ

なし

戻り値

型: System.PageReference

delete()

レコードを削除し、削除ページの PageReference を返します。

署名

```
bole e e e ep eme tgr d l f i c s g . Ra ( )
```

パラメータ

なし

戻り値

型: System.PageReference

edit()

標準編集ページの PageReference を返します。

署名

```
bole e e e e p com t gr d f i c s g . Ra i ( )
```

パラメータ

なし

戻り値

型: System.PageReference

getId()

Visualforce ページ URL の クエリ文字列パラメータの値に基づいて、現在コンテキストにあるレコードの ID を返します。

署名

```
public String getId();
```

パラメータ

なし

戻り値

型: String

getRecord()

Visualforce ページ URL の クエリ文字列パラメータの値に基づいて、現在コンテキストにあるレコードを返します。

署名

```
public SObject getRecord();
```

パラメータ

なし

戻り値

型: sObject

使用方法

関連付けられた Visualforce マークアップで参照される項目のみを、この SObject でクエリすることができます。関連するオブジェクトの項目など、その他のすべての項目については、SOQL 表現を使用してクエリする必要があります。



ヒント: クエリする任意の追加項目を参照する非表示コンポーネントを使用すれば、この制約を回避できます。コンポーネントの `isAccessible` 属性を `false` に設定して、コンポーネントを非表示にします。

例

```
tx=u&p=11cc&gv=c&a=.sa&y=f
```

reset()

新たに参照された項目へのアクセス権限を再取得するようにコントローラを強制します。このメソッドがコールされる前にレコードに加えられた変更は、すべて破棄されます。

署名

```
de bilt dr pc V i s ( )
```

パラメータ

なし

戻り値

型: Void

使用方法

dd これは、~~コンストラクタ~~ がコンストラクタ外でコールされる場合にのみ使用するメソッドで、~~がコール~~ される直前にコールする必要があります。

このメソッドは、動的な Visualforce バインドで使用されるコントローラのみに使用できます。

save()

変更を保存し、更新された PageReference を返します。

署名

```
de e e e pem gr f i c s . Ra v a ( )
```

パラメータ

なし

戻り値

型: System.PageReference

view()

標準詳細ページの PageReference オブジェクトを返します。

署名

```
de e e e pem gr f i c s . Ra v i ( )
```

パラメータ

なし

戻り値

型: System.PageReference

StandardSetController クラス

オブジェクトを使用して、Salesforce が提供する、プリビルドされた Visualforce リストコントローラと同様のリストコントローラ、またはその拡張としてリストコントローラを作成することができます。

名前空間

ApexPages

使用方法

クラスには、**プロトタイプオブジェクト**も含まれます。これは、Visualforce の StandardSetController クラスに含まれる单一の **sObjects** です。プロトタイプオブジェクトの項目が設定されている場合、それらの値は、保存操作中に使用されます。つまり、値は設定されたコントローラコレクションのすべてのレコードに適用されます。これは、一括更新(オブジェクトのコレクション内の項目に同一の変更を適用)を実行するページを記述するときに役立ちます。



メモ: 他の Salesforce オブジェクトに必要な項目は、プロトタイプオブジェクトに使用される場合にも必要です。

インスタンス化

次のいずれかの方法で、StandardSetController をインスタンス化することができます。

- **sObjects のリストを使用する場合:**

- **クエリロケータを使用する場合:**

メモ: StandardSetController のレコード数の上限は 10,000 件です。10,000 件を超えるレコードを返すクエリロケータを使用して StandardSetController をインスタンス化すると、LimitException が発生します。ただし、10,000 件を超えるレコードのリストを使用して StandardSetController をインスタンス化すると、例外が発生する代わりに、レコードが上限まで切り捨てられます。

例

次の例では、StandardSetController オブジェクトのカスタムリストコントローラのコンストラクタでの使用方法を示します。

```
public class AccountListController {
    private final StandardSetController<Account> controller;
    private final String query;

    public AccountListController() {
        controller = new StandardSetController<Account>(Account.class);
        query = 'SELECT Name, Type, Industry, AnnualRevenue, 
                (SELECT Name, Phone, Email FROM Contacts) 
                FROM Account';
    }

    public StandardSetController<Account> getController() {
        return controller;
    }

    public StringgetQuery() {
        return query;
    }
}
```

```

e t e newn ex A t C ent gt apds t e eld retC b Bat S g a os ro D c ( as@ a L Y
} e e et E Co tT Ct nL ro[S uM appD s ] AR O O i y; )
t e tn r rm o C s ;
} s ;

e e z len It o d t eGru l/i iora srd f ca i s s
t blt ten pto ut et pp g roL u i s pp i y o i i s ( ) s
ntr rut t en ro e u e tn pp o L g esd O ci y s ) . R s ;( )
}

```

次の Visualforce マークアップは、上記のコントローラをページ内で使用する方法を示します。

```

x t e n gorppld p cn & rs u n a pp a 2C="> iys "
x ex e B el{go pbl p fc aut< n kroa ju a pp r va o a" ! i i s "v = a" >
e x e n lo ue mpeu } o < m a v a" ! N "a > /
e x e n lo ue mpeut ol ) a v a" ! D s "a > /
ex ex e B l go pbl p c < T k/a a a>
B l go p p c < k/a a >
g p p < / a a

```

StandardSetController インスタンスマソッド

StandardSetController インスタンスマソッドを次に示します。

`cancel()`

元のページ（わかっている場合）、またはホームページの PageReference を返します。

`first()`

レコードの最初のページを返します。

`getCompleteResult()`

セット内に存在するレコード数がレコード数の上限を超えているかどうかを示します。false の場合、レコード数がリストコントローラを使用して処理できる数を超えています。レコード数の上限は 10,000 レコードです。

`getFilterId()`

現在のコンテキストでの検索条件の ID を返します。

`getHasNext()`

現在のページセットの後に、より多くのレコードがあるかどうかを示します。

`getHasPrevious()`

現在のページセットの前に、より多くのレコードがあるかどうかを示します。

`getListviewOptions()`

現在のユーザが使用できるリストビューのリストを返します。

`getPageNumber()`

現在のページセットのページ番号を返します。最初のページは 1 を返します。

getPageSize()

