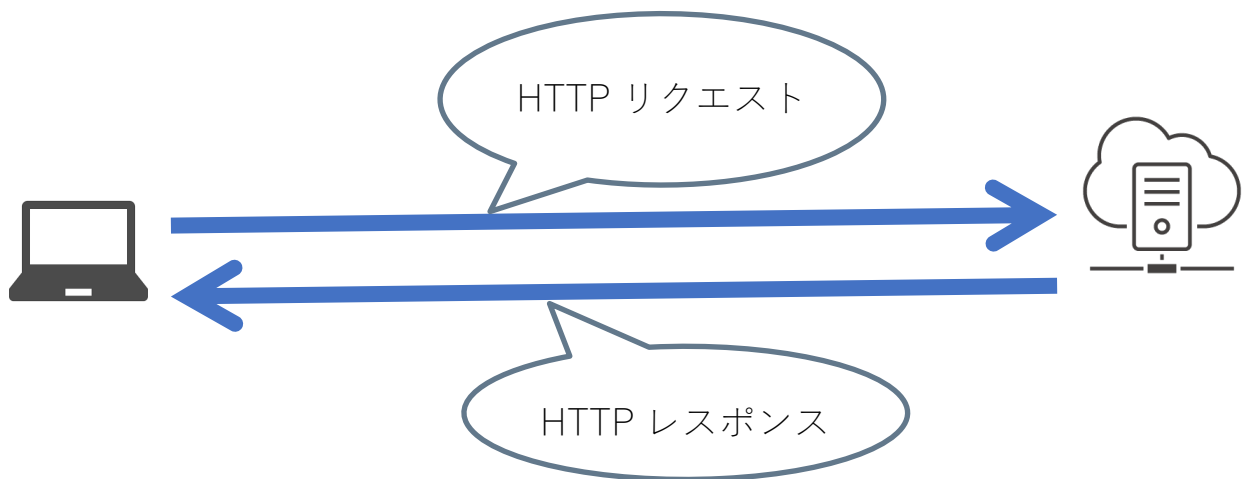


PHP 練習問題. 06 セッション

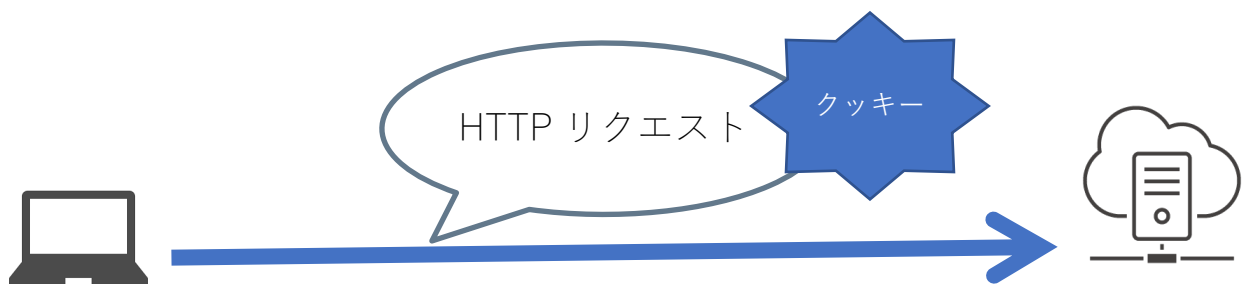
「セッション (`$_SESSION`)」とは、端的に言うと、「**サーバー側で保存できる連想配列の変数**」のことです。

PHP の変数は、ページ間で値を保持して使うことができません。
なぜなら、1 回の通信で、サーバーはクライアントのことを「忘れてしまう（覚えていない）」からです。



そこで、PHP の変数を他のページでも使いたい場合は「**クッキー**」という仕組みを使います。PHP では「`$_COOKIE`」という「**連想配列の変数**」で、設定、取得できます。

クッキーは**テキストファイル**で、ブラウザの中に保存されます。PHP で保存された変数の内容が「**キー=>値**」という形で保存されます。サーバーに対して HTTP リクエストを送信するときに、一緒にサーバーへ送信されます。



