全体設計書 (書籍販売システム【Laravel版】)

■ プロジェクト名 書籍販売システム (BookMangementSystem) 【Laravel版】

■ システム名 bmsweb10L

■ 文書管理番号

	審査			承認				
版	氏名	日付	印	氏名	日付	印		
1.0	神田ITスクール コンテンツ開発チーム	2022/1/11	富田	神田ITスクール 崔 光護	2022/1/13	崔		

	冷

システム名	書籍管理アプリケーション(Larave l版 Ver 1.0)	作成日	作成者	冨田 茜
	bmsweb10L	更新日	更新者	神田ITスクール
				コンテンツ開発Team

				12,22	
No.	ドキュメント名				
1 2	目次				
3 4	改訂記録				
5 6 7	Eclipse環境構築				
8 9	機能一覧				
10 11	ルーティング				
12 13	詳細設計書(Model)				
14 15	処理機能記述書				
16 17	画面遷移図				
18 19	エラー処理詳細記述書				
20 21	テスト実施				
22 23	DB設計書				
24 25					
26 27					
28 29					
30 31 32					
33 34					
35 36					
37 38					
39 40					
41 42					
43 44					
45 46					
47					

	-LA		システム名 書籍管理アプリケーション(Larave I版Ver 1.0)			作成日 更新日	作成者	国田 茜 神田ITスクール コンテンツ開発Team
	改訂	丁記録		bmsweb10L		更新日	更新者	神田ITスクール
								コンテンツ開発Team
版	改訂日付	改訂理由	改訂対象		改訂内容			改訂者
1. 0		初版	全て		改訂内容 初版作成			
	1							

Eclipseファイル構成

システム名	書籍管理アプリケーション (Laravel版Ver1.0)	作成日	作成者	富田 茜
	bmsweb10L	更新日	更新者	神田ITスクール
				コンテンツ開発Team

機能フローID	Eclipse環境準備
機能フロー名	
フロー区分	

説明

```
bmsweb10L_英字氏名
                         //Eclipseのプロジェクト名(アプリケーション名)
    -- app
                 // アプリケーションのプログラム部分がまとめられるところ
             Http
                       // Webアプリケーションにアクセスしたときの処理をまとめておくところ
                   - Controllers
                                   // コントローラのファイルを格納する
                            - — ListController.php
                                                    // 一覧表示処理のコントローラ
                             — InsertController.php
                                                    // 登録処理のコントローラ
                             - — DeleteController.php
                                                    // 削除処理のコントローラ
                            - — UpdateController.php
                                                    // 更新処理のコントローラ
                             - — DetailController.php
                                                    // 詳細処理のコントローラ
                            — SearchController.php
                                                    // 検索処理のコントローラ
             - Models
                         // データの管理を行うところ
                  └ — Book Info. php
                                    // DBアクセス処理ファイル
                       // リソース関係の配置場所 (プログラムのテンプレートファイルなどが用意されます)
    -- resources
                 menu.blade.php
                                    // メニュー画面
                - list.blade.php \sf J
                                     // 書籍一覧画面
                insert.blade.php
                                     // 書籍登録画面
                — delete. blade. php
                                    // 書籍削除画面
                                    // 書籍更新画面
                -update.blade.php
                — detail. blade. php
                                    // 書籍詳細画面
                - search. blade. php
                                    // 書籍検索画面
               └─ error.blade.php
                                    // エラー画面
   └──routes
                   // ルート情報の保存場所(アクセスするアドレスに割り当てられるプログラムの情報などが記されています)
                           // ルーティング情報を記述するファイル
            └ — web. php
```

機能一覧		システム名	書籍管理アプリケーション(Larave l版 Ver 1.0)	作成日	作成者	富田 茜
			bmsweb10L	更新日	更新者	神田ITスクール
						コンテンツ開発Team
機能フローID		説明				
機能フロー名						
機能フローID 機能フロー名 フロー区分						

機能フロー図

No.	機能名	フェーズ	備考
1	一覧機能	1	※サンプルを利用した場合は、機能の仕組みを理解して下さい。
2	登録機能	1	
3	削除機能	1	
4	更新機能	1	
	詳細機能	1	
6	検索機能	2	
7	入力チェック	2	入力値に問題がある場合エラー画面へ遷移させる。(詳細はエラー処理をご参照下さい)
8	テストケース実施	3	

ルーティング

システム名	書籍管理アプリケーション(Laravel版Ver1.0)	作成日	作成者	冨田 茜
	bmsweb10L	更新日	更新者	神田ITスクール
				コンテンツ開発Team