各ページセットに存在するレコード数を返します。

getRecord()

選択したレコードへの変更を示す `sObject` を返します。クラス内に含まれるプロトタイプオブジェクトを取得し、一括更新の実行に使用されます。

getRecords()

現在のページセットにある `sObject` のリストを返します。このリストは不变であるため、`clone()` をコールできません。

getResultSetSize()

セットに存在するレコード数を返します。

getSelected()

選択されている `sObject` のリストを返します。

last()

レコードの最後のページを返します。

next()

レコードの次のページを返します。

previous()

レコードの前のページを返します。

save()

新しいレコードを挿入するか、変更された既存のレコードを更新します。この操作が完了した後、元のページ(わかっている場合)、またはホームページの `PageReference` を返します。

setFilterID(String)

コントローラの検索条件 ID を設定します。

setpageNumber(Integer)

ページ番号を設定します。

setPageSize(Integer)

各ページセット内のレコード数を設定します。

setSelected(sObject[])

選択したレコードを設定します。

cancel()

元のページ(わかっている場合)、またはホームページの `PageReference` を返します。

署名

bole e e e pem grn fl i c cs . Ra a ()

パラメータ

なし

戻り値

型: System.PageReference

first()

レコードの最初のページを返します。

署名

```
public PageReference first();
```

パラメータ

なし

戻り値

型: Void

getCompleteResult()

セット内に存在するレコード数がレコード数の上限を超えているかどうかを示します。false の場合、レコード数がリストコントローラを使用して処理できる数を超えています。レコード数の上限は 10,000 レコードです。

署名

```
public Boolean getCompleteResult();
```

パラメータ

なし

戻り値

型: Boolean

getFilterId()

現在のコンテキストでの検索条件の ID を返します。

署名

```
public String getFilterId();
```

パラメータ

なし

戻り値

型: String

getHasNext()

現在のページセットの後に、より多くのレコードがあるかどうかを示します。

署名

```
e bI B tne do t xp c g i H a sN a ( )
```

パラメータ

なし

戻り値

型: Boolean

getHasPrevious()

現在のページセットの前に、より多くのレコードがあるかどうかを示します。

署名

```
e bI B tn e do p c g r ou i H a s va iP s ( )
```

パラメータ

なし

戻り値

型: Boolean

getListviewOptions()

現在のユーザが使用できるリストビューのリストを返します。

署名

```
e o h t t e p d m n et o wtc g i pn s o . S O ip IV i s i Oi s ( )
```

パラメータ

なし

戻り値

型: System.SelectOption[]

getPageNumber()

現在のページセットのページ番号を返します。最初のページは 1 を返します。

署名

```
e bI te e r p g I g p r m i Na ( )
```

パラメータ

なし

戻り値

型: Integer

getPageSize()

各ページセットに存在するレコード数を返します。

署名`public integer getPageSize() { }`**パラメータ**

なし

戻り値

型: Integer

getRecord()

選択したレコードへの変更を示す sObject を返します。 クラス内に含まれるプロトタイプオブジェクトを取得し、一括更新の実行に使用されます。

署名`public sObject getRecord() { }`**パラメータ**

なし

戻り値

型: sObject

getRecords()現在のページセットにある sObject のリストを返します。 このリストは不变であるため、 `clone()` をコールできません。**署名**`public List<sObject> getRecords() { }`**パラメータ**

なし

戻り値

型: sObject[]

getPageSize()

セットに存在するレコード数を返します。

署名

~~enable~~ t te r p g I zg l u i R s i S ()

パラメータ

なし

戻り値

型: Integer

getSelected()

選択されている sObject のリストを返します。

署名

~~selected~~ e te b e t p c g c] d i c s O [S ()

パラメータ

なし

戻り値

型: sObject[]

last()

レコードの最後のページを返します。

署名

~~last~~ d l p c V i i s a ()

パラメータ

なし

戻り値

型: Void

next()

レコードの次のページを返します。

署名

bhxn d p c v i i ()

パラメータ

なし

戻り値

型: Void

previous()

レコードの前のページを返します。

署名

bh d r p c ou p v i i v i s ()

パラメータ

なし

戻り値

型: Void

save()

新しいレコードを挿入するか、変更された既存のレコードを更新します。この操作が完了した後、元のページ（わかっている場合）、またはホームページの PageReference を返します。

署名

ole e e e p em gr f i c s . Ra a ()

パラメータ

なし

戻り値

型: System.PageReference

setFilterID(String)

を返します。

パラメータ***filterId***

型: String

戻り値

型: Void

setpageNumber(Integer)

ページ番号を設定します。

署名

setpageNumber(Integer) : void

パラメータ***pageNumber***

型: Integer

戻り値

型: Void

setPageSize(Integer)

各ページセット内のレコード数を設定します。

署名

setPageSize(Integer) : void

パラメータ***pageSize***

型: Integer

戻り値

型: Void

setSelected(sObject[])

選択したレコードを設定します。

署名

setSelected(sObject[] selected) : void

パラメータ

selectedRecords

型: sObject[]

戻り値

型: Void

付録 D

実行ガバナと制限について

Apex はマルチテナント環境で実行するため、Apex ランタイムエンジンは、回避 Apex が共有リソースを独占しないようさまざまな制限事項を強制します。一部の Apex コードが制限を超える場合、関連付けられたガバナは、処理できない実行時例外を発行します。

Apex 制限、つまりガバナでは、次の表とセクションで示される統計情報を追跡し、強制的に適用します。

- トランザクション単位の Apex 制限
- トランザクション単位の認定管理パッケージの制限
- Force.com プラットフォームの Apex 制限
- 静的な Apex 制限
- サイズ固有の Apex 制限
- その他の Apex の制限