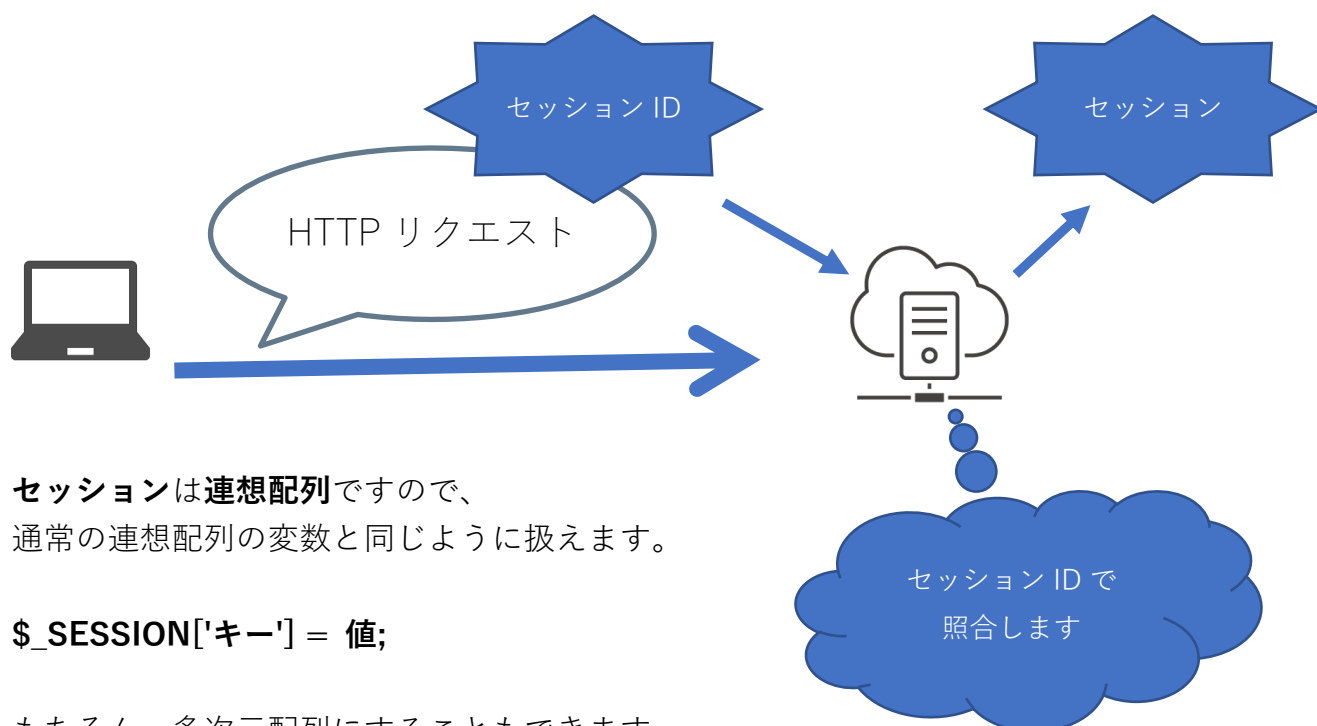
クッキーは**ブラウザの中に保存される**ため、ブラウザの不具合（脆弱性）によって、悪意を持った第三者（攻撃者）によって「**盗み見られる**」という事例が多発しました。例えば、ログインの情報をクッキーに保存しておくことで、その情報が盗まれたとき、他者がログインできてしまうことになってしまいます。

そこで**セッション**という仕組みが考えられました。

セッションも実は**クッキー**の仕組みを使っています。しかし、**クッキー**に保存されるのは、PHP が作った「**一意の（重複しない）ID**」のみです。

セッションに保存された値は、**サーバー上のテキストファイル**に保存されています（設定でデータベースに保存することも可能です）。**セッション**のファイル名は**クッキー**に保存されている ID と同じ名前になっています。

