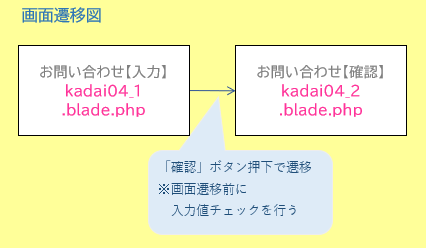
第４回　リクエスト・バリデーション

第4回目ではフォームに入力された値をPOST形式で送信します。また、その際にリクエストクラスを使用し、バリデーションチェックを行います。

■画面遷移図



課題４－１　入力画面

URL「/kadai04」にGETリクエストがきたら、Kadai04Controllerにて、お問い合わせ入力画面(kadai04\_1.blade.php)を呼び出す処理を作成してください。

作成手順、仕様詳細は下記です。

1. **コントローラの作成**
2. **ルーティングの設定**
3. **ビューの作成**

　■Kadai04Controller.phpお問い合わせ入力画面メソッド仕様

|  |  |
| --- | --- |
| アクセス修飾子 | public |
| メソッド名 | index |
| 引数 | なし |
| 処理 | viewヘルパー関数を使用し、kadai04\_1.blade.phpを　呼び出す。 |

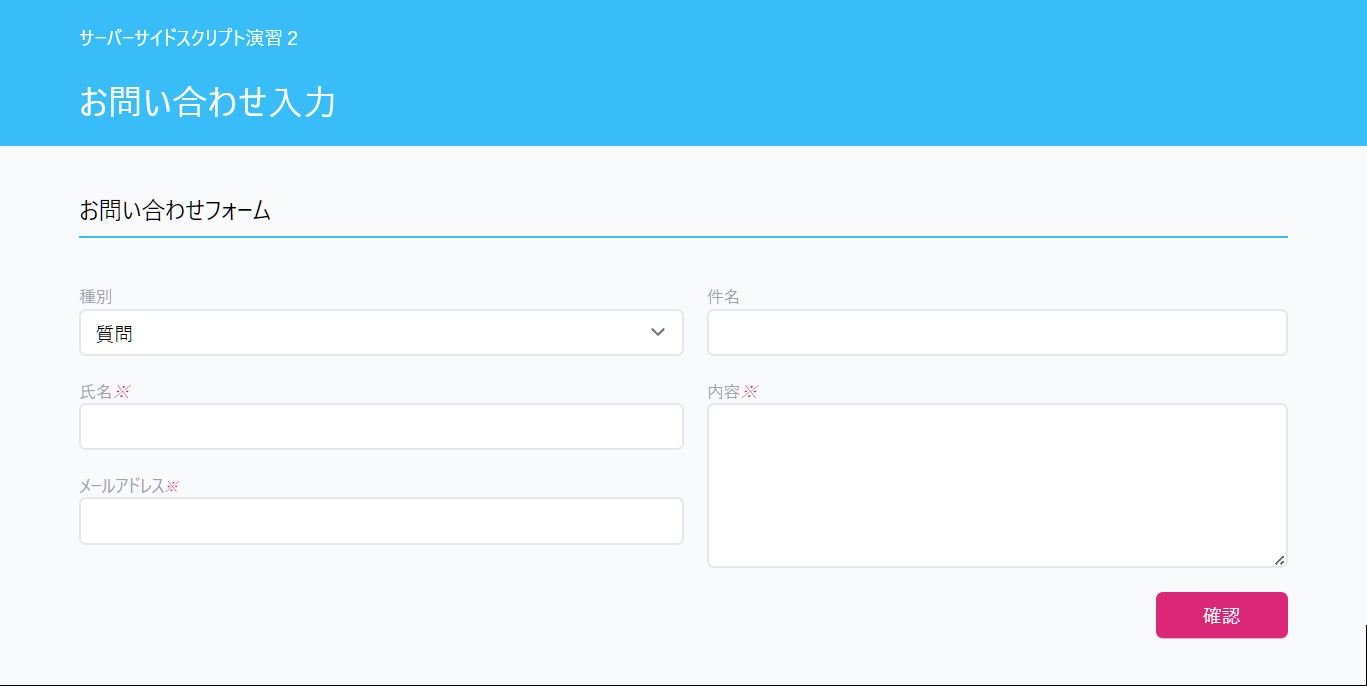
　■ルーティング設定

|  |  |
| --- | --- |
| リスエスト形式 | Get |
| URI | http://localhost:8000**/kadai04** |
| 呼び出しクラス・メソッド | Kadai04Controller の　index　メソッド |