このトピックでは、コア Apex ガバナ制限に加え、[メール制限](#)も参照しやすいように、この後に含まれています。

トランザクション単位の Apex 制限

これらの制限は、Apex トランザクション単位でカウントされます。Apex 一括処理の場合、これらの制限はメソッドでレコードのバッチの実行ごとにリセットされます。

次の表では、同期 Apex と非同期 Apex (Apex 一括処理と future メソッド) が異なる場合、それぞれの制限が含まれます。制限が同じ場合、表には、同期および非同期 Apex の両方に適用される 1 つの制限のみが記載されます。

説明	同期制限	非同期制限
発行される SOQL クエリの合計数 ¹	100	200
SOQL クエリによって取得されるレコードの合計数	50,000	
g n o r o D c によって取得されるレコードの合計数	10,000	
発行される SOSL クエリの合計数	20	
1 つの SOSL クエリによって取得されるレコードの合計数	2,000	
発行される DML ステートメントの合計数 ²	150	
DML ステートメント、r o pp p または v a ss の結果として処理されるレコードの合計数	10,000	

付録 D: 実行ガバナと制限について

説明	同期制限	非同期制限
トリガ、アントラゴン、またはステートメントによって繰り返しトリガする Apex 呼び出しのスタックの深さの合計 ³	16	
トランザクション内のコールアウト (HTTP 要求または Web サービスコード) の合計数	10	
トランザクション内のすべてのコールアウト (HTTP 要求または Web サービスコード) の最大タイムアウト値	120 秒	
Apex 呼び出し 1 回につき許可される アノテーションを持つメソッドの合計数	10	
許可されるメソッドの合計数	10	
使用できる定義の合計数 ⁴	100	
ヒープの合計サイズ ⁵	6 MB	12 MB
Salesforce サーバの最大 CPU 時間 ⁶	10,000 ミリ秒	60,000 ミリ秒
Apex トランザクションごとの最大実行時間	10 分	
参照される固有の名前空間の最大数 ⁷	10	

¹ 親-子リレーションのサブクエリを使用する SOQL クエリでは、各親-子リレーションは追加クエリとしてカウントされます。これらのクエリタイプには、最上位クエリ数の 3 倍に制限されています。これらのリレーションクエリの行数は、全体のコード実行の行数に加算されます。静的 SOQL ステートメントの他、次のメソッドへのコールは、要求内で発行された SOQL ステートメント数としてカウントされます。

² 次のメソッドへのコールは、要求内で発行された DML クエリ数としてカウントされます。

トリガ、アントラゴン、またはステートメントによってトリガを実行しない繰り返し Apex 処理は、1 つのスタックを使用する 1 つの呼び出し内に存在します。それに対し、トリガを実行した繰り返し Apex では、コードを実行した呼び出しとは別の新しい Apex 呼び出してトリガが発生します。Apex の新しい呼び出しの実行は、1 つの呼び出での繰り返しコールよりも手間のかかる操作であるため、これらの種類の繰り返しコールのスタックの深さには、より厳しいトリガ制限があります。

⁴ 定義には、次のメソッドおよびオブジェクトが含まれます。

- ChildRelationship オブジェクト
- RecordTypeInfo オブジェクト
- PicklistEntry オブジェクト

ld f コール i s
ld f コール i s s

⁵ メールサービスのヒープサイズは 36 MB です。

⁶ CPU 時間は、1 つの Apex トランザクションで発生する Salesforce アプリケーションサーバ上でのすべての実行 (Apex コードや、このコードからコールされるすべてのプロセス(パッケージコードやワークフローなど)の実行) に対して計算されます。CPU 時間は、1 つのトランザクション専用であり、他のトランザクションからは独立しています。アプリケーションサーバの CPU 時間を消費しない操作は、CPU 時間には加算されません。たとえば、実行時間のうち DML、SOQL、および SOSL 用のデータベースに費やされた時間や、Apex コールアウトの待ち時間はカウントされません。

⁷ 1 つのトランザクションでは、参照できる固有の名前空間は 10 個のみです。たとえば、オブジェクトを更新するときに、管理パッケージでクラスを実行するオブジェクトがあるとします。その後、クラスは 2 番目のオブジェクトを更新します。つまり、他のパッケージの他のクラスを実行します。最初に 2 番目のパッケージに直接アクセスしない場合でも、同じトランザクション内で発生するため、1 つのトランザクションでアクセスする名前空間の数に含まれます。

メモ:

- 制限は、各 M h に対して個別に適用されます。
- 実行中にコードのコード実行制限を決定するには、Limits メソッドを使用します。たとえば、プログラムによってすでにコールされた DML ステートメント数を決定するには、Mn Mn D L メソッドを使用できます。または、コードに使用できる DML ステートメントの合計数を決定するには、L メソッドを使用できます。

トランザクション単位の認定管理パッケージの制限

認定管理パッケージ (AppExchange のセキュリティレビューに合格した管理パッケージ) には、一部の制限を除き、トランザクション単位の制限に対して独自の制限セットが設けられます。認定管理パッケージは salesforce.com ISV パートナーによって開発され、Force.com AppExchange から組織にインストールされ、固有の名前空間を持ちます。

ここでは、DML ステートメントについて、認定管理パッケージに別個に設定される制限の例を説明します。認定管理パッケージをインストールすると、そのパッケージ内のすべての Apex コードには、組織のネイティブコードが実行できる 150 個の DML ステートメントに加え、独自に 150 個の DML ステートメントの制限が設定されます。つまり、管理パッケージのコードとネイティブの組織のコードの両方が実行されると、1 つのトランザクションで 150 個を超える DML ステートメントが実行される可能性があります。同様に、同期 Apex については、認定管理パッケージには組織のネイティブコードの 100 個の SOQL クエリ制限に加え、独自に 100 個の SOQL クエリ制限が設定されます。他の制限についても同様です。

