日日是Oracle APEX

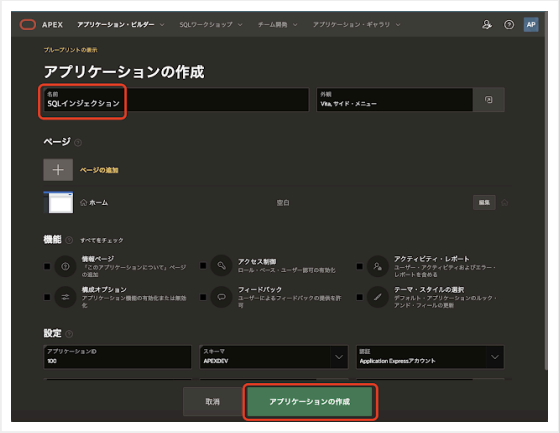
Oracle APEXを使った作業をしていて、気の付いたところを忘れないようにメモをとります。

2021年8月16日月曜日

データベース・セキュリティの活用(3) - テスト用アプリケーションの作成

ワークスペースAPEXDEVにサインインし、テストに使用するアプリケーションを作成します。このアプリケーションには、意図的にSQLインジェクションに脆弱であるSQLを含めます。

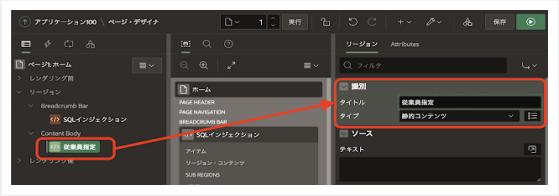
アプリケーション作成ウィザードを起動し、空のアプリケーションを作成します。**名前**は**SQLインジェクション**とします。**アプリケーションの作成**を実行します。



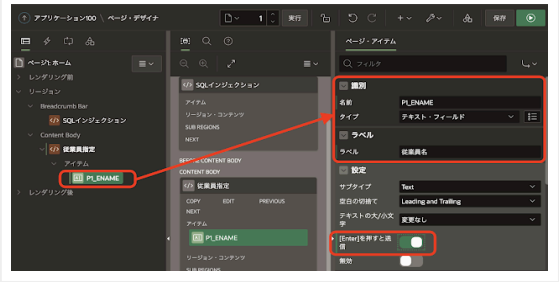
アプリケーションが作成されたら、**ページ・デザイナ**で**ホーム**・ページ(ページ番号1)を開きます。

ホーム・ページにリージョンを３つ作成します。左ペインの**レンダリング・ビュー**にある**Content Body**上でコンテキスト・メニューを開き、**リージョンの作成**を実行します。

１つ目のリージョンは、**識別**の**名前**を**従業員指定**、**タイプ**を**静的コンテンツ**とします。



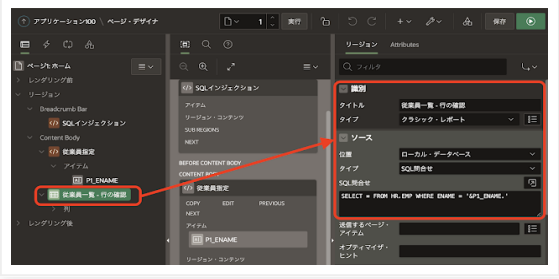
従業員名を入力するページ・アイテムを作成します。Content Body以下のリージョン**従業員指定**の上でコンテキスト・メニューを開き、**ページ・アイテムの作成**を実行します。**識別**の**名前**を**P1_ENAME**、**タイプ**を**テキスト・フィールド**、**ラベル**を**従業員名**、**設定**の**[Enter]を押すと送信**を**ON**にします。



表HR.EMPを一覧するリージョンを作成します。**リージョンの作成**を実行し、**識別**の**タイトル**を**従業員一覧 - 行の確認**とし、**タイプ**には**クラシック・レポート**を選択します。**ソース**の**位置**は**ローカル・データベース**、**タイプ**は**SQL問合せ**、**SQL問合せ**として以下を記述します。

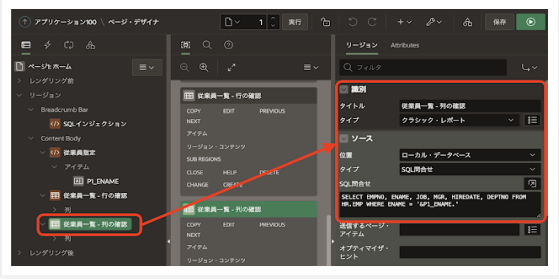
```
SELECT * FROM HR.EMP WHERE ENAME = '&P1_ENAME.'
```

上記の**SELECT**文は文中に置換文字列**&P1_ENAME.**を含むため、SQLインジェクションに対して脆弱になっています。正しくはバインド変数**:P1_ENAME**を使用します。今回はデータベースのセキュリティ機能の効果を確認するため、あえてこのようにしています。