機能フローID 機能フロー名 フロー区分

説明

機能フロー図

No.	機能名	HTTPメソッド	パス	コントローラ名	アクション名	クロージャの処理	
1	トップページ	GET	/			view('menu')	メニュー画面を表示
2	一覧機能	GET	/list	ListController	index		
3	登録機能	GET	/insert			view('insert')	書籍登録画面を表示
4	登録機能	POST	/insert	InsertController	insert		
5	登録機能	GET	/insert_redirect			redirect('/insert')	書籍登録画面ヘリダイレクト
6	詳細機能	GET	/detail	DetailController	index		
7	更新処理	GET	/update	UpdateController	index		変更画面初期表示
8	更新処理	POST	/update	UpdateController	update		変更画面登録処理
9	削除処理	GET	/delete	DeleteController	index		
10	検索処理	POST	/search	SearchController	index		
11	検索処理	GET	/list_redirect			redirect('/list')	一覧画面へリダイレクト
12	エラー処理	GET	/error	DeleteController	index		

■ ルーティングの例を以下に示します。 ※ routesフォルダのweb. phpに記入してください。

```
<?php
    use Illuminate\Support\Facades\Route;
                                    クロージャ(無名関数)を使う場合、
                                   view関数の引数にテンプレート名を指定し
    use App\Http\Controllers\ListControlle
    必要に応じて追加
    // トップページ
    Route::get('/', function () {
                                                 コントローラを呼び出す場合、
       return view('menu');
                                                 呼び出すコントローラクラス名と、アクション
    });
                                                 名を指定します。
                                                ※その際、useを使用してコントローラを参照
    // 一覧表示
                                                 することを忘れないように!
    Route: get('/list',
[ListController::class, 'index']);
    必要に応じて追加
```

詳細設計書(Model)		システム名		書籍管理アプリケーション(Laravel版Ver1.0)	作成日	作成者	富田 茜
				bmsweb10L	更新日	更新者	神田ITスクール
							コンテンツ開発Team
機能フローID			説明				
機能フロー名							
フロー区分							

機能フロー図

■ BookInfo. php

- ・以下の設定に従いモデルクラスであるBook Info. phpファイルの作成を行う。
 - ・以下のメンバ変数を定義する

①BookInfoモデルクラスの対象となるテーブルを指定

protected \$table = 'bookinfo';

②プライマリーキーを指定

protected \$primarykey = 'isbn';

③プライマリーキーがオートインクリメントではないため設定をオフ

public \$incrementing = false;

④データの作成日時、更新日時がデフォルトで自動更新されるので、オフ

public \$timestamps = false;

・以下の関数を定義する

① 引数のISBNを元にDBのbookinfoテーブルから該当書籍データの検索を行う関数定義

関数名 : selectByIsbn 引数 : \$isbn

戻り値 : Collectionクラス

クエリ例 : BookInfo∷where('isbn', '=', \$isbn) → get();

呼び出し元 :書籍更新機能(UpdateController)、書籍削除機能(DeleteController)、書籍詳細機能(DetailController)

② 引数の書籍データを元にDBのbookinfoテーブルに新規登録処理を行う関数定義

関数名 : insertBook 引数 : \$insertBook 戻り値 : -

庆り世 -

クエリ例 : BookInfo∷insert(\$insertBook); 呼び出し元 : 書籍登録機能(InsertController)

③ 引数の書籍データを元にDBのbookinfoテーブルから該当書籍データの更新処理を行う関数定義

関数名: updateBook 引数: \$isbn, \$newBooK

戻り値 :-

クエリ例 : BookInfo∷where('isbn','=',\$isbn) -> update(\$newBook);

呼び出し元 :書籍更新機能(UpdateController)

④ 引数の書籍データ (ISBN) を元にDBのbookinfoテーブルから該当書籍データの削除処理を行う関数定義

関数名 : deleteBook 引数 : \$isbn 戻り値 : -

クエリ例 :BookInfo∷where('isbn', '=', \$isbn) -> delete();

呼び出し元 :書籍削除機能(DeleteController)