認定管理パッケージでは、次を除くすべてのトランザクション単位の制限は別個にカウントされます。

- ヒープの合計サイズ
- 最大 CPU 時間
- 最大トランザクション実行時間
- 固有の名前空間の最大数

これらの制限は、同じトランザクションで実行されている認定管理パッケージの数に関係なく、トランザクション全体に対してカウントされます。

また、salesforce.com ISV パートナー以外が作成した未認定の AppExchange からパッケージをインストールする場合、そのパッケージのコードには、別個に独自のガバナ制限数はありません。使用するリソースは、組織の合計数に含まれます。累積リソースメッセージと警告メールも、管理パッケージの名前空間に基づいて生成されます。

salesforce.com ISV パートナーパッケージについての詳細は、「[salesforce.com Partner Programs](#)」を参照してください。

Force.com プラットフォームの Apex 制限

次の表の制限は、Apex トランザクションに固有ではなく、Force.com プラットフォームによって適用されます。

説明	制限
24 時間あたりの非同期 Apex メソッド実行 (Apex 一括処理、future メソッド、およびスケジュール済み Apex) の最大数 ¹	250,000 か、組織内のユーザライセンス数 × 200 の大きい方の値
組織ごとの、5 秒以上かかる長時間の要求に対する同期同時要求数。 ²	10
同じホストの URL へのコールアウト要求の最大同時要求数 ³	20
同時にスケジュールされる Apex クラスの最大数	100
同時に実行される Apex 一括処理ジョブの最大数	5
Apex 一括処理ジョブの メソッドの最大同時実行数 ⁴	1
24 時間でキューできるテストクラスの合計数 ⁵	500 または組織のテストクラス数の 10 倍の大きいほう
ユーザごとに同時に開くクエリカーソルの最大数 ⁶	50
Apex 一括処理の メソッドでユーザごとに同時に開くクエリカーソルの最大数	15
Apex 一括処理の <code>c</code> および <code>f</code> メソッドでユーザごとに同時に開くクエリカーソルの最大数	5

¹ Apex 一括処理の場合、メソッド実行には、`u`、`n`、`c`、および `f` メソッドの実行が含まれます。これは組織全体の制限で、他のすべての非同期 Apex (Apex 一括処理、スケジュール済み Apex、および future メソッド) と共有されます。この制限のカウント対象となるライセンスは、Salesforce フルユーザライセンスまたは Force.com アプリケーションサブスクリプションのユーザライセンスです。Chatter Free、Chatter カスタマーユーザ、カスタマーportalユーザ、およびパートナーポータルユーザライセンスは含まれません。

² 10 個の長時間の要求が実行されている間に追加の要求を行うと、要求は拒否されます。

³ ホストは URL の一意のサブドメインで定義されます。たとえば `m.eot.mrc` と `m.o.my.c` は 2 つの異なるホストです。この制限は、同じホストにアクセスするすべての組織で計算されます。この制限を超えると、`C_O_E_C_P` が発生します。

⁴ キュー内のまだ開始されていないバッチジョブは、開始されるまで保持されます。なお、この制限により一括処理ジョブが失敗することはありません。また、複数のジョブが実行されている場合は、Apex の一括処理ジョブの `c` メソッドが並行して実行されます。

⁵ この制限は、テストの非同期実行に適用されます。これには、開発者コンソールを含め、Salesforce ユーザインターフェースから開始するテストが含まれます。

⁶ たとえば、50 個のカーソルが開いていて、同じユーザとしてログインしたままのクライアントアプリケーションが新しいカーソルを開こうとすると、50 個のカーソルのうち最も古いカーソルが解放されます。異なる Force.com 機能のカーソル制限は個別に追跡されます。たとえば、50 個の Apex クエリカーソル、Apex 一括処理 メソッドに 15 個のカーソル、Apex 一括処理の `c` および `f h` メソッドにそれぞれ 15 個のカーソル、および 5 個の Visualforce カーソルを同時に開くことができます。

静的な Apex 制限

説明	制限
トランザクション内のコールアウト (HTTP 要求または Web サービスコール) のデフォルトのタイムアウト値	10 秒
コールアウト要求または応答 (HTTP 要求または Web サービスコール) の最大サイズ ¹	3 MB
SOQL クエリの最大実行時間。この時間を超えると、Salesforce でトランザクションをキャンセルできます。	120 秒
Apex リリース内のクラスとトリガの最大コードユニット数	5,000
ループリストのバッチサイズ用	200
の 1 回の Apex 一括処理クエリで返される最大レコード数	5,000 万

¹ HTTP 要求のサイズおよび応答のサイズは、ヒープサイズの合計にカウントされます。

サイズ固有の Apex 制限

説明	制限
クラスの最大文字数	100 万
トリガの最大文字数	100 万
組織内のすべての Apex コードで使用されるコードの最大量 ¹	3 MB
メソッドのサイズ制限 ²	コンパイル形式で 65,535 バイトコード命令

¹ この制限は、AppExchange からインストールされた認定管理パッケージ (AppExchange Certified とマークされたアプリケーション) には適用されません。これらのパッケージタイプのコードは、組織のコードとは異なる独自の名前空間に属しています。AppExchange Certified パッケージについての詳細は、Force.com AppExchange オンラインヘルプを参照してください。この制限は、`Apex` アッテーションで定義されたクラスに含まれるコードにも適用されません。

² 制限を超える大規模なメソッドはコードの実行中に例外が発生する場合があります。

その他の Apex の制限

SOQL クエリのパフォーマンス

最高のパフォーマンスを得るためにには、特にトリガ内のクエリに対しては、セレクティブ SOQL クエリを使用する必要があります。実行時間が長時間に渡ることを回避するために、セレクティブ以外の SOQL クエリはシステムより終了される場合があります。100,000 件を超えるレコードを含むオブジェクトに対してトリガでセレクティブではないクエリを使用すると、エラーメッセージが表示されます。このエラーを回避するには、必ずセレクティブクエリを使用します。「より効率的な SOQL クエリ」を参照してください。