もうひとつクラシック・レポートのリージョンを作成します。**リージョンの作成**を実行し、**識別**の**タイトル**を**従業員一覧 - 列の確認**とします。**SQL問合せ**に以下を記述します。検索対象の列から**SAL**および**COMM**を除外しています。

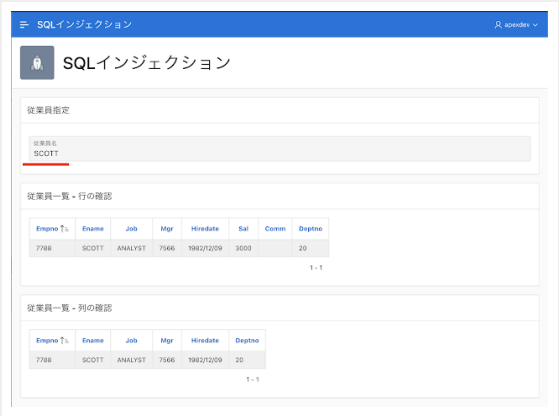
```
SELECT EMPNO, ENAME, JOB, MGR, HIREDATE, DEPTNO FROM HR.EMP WHERE ENAME = '&P1_ENAME.'
```



これで表**HR.EMP**を一覧するページが完成しました。ページを実行し動作を確認します。

ワークスペースAPEXDEVにサインインしたユーザー(通常は**APEXDEV**)にて**サインイン**します。

従業員名に**SCOTT**を入力して**Enter**を押します。SCOTTの情報がそれぞれのクラシック・レポートに 1 行だけ表示されています。

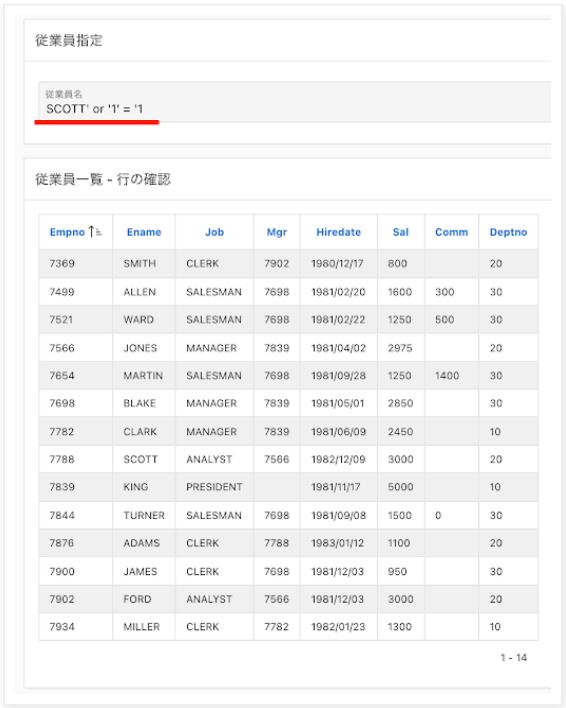


アプリケーションが従業員名として想定している入力は単一の従業員名です。別の従業員名を入力しても、レポートに表示されるのは 1 行のみです。

SQLインジェクションをシミュレートするため、**従業員名**として以下を入力します。

SCOTT' or '1' = '1

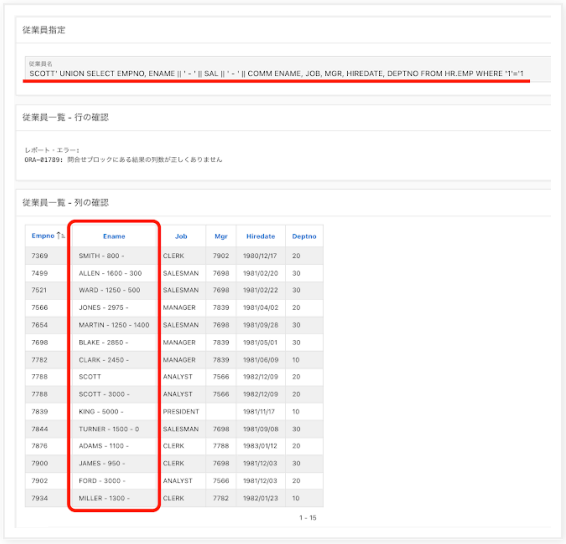
レポートにはすべての従業員の情報が一覧されます。



続いて、**従業員名**に以下を入力します。

SCOTT' UNION SELECT EMPNO, ENAME || ' - ' || SAL || ' - ' || COMM ENAME, JOB, MGR, HIREDATE, DEPTNO FROM HR.EMP WHERE '1'='1