　■kadai04\_1.blade.php　の仕様

|  |  |
| --- | --- |
| テンプレート | resources\views\layouts\kadai.blade.php |
| pageTitle | kadai04\_1 |
| title | お問い合わせ入力 |
| content | ※配布した「kadai04\_1.txt」の内容をコピー＆ペースト。  ・ボタン押下時のフォームデータ送信は課題４－２で行う。 |

　■kadai04実行結果 入力画面表示



課題４－２　確認画面

URL「/kadai04」にPOSTリクエストがきたら、Kadai04Controllerにて、お問い合わせ確認画面 (kadai04\_2.blade.php)を呼び出す処理を作成してください。

ただし、画面遷移前にリクエストデータのチェックを行うこと。

作成手順、仕様詳細は下記です。

1. **リクエストクラスの作成、コントローラの作成（メソッド追加）**
2. **ルーティングの設定**
3. **ビューの作成**

　■Kadai04Request.php仕様 【メソッドは3件】

※クラスファイルは**コマンドで作成**すること。

　　コマンドで作成した場合、以下の一部のメソッドは定義されているため、処理を仕様通りに変更すること。定義の無いメソッドについては追記すること。

|  |  |
| --- | --- |
| アクセス修飾子 | public |
| メソッド名 | **authorize** |
| 引数 | なし |
| 戻り値 | bool |
| 処理 | パスが「kadai04」であるとき、trueを返す。  そうでなければfalseを返す。 |

|  |  |
| --- | --- |
| アクセス修飾子 | public |
| メソッド名 | **rules** |
| 引数 | なし |
| 戻り値 | array |
| 処理 | 下記のルールを追加。それぞれのリクエストデータの項目名は、入力画面のHTMLにて確認すること。  ■「名前」　必須  ■「メール」　必須、　メールアドレス形式であること  ■「内容」　必須 |

|  |  |
| --- | --- |
| アクセス修飾子 | public |
| メソッド名 | **messages** |
| 引数 | なし |
| 処理 | 下記のメッセージを配列形式で返す。  ■名前.必須エラー  '名前が入力されていません',  ■メール．必須エラー  'メールアドレスが入力されていません',  ■メール．形式エラー  'メールアドレスの形式が間違っています',  ■内容．必須エラー  '内容が入力されていません' |
|  |  |

　■Kadai04Controller.phpお問い合わせ確認画面メソッド仕様

　　※コントローラクラスは作成済みなので、メソッドを追加する。

|  |  |
| --- | --- |
| アクセス修飾子 | public |
| メソッド名 | post |
| 引数 | Kadai04Request $request |
| 処理 | 1. 引数のリクエストデータを変数に格納 2. トークンを破棄 3. viewヘルパー関数を使用し、変数の値を渡し、   kadai04\_2.blade.php を　呼び出す。 |

　■ルーティング設定

|  |  |
| --- | --- |
| リスエスト形式 | post |
| URI | http://localhost:8000**/kadai04** |
| 呼び出しクラス・メソッド | Kadai04Controller の　post　メソッド |

　■kadai04\_2.blade.php　の仕様

|  |  |
| --- | --- |
| テンプレート | resources\views\layouts\kadai.blade.php |
| pageTitle | kadai04\_2 |
| title | お問い合わせ確認画面 |
| content | ※配布した「kadai04\_2.blade.txt」の内容をコピー＆ペースト  以下の処理を記載。  ・入力画面での入力データ表示  ・「戻る」ボタン（リンク）押下　→　お問い合わせ入力画面へ遷移。 |