イベントレポート

システム管理者以外のユーザの場合、行動レポートが返すレコードの最大数は 20,000 件です。システム管理者の場合、100,000 件です。

Data.com クリーンアップ

Data.com クリーンアップ製品とその自動ジョブを使用していて、取引先、取引先責任者、またはリードコードで実行する SOQL クエリの Apex トリガを設定している場合、それらのオブジェクトでクエリがクリーンアップジョブに干渉する可能性があります。Apex トリガ(合計)は、パッチあたり 200 個以下の SOQL クエリにしてください。この制限を超えると、そのオブジェクトに対するクリーンアップジョブが失敗します。また、~~トリガが~~^f メソッドをコールする場合は、パッチあたり~~10~~¹⁰ 個の ~~f~~ コールに制限されます。

メール制限

受信メール制限

メールサービス: 処理されるメールメッセージの最大数 (オンデマンドメール-to-ケースの制限を含む)	ユーザライセンス数 × ユーザライセンス数 × 1,000、1 日あたりの最大数 1,000,000
メールサービス: メールメッセージの最大サイズ(本文および添付ファイル)	10 MB ¹
オンデマンドメール-to-ケース: メールの添付ファイルの最大サイズ	10 MB
オンデマンドメール-to-ケース: 処理されるメールメッセージの最大数 (メールサービスの制限に対してカウントする)	ユーザライセンス数 × 1,000、1 日あたりの最大数 1,000,000

¹ メールサービスのメールメッセージの最大数は、言語および文字セットによって異なります。

メールサービスを定義するときには、次の点に注意してください。

- メールサービスは、そのアドレスの 1 つが受信したメッセージを処理するだけです。
- Salesforce は、[オンデマンドメール-to-ケース] など、すべてのメールサービスを合計した 1 日に処理できるメッセージの総数を制限します。この制限を超えたメッセージは、各メールサービスの失敗時の応答設定に基づいて、戻す、破棄する、あるいは翌日処理するためのキューに入れます。Salesforce は、ユーザライセンス数 × 1,000 で制限値を算出します。1 日の最大は 1,000,000 件です。たとえば、ライセンス数が 10 の場合、1 日最大 10,000 件のメールメッセージを処理できます。
- sandbox 内に作成したメールサービスアドレスは、本番組織にコピーできません。
- メールサービスごとに Salesforce に通知して、送信者のメールアドレスではなく、特定のアドレスにエラーメールメッセージを送信できます。

- メール(結合された本文テキスト、本文 HTML および添付ファイル)が約 10 MB を超える場合(言語や文字セットに応じて異なる)、メールサービスはメールメッセージを拒否し、送信者に通知します。

送信メール: Apex を使用して送信する單一メールおよび一括メールの制限

API または Apex を使用して、グリニッジ標準時間(GMT)に基づいて、1 日に最大 1,000 個の外部メールアドレスに単一メール送信できます。Salesforce アプリケーションを使用して送信する単一メールはこの制限にカウントされません。取引先、取引先責任者、リード、商談、ケース、キャンペーン、カスタムオブジェクトの各ページから、組織の取引先責任者、リード、個人取引先、ユーザに個別のメールを送信する場合は、制限はありません。

単一メールを送信する場合は、次の点に注意してください。

Single Email ごとに 100 個までのメールを送信できます。

Single Email を使用して組織の内部ユーザにメールを送信するときに、ユーザ ID を指定すると、メールが 1 日あたりの制限値にカウントされません。ただし、内部ユーザのメールアドレスを指定すると、制限値にカウントされます。

グリニッジ標準時間(GMT)に基づいて、1 組織あたり 1 日に最大 1,000 個の外部メールアドレスに一括メール送信できます。各一括メール送信に含むことのできる外部メールアドレスの最大数は、次のようにエディションに応じて異なります。

エディション	一括メール送信あたりの外部アドレス制限
Personal Edition、Contact Manager Edition、および Group Edition	一括メール送信は使用できません
Professional Edition	250
Enterprise Edition	500
Unlimited Edition と Performance Edition	1,000



メモ: 次のメール制限に注意してください。

- 単一メールおよび一括メールの制限では、アドレスが一意であるかどうかは考慮されません。たとえば、メールに 10 回含まれている場合、制限に対して 10 とカウントされます。
- ポータルユーザを含め、組織の内部ユーザに送信できるメールには制限はありません。
- トライアル期間中の Developer Edition 組織と Salesforce を評価する組織では、1 日あたり 10 個を超える外部メールアドレスに一括メール送信できません。この低い制限は、組織が Winter'12 リリースより前に作成されており、一括メール送信が高い制限すでに有効になっている場合は適用されません。

用語集

[A](#) | [B](#) | [C](#) | [D](#) | [E](#) | [F](#) | [G](#) | [H](#) | [I](#) | [J](#) | [K](#) | [L](#) | [M](#) | [N](#) | [O](#) | [P](#) | [Q](#) | [R](#) | [S](#) | [T](#) | [U](#) | [V](#) | [W](#) | [X](#) | [Y](#) | [Z](#)

A

取引先

取引先とは状況を把握したい組織、会社、または消費者。たとえば、顧客、パートナー、競合会社など。

活動(行動 & ToDo)

計画されている行動やToDo。必要に応じて、取引先、取引先責任者、リード、商談またはケースなど、別の種別のレコードに関連付けることもできます。

管理者(システム管理者)

アプリケーションの設定およびカスタマイズができる組織内の1人以上のユーザ。システム管理者プロファイルに割り当てられているユーザは、管理者権限があります。

Apex

Apex は、開発者が Force.com プラットフォームサーバでフローとトランザクションの制御ステートメントを Force.com API へのコールと組み合わせて実行できるようにした、強く型付けされたオブジェクト指向のプログラミング言語です。Java に似た構文を使い、データベースのストアドプロシージャのように動作する Apex を使用して、開発者は、ボタンクリック、関連レコードの更新、および Visualforce ページなどのほとんどのシステムでのイベントに対しビジネスロジックを追加できます。Apex コードは、Web サービス要求、およびオブジェクトのトリガから開始できます。