	システム名	書籍管理アプリケーショ	ン(Laravel版Ver1.0)	作成日	作成者	富田 茜
詳細設計書(Mode		bmsweb10L		更新日	更新者	神田ITスク
						コンテンツ
	_					
·ID		説明				
名						
Z						
€ 引動のタニーカキニ	=DB@bookinfo=	。 該当書籍データの絞込み検索処理を	仁 2 胴 粉 中 華			
関数名 : searchBook	CDBOODOOKIIIIO 7 — J NOSI E	の数当者精ブーブの収込の快来処理を	1] 7 民致足襲			
列数 : \$isbn, \$tit	le \$nrice					
戻り値 : Collection						
	= DB::table('bookinfo');					
	k->where('isbn', 'like', '%'	\$ishn '%'):				
	k->where('isbn', 'like', '%'					
	k->where('isbn', 'like', '%'					
\$searchBoo		. фр. 100.				
呼び出し元 : 書籍検索機:						
,						

b	処理機能記述書	システム名	書籍管理アプリケーション (Laravel版Ver1.0) bmsweb10L	作成日 更新日	作成者 更新者	富田 茜神田 ゴス・
X	2.年成形心心盲		DIISMEDTOL	上 机口	史 柳有	コンテン
-ID -名 }			説明			
-図 メニュ						
No.	ファイル名		処理内容	♡ 兩面乳乳毒素	備考	
1	menu. blade. php	①メニュー画面のhtml を記述	<u> </u>	※ 画面設計書をご参考下さい		
No.	一覧(list) ファイル名 ListController.php ①	D Book Infoモデルを使用でる	<u>処理内容</u> きるようにする。		備考	
	2	・bookinfoテーブルからst	リンク、全検索ボタンから遷移してきたときに以下の処理を行う。 全件データを取得する MM bookinfo ORDER BY ISBN;			
	list. blade. php (3	③ 書籍一覧画面のhtml を記述 ※②のSQL結果(書籍情報-		※ 画面設計書を	ご参考下さい	
2						
	検索(search) オブション機能	<u></u>	処理内容		備考	

No.	ファイル名	処理内容	備考
1	SearchController.php	① BookInfoモデルを使用できるようにする。	
		② 書籍一覧画面から検索ボタンを押下されて遷移してきた時以下の処理を行う。 ・書籍一覧機能からの入力データ(ISBN、Title、Price)を取得する。 例) \$isbn = \$request->isbn;	
		・取得出来た入力データを元に検索用のSQL文を作成し、bookinfoテーブルから データを取得する あいまい絞込み検索を実装する際はDBクラスを使用してSQL文を作成する。それぞれの入力データが空でない場合SQLな	
		■SQL作成 SQL例)\$sql = "SELECT * FROM bookinfo WHERE isbn Like '%\$isbn%' AND title Like '%\$title%' AND price Like '%\$price%' ORDER BY isbn";	3つのデータの入力が無い場合は 全検索と同じSQL文を発行する。
2	search. blade. php	③ 書籍一覧画面のhtml を記述する ※②のSQL結果 (書籍情報一覧) を表示する	※ 画面設計書をご参考下さい

		シス	テム名	書籍管理アプリケーション (Laravel版Ver1.0)	作月	10日	作成者	富田 茜
処理機能記述書				bmsweb10L	更新	斤日	更新者	神田ITスクール
								コンテンツ開発To
-ID			説明					
<u>−名</u> }								
<i>'</i> J								
-2								
書籍登	録(insert)							
No.	ファイル名			処理内容			備考	
1	InsertController.php	① Book Infoモラ	ルを使用できる。	ようにする。				
		@ o+		ulber 7				
		②・エラーの有 ■エラーカ	「無で行う処理を判 ぶ無い場合	刊断する		※ ボタン川聿築名	登録画面からの遷移	
				一スに登録する ※③以降の処理を行う。		ス パノンは自和立	立外四田がり ひだり	
				INTO bookinfo VALUE('{\$isbn}','{\$title}',\$price)";				
		■エラーた						
		• error	r.phpへ遷移する。					
			で記述した各録ボ	「タンの名前を取得し、値が設定されていれば登録ボタンからの遷移と !	判断出来る			
		Actions	て旧たした豆豆が	プランの目前を取付してにお 飲んですのです すいは 亜銀行 プラン りのだりです	THIMN U			
			をデータベース					
				ためのインスタンスを作成する。			A 6 1	
				ンスに代入する(ISBN, Title, Price)		※ エラー処理をご	ご参照下さい	
			ァータ をDDに登録 データを表示用に	する(ISBN, Title, Price) 一時保存する				
		. 77	/ ノで衣小用に	PT 本工 大工 T T T T T T T T T				
2	insert.blade.php	④ 書籍登録画面	iのhtmlを記述する	3				
				と を判断して表示を行う。				
		※続けて登録	を行う際には登録	録画面へリダイレクトを行う。		※ 画面設計書をこ	ご参考下さい	