クッキーに保存されている ID と一致するサーバー上のテキストファイルを照らし合わせることで、別のページ間で同じ変数の値を使用することができるわけです。



セッションは**連想配列**ですので、通常の連想配列の変数と同じように扱えます。

```
$_SESSION['キー'] = 値;
```

もちろん、多次元配列にすることもできます。

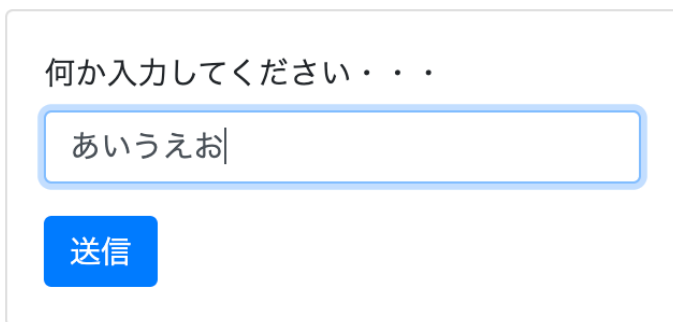
```
$_SESSION['キー'] = ['a' => 'あ', 'b' => 'い'];
```

設問1. index.php に form でテキストボックスを一つ作成してください。

action.php で送信された値を表示し、同じ値をセッションに保存します。
action.php には index.php に戻るリンクを作成します。index.php に戻ったときに、テキストボックスに入力した値が消えずに入力されているようにしてください。

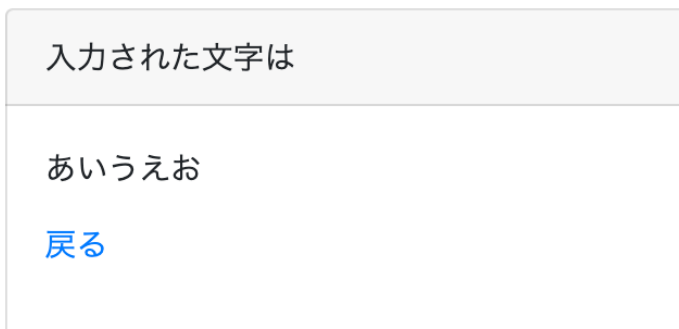
(表示例)

- index.php



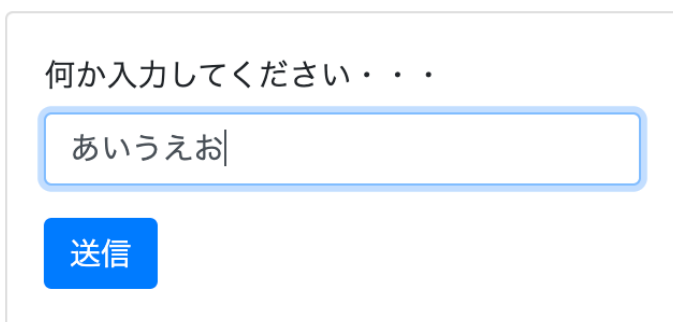
A screenshot of a web form on index.php. It features a light gray rectangular container. Inside, at the top, is the text '何か入力してください...' in black. Below this is a text input field with a light blue border, containing the text 'あいうえお'. At the bottom of the container is a blue rectangular button with the white text '送信'.

- action.php



A screenshot of the action.php page. It has a light gray rectangular container. The top section has a gray header with the text '入力された文字は'. Below the header, the text 'あいうえお' is displayed. At the bottom of the container is a blue text link that says '戻る'.

- index.php (「戻る」リンクをクリックして index.php に戻ったとき)



A screenshot of index.php after clicking the '戻る' link. It shows the same form as the first screenshot: a light gray container with the text '何か入力してください...', a text input field containing 'あいうえお', and a blue '送信' button.

設問2. セッションを使って、クリックするたびにカウンタの値が1ずつ増えていくようにしてください。リセットボタンをクリックすると0に戻るようにします。

(表示例)

- index.php

セッションを使ったカウンタ	
0	
カウント	リセット

- 「カウント」をクリックすると、数字が1ずつ増えます。

セッションを使ったカウンタ	
7	
カウント	リセット

- 「リセット」をクリックすると、数字が0に戻ります。

セッションを使ったカウンタ	
0	
カウント	リセット

(ヒント)

jQuery を使うと、form を使わなくても実現できます。

(例)

```
<button id="count">カウント</button>
<button id="reset">リセット</button>

<script src="https://code.jquery.com/jquery-3.4.1.min.js"></script>
<script>
  $('#count').click(function() {
    location.href = './';
  });

  $('#reset').click(function() {
    location.href = './?reset';
  });
</script>
```