Apex コントローラ

コントローラ、Visualforce を参照してください。

Apex ページ

「Visualforce ページ」を参照してください。

API バージョン

「バージョン」を参照してください。

アプリケーション

「アプリケーション」の短縮形です。特定のビジネス要件を扱うタブ、レポート、ダッシュボードおよび Visualforce ページなどのコンポーネントの集合です。Salesforce では、セールスおよびコールセンターなどの標準アプリケーションを提供しています。お客様のニーズに合わせてこれらの標準アプリケーションをカスタマイズできます。また、アプリケーションをパッケージ化して、カスタム項目、カスタムタブ、カスタムオブジェクトなどの関連コンポーネントと共に AppExchange にアップロードできます。そのアプリケーションを AppExchange から他の Salesforce ユーザが利用できるようにすることもできます。

B**Boolean 演算子**

Boolean 演算子をレポートプロファイルで使用して、2つの値の間の論理関係を指定できます。たとえば、2つの値の間で AND 演算子を使用すると、両方の値を含む検索結果が生成されます。同様に、2つの値の間で OR 演算子を使用すると、どちらかの値を含む検索結果が生成されます。

C**キャンペーン**

広告、ダイレクトメール、セミナーなど、見込み客を創出し、ブランド名を浸透させるために実施するマーケティング活動。

ケース

顧客からのフィードバック、問題、質問に関する詳細説明。顧客の問題の追跡および解決に使用します。

コピー

コピーは、取引先責任者や商談など、既存の項目から情報をコピーして新しい項目を作成できるボタンまたはリンクの名前です。

折りたたみ可能なセクション

詳細ページで、ユーザが表示または非表示を切り換えるセクション。

取引先責任者

取引先責任者は、取引先に関連のある個人です。

契約

契約とは、複数の集団の間での業務条件を定義した合意です。

コントローラ、Visualforce

Visualforce ページに実行する必要のあるデータおよびビジネスロジックを提供する Apex クラス。Visualforce ページは、デフォルトですべての標準オブジェクトまたはカスタムオブジェクトに付属する標準コントローラを使用、またはカスタムコントローラを使用できます。

コントローラ拡張

コントローラ拡張は、標準コントローラまたはカスタムコントローラの機能を拡張する Apex クラスです。

コンポーネント、Visualforce

などの一連のタグを使用して Visualforce ページに追加できます。Visualforce には、多くの標準コンポーネントが含まれていますが、独自のカスタムコンポーネントを作成することもできます。

コンポーネントの参照、Visualforce

組織で使用できる Visualforce の標準コンポーネントおよびカスタムコンポーネントの説明。Visualforce ページの開発フッターまたは [『Visualforce 開発者ガイド』](#) からコンポーネントライブラリにアクセスできます。

Cookie

ユーザ固有の情報、セッション固有の情報を保存する、Web アプリケーションが使用するクライアント固有のデータ。Salesforce は、特定のセッションの時間内に暗号化された認証情報を記録するためだけに、セッション「Cookie」を発行します。

カスタムコントローラ

カスタムコントローラは、標準コントローラを使用せずにページのすべてのロジックを実装する Apex クラスです。Visualforce ページを完全にシステムモードで実行する場合に、カスタムコントローラを使用します。システムモードでは現在のユーザの権限と項目レベルのセキュリティが適用されません。

カスタム項目

組織の必要に応じて Salesforce をカスタマイズするために標準項目の他に追加できる項目。

カスタムヘルプ

システム管理者が、標準項目、カスタム項目、またはカスタムオブジェクトに固有の情報を画面上に表示するために作成するカスタムテキスト。

カスタムリンク

カスタムリンクとは管理者によって定義された URL。これを使用して、Salesforce データを外部 Web サイトとバックエンドのオフィスシステムと統合します。以前は Web リンクと呼ばれていました。

カスタムオブジェクト

組織固有の情報を保存することが可能なカスタムレコード。

カスタム S コントロール



メモ: S コントロールは、Visualforce ページに置き換えられました。2010 年以降、新しい組織同様、S コントロールを作成したことのない組織は、S コントロールを作成できなくなります。既存の S コントロールに影響はありません。今後も編集できます。

カスタムリンクで使用するカスタム Web コンテンツ。カスタム S コントロールには、Java アプレット、Active-X コントロール、Excel ファイル、カスタム HTML web フォームなど、ブラウザに表示できるあらゆる種類のコンテンツを入れることができます。

カスタムアプリケーション

「アプリケーション」を参照してください。

D

データの状態

特定の時点でのオブジェクトに含まれるデータの構造。

連動項目

対応する制御項目で選択された値に基づいて、使用可能な値が表示される、カスタムの選択リストまたは複数選択の選択リストの項目。

詳細

単一のオブジェクトレコードに関する情報を表示するページ。レコードの詳細ページでは情報を表示できますが、編集ページでは変更が可能です。

レポートで、概要情報とレポートにあるすべての情報のすべての列データを含むものとを区別するための用語。[詳細の表示]/[詳細を非表示] を使用して、レポートの詳細の表示/非表示を切り替えることができます。

詳細ビュー

エージェントコンソールの中央に表示されるフレームで、コンソールの他のフレームから選択されたレコードの詳細ページが表示されます。詳細ビューには、そのオブジェクトの詳細ページで定義されたページレイアウトと同じものが表示されます。詳細ビューに表示されているレコードは、リストビューで強調表示されます。

Developer Edition

開発者がForce.com プラットフォームを使用して拡張、統合、開発するよう開発された無料でフル機能の Salesforce。Developer Edition のアカウントは、developer.force.com で登録できます。

E

メールテンプレート

新しい従業員へのウェルカムレターや、カスタマーサービス要求の申請が受信された場合の通知など、標準メッセージを知らせる書式設定されたメール。メールテンプレートは、差し込み項目でカスタマイズしたり、テキスト、HTML、またはカスタム形式で作成したりできます。