レポートは列SALおよびCOMMを含んでいませんが、列ENAMEにSALおよびCOMMの情報が表示されています。



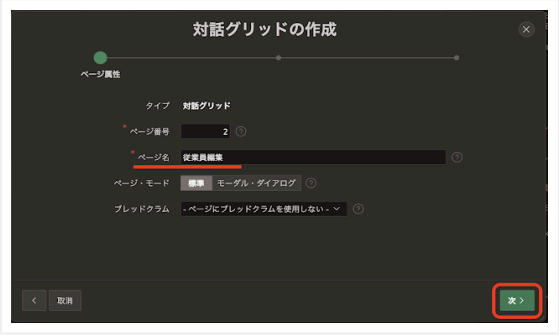
SQLインジェクションを確認するページは出来上がりました。

表HR.EMPを編集する対話グリッドのページを作成します。**ページの作成**を実行し、**ページ作成ウィザード**を起動します。

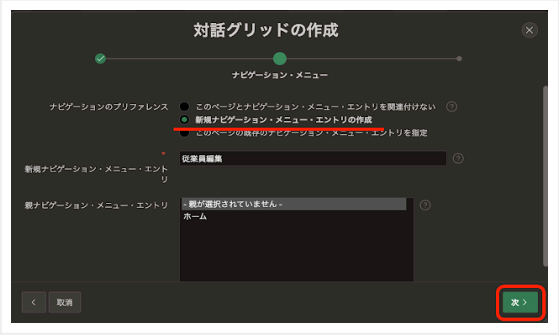
フォームを選択します。

編集可能対話グリッドを選択します。

ページ名を**従業員編集**として、**次**へ進みます。



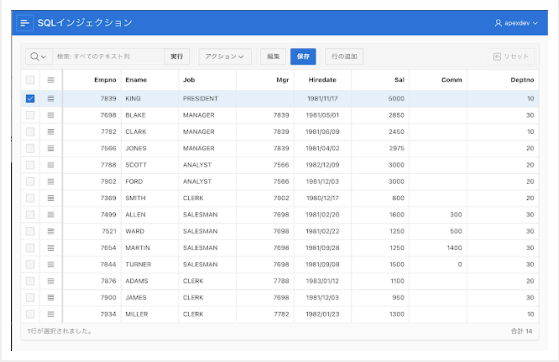
ナビゲーションのプリファレンスとして、**新規ナビゲーション・メニュー・エントリの作成**を選びます。**次**へ進みます。



データ・ソースを**ローカル・データベース**、**編集が有効**を**ON**とします。**ソース・タイプ**として**SQL問合せ**を選択し、**SQL問合せ**として以下のSQLを記述します。**作成**をクリックします。

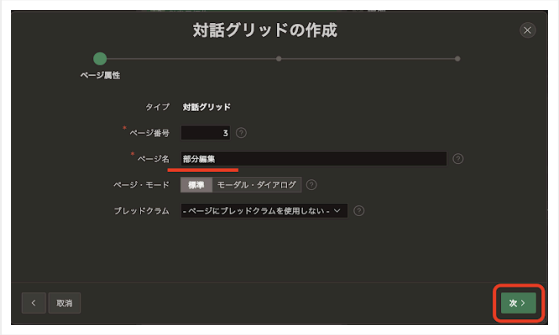
SELECT ROWID, EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO FROM HR.EMP

作成したページを実行すると、以下になります。従業員の挿入、更新、削除といった操作を行えます。



対話グリッドのページをもう一枚作成します。操作の流れは同じなので、異なる部分だけを示します。**ページ作成ウィザード**を起動し、**編集可能対話グリッド**を選択します。

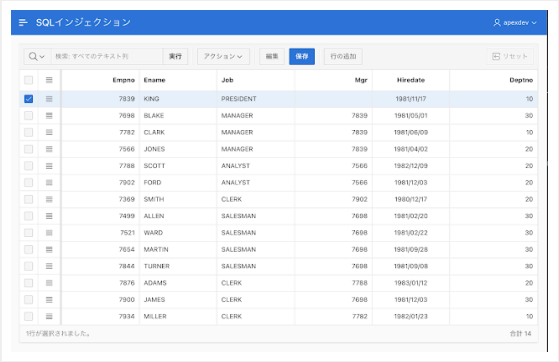
ページ名は**部分編集**とします。



SQL問合せからは列SALおよびCOMMを除いた以下のSQLを、**SQL問合せ**として記述します。**作成**をクリックします。

SELECT ROWID, EMPNO, ENAME, JOB, MGR, HIREDATE, DEPTNO FROM HR.EMP

作成されたページは以下になります。列SAL(Sal)とCOMM(Comm)が除かれています。



以上でテストに使用するアプリケーションが完成しました。

アプリケーションのエクスポートは以下に置いています。
https://github.com/ujnak/apexapps/blob/master/exports/sqlinjection-for-seminar.sql

続く

Yuji N. 時刻: 17:49

共有

‹ ホーム ›

ウェブ バージョンを表示

自己紹介

Yuji N.

日本オラクル株式会社に勤務していて、Oracle APEXのGroundbreaker Advocateを拝命しました。
こちらの記事につきましては、免責事項の参照をお願いいたします。

詳細プロフィールを表示

Powered by Blogger.