　■kadai04\_1.blade.php　（課題４－１　お問い合わせ入力画面）の追加仕様

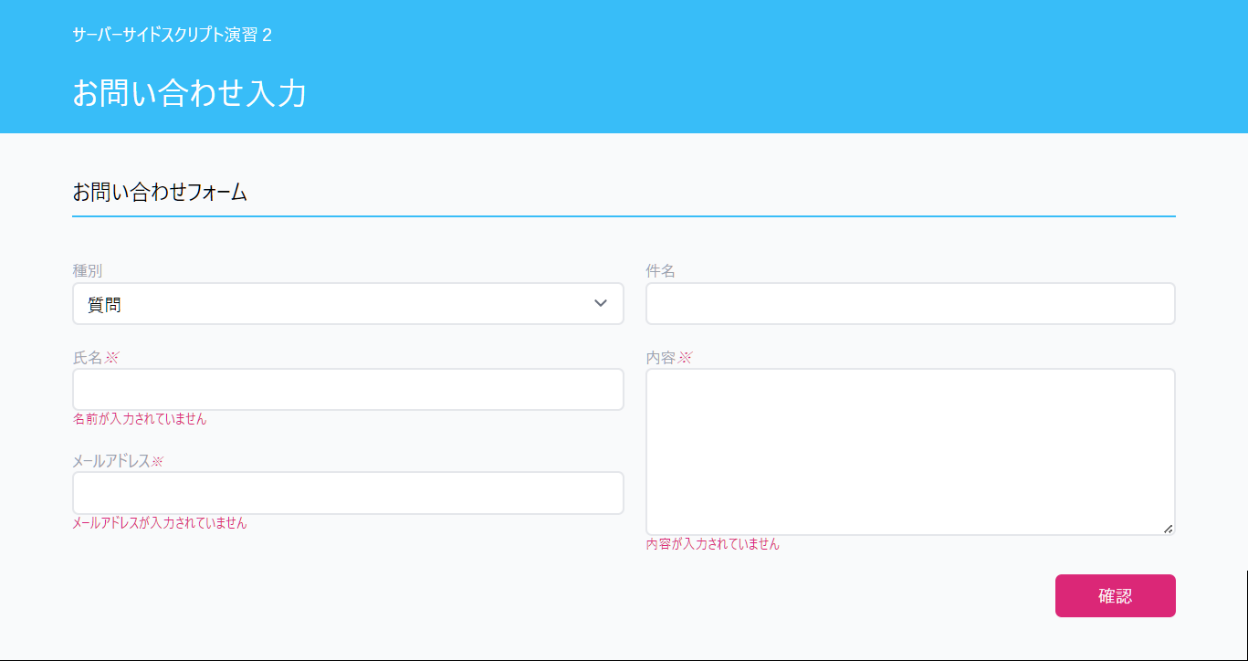
|  |  |
| --- | --- |
| content | ・「確認」ボタン押下　→　POST形式でデータを送信。送信先は「/kadai04」  ・入力値にエラーがあった場合、エラーがあった項目の下部にメッセージが表示されること。（実行結果参照） |

（実行結果は次ページ）

　■kadai0４実行結果

※必ず、**「お問い合わせ入力画面」の「確認」ボタンをクリックすることで、「お問い合わせ確認画面（この画面）」へ遷移**すること。URLからダイレクトに確認画面を表示できないことも確認する。

　　★**【パターン１】　入力項目に不備があった場合。入力画面のまま、エラーメッセージが表示される。**



　　★**【パターン２】　入力項目OKの場合、確認画面にデータが表示される。**



課題４　プラスアルファ課題

入力→確認→**完了画面（kadai04\_3.blade.php）**を追加**。**

**※ただし、下記に留意すること。（ただの画面遷移ではNG）**

1. **確認画面で表示したデータは、確認画面で送信ボタンが押された時に利用できるようにすること。**

**たとえば、今後、DB登録やメール送信などの処理を実装しようとした場合、データ利用可能であること。**

1. **ルーティングの設定やコントローラのメソッド追加など、設計を行うこと。**
2. **確認画面から入力画面に戻ったとき、入力画面に入力した値を保持できていること。**

**【ヒント】　講義していないものも含めて、下記を調べて設計→実装を行う。**

**・ルーティングとメソッドの処理の順序を整理する。**

**・セッション**

**・リダイレクト**

**・route関数**