行動

行動とは、時間がスケジュールされている活動のことです。たとえば、ミーティングまたは予定された電話などです。

F

Facet

表示された親領域を facet の内容で上書きできるようにする、別の Visualforce コンポーネントの子です。

項目レベルのヘルプ

標準項目またはカスタム項目について提供できるカスタムヘルプテキスト。その項目の隣にあるヘルプアイコンにマウスポインタを停止させると表示されます。

Force.com アプリケーションメニュー

カスタマイズ可能なアプリケーション(別称「アプリケーション」)を 1 クリックで切り替えることができるメニュー。Force.com アプリケーションメニューは、ユーザインターフェースの各ページの上部に表示されます。

数式項目

カスタム項目の一種。差し込み項目、式、またはその他の値に基づいて、値を自動的に計算します。

関数

あらかじめ用意されている数式。入力パラメータを使用してカスタマイズできます。たとえば、DATE 関数は、年、月、および日付から日付データ型を作成します。

G

get 要求

get 要求は、ユーザが URL を入力するか、リンクまたはボタンをクリックして最初に Visualforce ページを要求したときに実行されます。

getter メソッド

開発者がページのマークアップにデータベースその他の計算値を表示するためのメソッド。

値を返すメソッドです。setter メソッドを参照してください。

H

該当用語はありません。

I

該当用語はありません。

J

連結オブジェクト

2つの主従関係を持つカスタムオブジェクトです。カスタム連結オブジェクトを使用して、2つのオブジェクト間の「多対多」リレーションをモデル化できます。たとえば、「バグ」という名前のカスタムオブジェクトを作成し、1つのバグを複数のケースに、また1つのケースを複数のバグに関連付けることができます。

K

該当用語はありません。

L

リード

リードとは、あなたの製品や会社に興味を示した、販売が見込める客のことです。

文字数/桁数

テキスト項目の場合、カスタム項目に入力できる最大文字数(255文字まで)を指定するパラメータ。

数値、通貨、パーセント項目の場合、整数部として入力できる桁数を指定するパラメータ。たとえば、123.98の場合は3と指定します。

M

主従関係

2つの異なる種別のレコード間にの関係で、互いにレコードを関連付けます。たとえば、取引先には商談との主従関係があります。このような種類の関係は、レコードの削除、セキュリティに影響を与え、ページレイアウトに必要な参照関係項目を作成します。

差し込み項目

差し込み項目は、メールテンプレート、メールの差し込みテンプレート、カスタムリンク、またはレコードの値を投入する数式を入力できる項目です。^tたとえば、○ r C C M は取引先責任者 N a , 差し込み項目を使用して、取引先責任者レコードの「名】項目の値を取得し、メール受信者を名前で示します。

モバイル設定

Salesforce がユーザのモバイルデバイスに転送するデータと、そのデータをモバイルデバイスで受信するユーザを決定するパラメータのセットです。複数のモバイルユーザの異なるニーズを同時に満たせるように、組織で複数のモバイル設定を作成できます。

N**メモ**

特定のレコードに関連するその他の情報。

O**オブジェクト**

Salesforce 組織に情報を保存するために使用するオブジェクト。オブジェクトは、保存する情報の種類の全体的な定義です。たとえば、ケースオブジェクトを使用して、顧客からの問い合わせに関する情報を保存できます。各オブジェクトについて、組織は、そのデータ型の具体的なインスタンスに関する情報を保存する複数のレコードを保有します。たとえば、佐藤次郎さんから寄せられたトレーニングに関する問い合わせに関する情報を保存するケースレコードと、山田花子さんから寄せられたコンフィグレーションの問題に関する情報を保存するケースレコードなどです。

オブジェクトレベルのヘルプ

カスタムオブジェクトに提供できるカスタムヘルプのテキスト。カスタムオブジェクトレコードのホーム(概要)、詳細、編集ページ、リストビューや関連リストに表示されます。

商談

商談は、販売と進行中の商談を追跡します。

組織

ライセンスユーザセットが定義された Salesforce のリリース。組織は、salesforce.com の各お客様に提供される仮想スペースです。組織には、すべてのデータおよびアプリケーションが含まれてあり、他のすべての組織から独立しています。

アウトバウンドメッセージ

アウトバウンドメッセージは、外部サービスなどの指定したエンドポイントに指定の情報を送信するワークフロー、承認、およびマイルストンアクションです。アウトバウンドメッセージは、エンドポイントに対し、特定の項目内のデータを SOAP メッセージとして送信します。アウトバウンドメッセージは、Salesforce の設定メニューで設定します。その後で、外部エンドポイントを設定する必要があります。SOAP API を使用して、メッセージのリスナーを作成できます。

所有者

レコード(取引先責任者またはケースなど)が割り当てられる個別ユーザです。

P**パッケージバージョン**

パッケージバージョンは、パッケージでアップロードされる一連のコンポーネントを特定する番号です。バージョン番号の形式は `majorNumber.minorNumber.patchNumber`(例: 2.1.3) です。メジャー番号とマイナー番号は、メジャーリリース時に指定した値に増えます。`patchNumber` は、パッチリリースにのみ生成および更新されます。

未管理パッケージはアップグレードできないため、各パッケージバージョンは単に配布用コンポーネントのセットです。パッケージバージョンは管理パッケージでより大きな意味を持ちます。パッケージは異なるバージョンで異なる動作をします。公開者は、パッケージバージョンを使用して、後続のパッケージバージョンをリリースすることにより、そのパッケージを使用する既存の顧客のインテグレーションを分割することなく管理パッケージのコンポーネントを強化することができます。「パッチ」と「パッチ開発組織」も参照してください。

ページレイアウト

ページレイアウトとは、レコードの詳細ページまたは編集ページの項目、カスタムリンク、および関連リストの構成。主にユーザのページを構成するページレイアウトを使用します。Enterprise Edition、Unlimited Edition、Performance Edition、およびDeveloper Edition では、項目レベルのセキュリティを使用して、特定の項目に対するユーザのアクセス権限を制限します。

部分ページ

何らかのユーザアクションの後に、ページ全体が再読み込みされるのではなく、ページの特定の部分のみが更新される AJAX の動作です。

postback 要求

postback 要求は、ユーザが [保存] ボタンをクリックして `save` アクションをトリガする場合など、ユーザ操作で Visualforce ページの更新が必要なときに行われます。