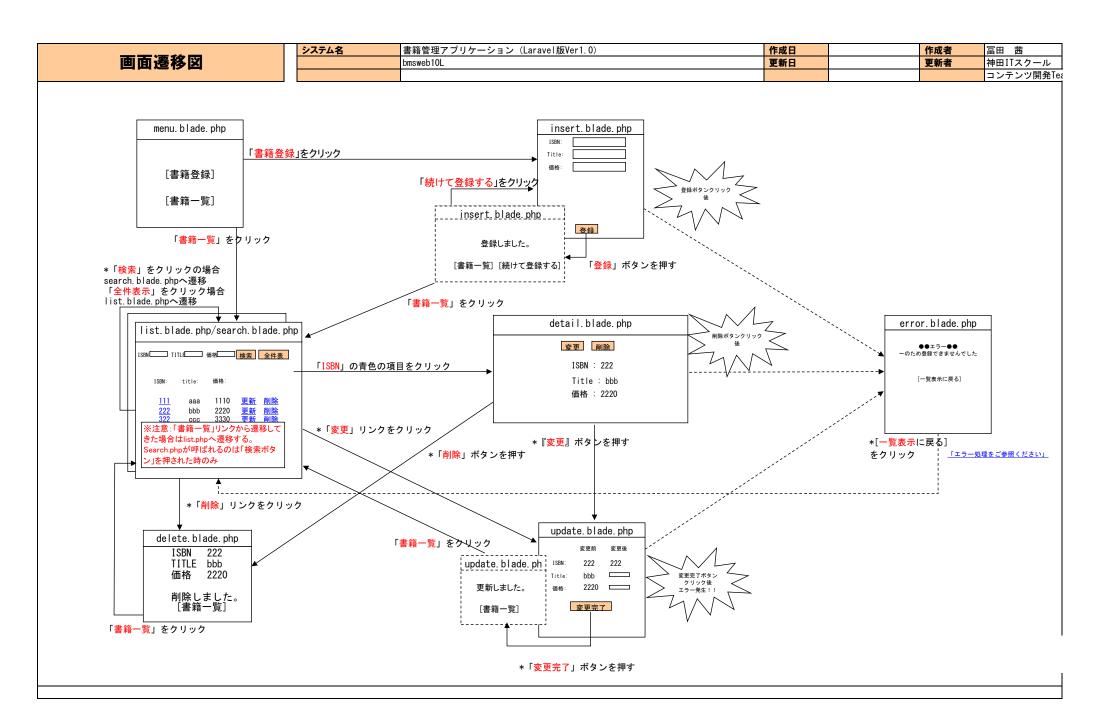
		システム名	書籍管理アプリケーション(Laravel版Ver1.0)	作成日	作成者	富田 茜
処理機能記述	述書		bmsweb10L	更新日	更新者	神田ITスクール
						コンテンツ開発Te
			説明			
籍更新(update)	<i>b</i> 7		加州中京		備考	
No. ファイル: 1 UpdateControll		Pook Infoエニッた体田で	処理内容 キストネニオス)佣石	
upuatecontroll	er prip	BookInfoモデルを使用で	さるようにする。			
	@ I	ICDN た To 但 ナス (DOCT 半)	信とGET送信の2パターン)			
	_	ISBNを取得する(FUST区) ■GET送信の場合	B C OLI 医信の2バターク)			
	'	■ULT区間の場合 1. 変更対象のisbnを	取得する			
		イ. 変更対象のTSbire 例)\$isbn = \$requ				
		1717 WIGOII WIGHT	300 / 10DIII			
		2. エラー回避のため	\$newBookを空文字で初期化する。			
			タがデータベースに存在するか確認する	※ エラー処理を	ご参照下さい	
		■データが存在しなか	った場合			
		・error.phpへ遷移				
		■データが存在した場				
		検索結果のTitleとPr	=			
		例) \$updateBook =	\$objBook->selectByIsbn(\$isbn);			
		• 元/II · 王かり在	-			
		4. 取得した更新対象				
		1列) \$updateBook =	<pre>\$updateBook[0]->toArray();</pre>			
		■POST送信の場合				
	"	■P031送信の場合 1. 変更対象のisbnを	取得する			
		1. 友文内外の1900で	4A IT 7 'V 0			
		2 更新対象ISRNデー	タがデータベースに存在するか確認する	※ エラー処理を	ご参照下さい	
		■データが存在した			ではなくhiddenで渡される	5値
		・error.phpへ遷		,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,		
		■データが存在した				
		検索結果のTitleと				
		例)\$isbn = \$requ	uest->isbn;			
		\$title = \$red	uest->title;			
		\$price = \$red	uest->price;			
		3. 取得した更新対象	データを配列化する。			
		4.更新データを\$new	Bookに登録する。			
		例)\$newBook = [A			
		'title' =>				
		'price' =>	\$price			
];				
		E 亜鉱ご カナDD/-	☆ ¹ → 7	₩ ICDNI+ 3 + I±	ではなくhiddonで流される	z 庙
		5. 更新データをDBに		※ 19時間は人力値	ではなくhiddenで渡される	の旧
		UPDATE BOOKINTO S	ET title='\$newTitle',price='\$newPrice' WHERE isbn ='\$isbn';			