主取引先責任者

組織の主担当を表示する会社情報の項目。

取引先、契約、または商談と関連付けられた主担当者とともに示します。取引先、契約、または商談の [取引先責任者の役割] 関連リストのチェックボックスで指定します。

商品

組織で販売している項目またはサービス。商品は価格表で定義され、商談に追加できます。使用可能なエディションは、Professional Edition、Enterprise Edition、Unlimited Edition、Performance Edition、およびDeveloper Edition のみです。

プロトタイプオブジェクト

これは、Visualforce の `StandardSetController` クラスに含まれる單一の `Object` です。プロトタイプオブジェクトの項目が設定されている場合、それらの値は、保存操作中に使用されます。つまり、値は設定されたコントローラコレクションのすべてのレコードに適用されます。

Q

該当用語はありません。

R

参照のみ

ユーザに割り当て可能な標準プロファイルの 1 つ。アクセス権が参照のみのユーザは、組織内での役割に基づいて、情報を表示し、レポートできます。(つまり、CEO のアクセス権が参照のみの場合は、システム内の全データを表示できます。アクセス権が参照のみのユーザに西日本営業担当の役割が割り当てられている場合は、自分の役割のデータ、および階層内で自分より下の役割のデータをすべて表示できます)。

レコード

Salesforce オブジェクトの单一インスタンス。たとえば、「John Jones」は取引先責任者レコードの名前となります。

レコードタイプ

レコードタイプとは、そのレコードの標準およびカスタムの選択リスト項目の一部またはすべてを含めることができる特定のレコードに使用可能な項目。レコードタイプをプロファイルに関連付けて、含まれている選択リストの値のみがそのプロファイルのユーザに使用できるようにできます。

関連リスト

レコードに関連する項目が表示される、レコードまたは他の詳細ページのセクション。たとえば、商談の[フェーズの履歴]関連リストや、ケースの[活動予定]関連リストなど。

関連オブジェクト

特定のタイプのレコードがコンソールの詳細ビューに表示されている状況で、システム管理者がエージェントコンソールのミニビューへの表示を指定できるオブジェクトです。たとえば、システム管理者は、ケースが詳細ビューに表示されているときにミニビューに表示される項目として、関連する取引先、取引先責任者、納入商品などを指定できます。

リレーション

ページレイアウト内の関連リストおよびレポート内の詳細レベルを作成するために使われる、2つのオブジェクトの間の接続です。両方のオブジェクトの特定の項目において一致する値を使用して、関連するデータにリンクします。たとえば、あるオブジェクトには会社に関連するデータが保存されていて、別のオブジェクトには人に関連するデータが保存されている場合、リレーションを使用すると、その会社で働いている人を検索できます。

レポート

レポートは、一定の条件を満たすレコードセットを返し、行と列に整理して表示します。レポートデータは、条件で絞り込んだり、グループ化したり、グラフなどの図にして表示したりすることができます。レポートはフォルダに保存され、フォルダごとに誰にアクセス権を与えるかを制御します。表形式レポート、サマリーレポート、マトリックスレポートを参照してください。

S

Sコントロール



メモ: Sコントロールは、Visualforce ページに置き換えられました。2010 年以降、新しい組織同様、Sコントロールを作成したことのない組織は、Sコントロールを作成できなくなります。既存の Sコントロールに影響はありません。今後も編集できます。

カスタムリンクで使用するカスタム Web コンテンツ。カスタムSコントロールには、Java アプレット、Active-X コントロール、Excel ファイル、カスタム HTML Web フォームなど、ブラウザに表示できるあらゆる種類のコンテンツを入れることができます。

Salesforce API バージョン

「バージョン」を参照してください。

サイト

Force.com サイトでは、公開 Web サイトとアプリケーションを作成できます。それらは Salesforce 組織と直接統合されるため、ユーザがログインする場合にユーザ名やパスワードは必要ありません。

スケルトンテンプレート

□ o m p <タグを使用する Visualforce テンプレートの種別です。スケルトンテンプレートでは、標準構造を定義し、その構造によって後続のページの実装を要求します。

ソリューション

ソリューションとは、お客様の問題に対する解決策の詳細説明です。

T**テキスト**

文字、数値、記号を組み合わせて入力できるカスタム項目のデータ型(文字数は最大 255 文字)。

テキストエリア

各行最大 255 文字の入力ができるカスタム項目のデータ型。

ロングテキストエリア

ロングテキストエリアを参照してください。

U**ユーザインターフェース**

データモデルの表示方法を指定するレイアウトです。

V**バージョン**

項目のリリースを示す数値。バージョンを表示できる項目は、API オブジェクト、項目およびコール、Apex クラスおよびトリガ、Visualforce ページおよびコンポーネントです。

ビュー

Visualforce で定義された Model-View-Controller モデルのユーザインターフェース。

ビューステート

要求間のデータベース状態を維持するために必要なすべての情報が、ビューステートに保存されます。

Visualforce

開発者が、プラットフォームに作成されたアプリケーションのカスタムページおよびコンポーネントを容易に定義できる、単純で、タグベースのマークアップ言語。各タグが、ページのセクション、関連リスト、または項目など、大まかなコンポーネントときめの細かいコンポーネントのどちらにも対応しています。コンポーネントは、標準の Salesforce ページと同じロジックを使用して制御することができます。また、開発者が独自のロジックを Apex で記述されたコントローラと関連付けることもできます。

Visualforce ライフサイクル

ユーザセッションでページがどのように作成されて破棄されるかを示す Visualforce ページの各実行フェーズ。

Visualforce ページ

Visualforce を使用して作成された Web ページ。通常、Visualforce ページには組織に関連する情報が表示されますが、データの変更や取得も可能です。PDF ドキュメントやメールの添付ファイルなど、さまざまな方法で表示できます。また CSS スタイルに関連付けることもできます。

W**該当用語はありません。****X****該当用語はありません。**

Y

該当用語はありません。

Z

該当用語はありません。