⑤ 書籍更新画面のhtml を記述する ※初期画面と更新成功画面かを判断して表示を行う。

update. blade. php

※ 画面設計書をご参考下さい

		システム名 書籍管理アプリケーション (Laravel版Ver1.0)	作成日	作成者 冨田 茜	
如	理機能記述書	bmsweb10L	更新日	更新者 神田 ITスクー	
				コンテンツ開	
ID		説明			
名					
×					
	除(delete)				
No.	ファイル名	処理内容		備考	
1	DeleteController.php	<u>煙埋内谷</u> ① Book Infoモデルを使用できるようにする。		um -3	
'	borocooner or ror, prip	DOUNTING ETTINE KING CEROS TICT TO S			
		② 削除対象ISBNを取得する (POST送信とGET送信の2パターン)			
		③ 削除対象ISBNデータがデータベースに存在するか確認する			
		SQL例) SELECT * FROM bookinfo WHERE isbn = '\$isbn';			
		■データが存在しなかった場合	※ エラー処理を	ご参照下さい	
		・error.phpへ遷移する			
		■データが存在した場合			
		・該当ISBNをデータベースから削除する			
		SQL例)\$sql = "DELETE FROM bookinfo WHERE isbn = '{\$isbn}'";			
2	delete.blade.php	(4) 書籍削除画面のhtml を記述する	※ 画面設計書を	ご参考下なり	
2	delete.blade.prip	※③でデータベースを検索した時に取得できた書籍情報を表示する	☆ 画面政司音をこが行ってい		
		一人のです。人と技术した時に私情でとた自和情報と数かする	l		
書籍語	細(detail)				
No.	ファイル名	処理内容		備考	
1	DetailController.php	① BookInfoモデルを使用できるようにする。			
		② 詳細表示対象ISBNを取得する(GET送信)			
		② 子如末二人名 IODN O 字符柱切上 取得上 7			
		③ 詳細表示対象ISBNの書籍情報を取得する			
		SQL例) SELECT * FROM bookinfo WHERE isbn = '\$isbn';			
		■データが取得出来なかった場合			
		・error.phpへ遷移する			
		יוסווקויסווס אפריסווקויסווס			
2	detail.blade.php	- (4) 書籍詳細画面のhtml を記述する	※ 画面設計書を		
-		※③で取得したデータを表示する			
		# At-	·		
	表示(error) <mark>オプション</mark> 権			備考	
No.	ファイル名	処理内容		V用行	
1	error.blade.php	① エラーメッセージを取得する ② エラー画面のhtmlを記述する	※ 画面設計書を	プ 糸 孝下 オリ	
		② エラー画面の川川で記述する ※①で取得したエラーメッセージをメッセージの個数分繰り返し表示する。	本 四 回 政 前 音 を	C 2 7 1 C 6.	
		小少く私付したエフ アンピ フセアノピ フの間数力様う感し数かする。			



エラー処理詳細記述書

システム名	書籍管理アプリケーション(Laravel版Ver1.0)	作成日	作成者	冨田 茜
	bmsweb10L	更新日	更新者	神田ITスクール
				コンテンツ開発Tea

■エラー処理実装説明

- ・各処理内で入力値判定を行わないとDBに登録する時等に不正なデータで登録が出来てしまう。 それを防ぐ為に事前に必要なパラメータチェックを行い、エラー画面に遷移する必要があります。 エラー画面に遷移する理由はシステムを強化すると共に、どういう内容のエラーなのかをユーザー に示し正しく操作を行ってもらう意味も含みます。
 - 例) InsertController内
 - ・ISBNの空白チェック→空白だった場合はエラー画面へ遷移し、ユーザーにISBNが未入力である事を示す。

■error. blade. php画面遷移ケース一覧 ・エラー処理は実装有無に従って各処理ファイルに判定処理を追加してください。

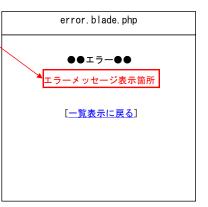
ファイル名	各ファイル判定エラーケース	エラー画面に表示するメッセージ	遷移先	実装有無
InsertController.php	登録ボタン押下時、ISBNが未入力	ISBNが未入力の為、書籍登録処理は行えませんでした。	list.blade.php	必須
	登録ボタン押下時、入力ISBNが重複	入力ISBNは既に登録済みの為、書籍登録処理は行えませんでした。	list.blade.php	必須
	登録ボタン押下時、Titleが未入力	タイトルが未入力の為、書籍登録処理は行えませんでした。	list.blade.php	必須
	登録ボタン押下時、価格が未入力	価格が未入力の為、書籍登録処理は行えませんでした。	list.blade.php	必須
	登録ボタン押下時、価格が数値以外	価格の値が不正の為、書籍登録処理は行えませんでした。	list.blade.php	必須
DetailController.php	画面遷移時に詳細表示対象データが存在しない	詳細対象の書籍が存在しない為、詳細情報処理は行えません。	list.blade.php	必須
UpdateController.php	変更完了ボタン押下時、Titleが未入力	タイトルが未入力の為、書籍更新処理は行えませんでした。	list.blade.php	必須
	変更完了ボタン押下時、価格が未入力	価格が未入力の為、書籍更新処理は行えませんでした。	list.blade.php	必須
	変更完了ボタン押下時、価格が数値以外	価格の値が不正の為、書籍更新処理は行えませんでした。	list.blade.php	必須
	変更対象のデータが存在しない	更新対象の書籍は存在しない為、更新処理は行なえませんでした。	list.blade.php	必須
DeleteController.php	削除ボタン押下時、削除対象のデータが存在しない	削除対象の書籍は存在しない為、削除処理は行なえませんでした。	list.blade.php	必須

■補足説明

・エラー情報を取得する方法

{{\$error}}

繰り返しを使い表示してください。



	システム名	書籍管理アプリケーション(Laravel版Ver1.0)	作成日	作成者	冨田 茜
テスト実施		bmsweb10L	更新日	更新者	神田ITスクール
					コンテンツ開発Team

■テスト実施方法について

- ・完成したアプリケーションは、仕様通り動くかテストを行います。
- ・テストを実施する際にはテストケースと呼ばれるテスト内容をもとにテストを行い、 正しく動くことを証明するために、エビデンスと呼ばれる資料を作成します。

■テストケース

- ・書籍管理アプリケーション(Web版Ver1.0)の「StepByStep」ページで取得したものをコピーして利用してください。
- ・エビデンス取得作業は不要です。
- ・記入が完了したテストケースファイルは、提出用フォルダに格納して提出してください。

			_	
- 17	ш	=,,,	-	書
	-	= =		_

システム名	書籍管理アプリケーション (Laravel版Ver1.0)	作成日	作成者	冨田 茜
	bmsweb10L	更新日	更新者	神田ITスクール
				コンテンツ開発Team

機能フローID	
機能フロー名	
フロー区分	

説明

機能フロー図

MySQLデータベース使用を前提とします。

ユーザー名 root パスワード root123

データベース名 mybookdb

テーブル名 bookinfo

7 7 70 1	DOURTHIO			
列名	データ型	列制約	説明	
isbn	VARCHAR (20)	PRIMARY KEY	主キー制約、図書のISBN	
title	VARCHAR (100)	無し	図書のタイトル	
price	INTEGER	無し	図書の価格	

テーブル作成のSQL文

CREATE TABLE bookinfo (
isbn VARCHAR(20) PRIMARY KEY,
title VARCHAR(100),
price INTEGER
)ENGINE=InnoDB,DEFAULT CHARSET=utf8;

サンプルデータ登録用のSQL文

INSERT INTO bookinfo VALUES('0001','java',1001); INSERT INTO bookinfo VALUES('0002','c++',1002); INSERT INTO bookinfo VALUES('0003,'ruby',1003); INSERT INTO bookinfo VALUES('0004','perl',1004); INSERT INTO bookinfo VALUES('0005','database',1005);