設問3. index.php で、下記の「商品のリスト」の連想配列を作成してください。

```
0 =>  
  product_name => みかん  
  price => 300  
1 =>  
  product_name => りんご  
  price => 500  
2 =>  
  product_name => バナナ  
  price => 150
```

1. 上記の配列の内容と注文数を index.php で一覧表示できるようにしてください。
2. 「注文数」を入力するテキストボックスと、「カートに入れる」ボタンを form を使って作成してください。form の action 属性には「./cart_add.php」を指定してください。
3. a タグを使って、「カートを見る」というリンクを作ってください。href 属性には「./cart_show.php」を指定してください。

(表示例)

● index.php

商品名	価格	注文数	
みかん	300円	<input type="text"/>	<button>カートに入れる</button>
りんご	500円	<input type="text"/>	<button>カートに入れる</button>
バナナ	150円	<input type="text"/>	<button>カートに入れる</button>

[カートを見る](#)

(ヒント)

商品ごとに<form></form>を作ります。

設問4. (設問 3.に引き続き)

1. index.php で「カートに入れる」 ボタンをクリックすると
cart_add.php に POST され、index.php から送信された商品名、価格、個数をセッションに保存します。処理が終わると、index.php にリダイレクトします。
2. index.php の「カートを見る」リンクをクリックすると、
cart_show.php に遷移します。
3. cart_show.php
 - ① セッションに保存されている商品と価格、注文数、小計金額の一覧、合計金額を表示してください。
 - ② form で「削除」 ボタンを作ってください。form の action 属性には「./cart_del.php」を設定してください。
 - ③ a タグを使って「カートのを空にする」リンクを作成してください。href 属性には「./cart_del_all.php」を指定してください。

(表示例)

- index.php 「注文数」に数字を入力して、「カートに入れる」をクリックします。

商品名	価格	注文数	
みかん	300円	<input type="text" value="3"/>	<button>カートに入れる</button>
りんご	500円	<input type="text"/>	<button>カートに入れる</button>
バナナ	150円	<input type="text"/>	<button>カートに入れる</button>

[カートを見る](#)

- cart_add.php で、index.php から POST されたデータをセッションに保存し、index.php にリダイレクトします。

商品名	価格	注文数	
みかん	300円	<input type="text"/>	カートに入れる
りんご	500円	<input type="text"/>	カートに入れる
バナナ	150円	<input type="text"/>	カートに入れる

[カートを見る](#)

- index.php の「カートを見る」のリンクをクリックしたら cart_show.php が表示されます。セッションに保存されているデータが一覧表示されます。

商品名	価格	注文数	小計	
みかん	300円	3	900円	削除

合計金額：900円

[カートを空にする](#)

[戻る](#)

- ※ カートは、商品ごとに注文数を合算する必要はありません。
 同じ商品を複数回カートに入れたときは、複数行になっても構いません。

商品名	価格	注文数	小計	
みかん	300円	1	300円	削除
みかん	300円	3	900円	削除

設問5. （設問 4.に引き続き）cart_show.php で、「削除」 ボタンをクリックしたときに、カートに入っている該当の商品を削除できるようにします。
POST 先の cart_del.php で該当のセッションを削除します。削除完了後は cart_show.php にリダイレクトします。

（表示例）

- cart_show.php 「りんご」 の削除ボタンをクリックしてみます。

商品名	価格	注文数	小計	
みかん	300円	3	900円	削除
りんご	500円	2	1000円	削除
バナナ	150円	4	600円	削除
合計金額：2500円				
カートを空にする				
戻る				

- cart_del.php に POST され、「りんご」 のデータをセッションから削除して、cart_show.php にリダイレクトします。
- cart_show.php が表示されます。

商品名	価格	注文数	小計	
みかん	300円	3	900円	削除
バナナ	150円	4	600円	削除
合計金額：1500円				
カートを空にする				
戻る				

設問6. (設問 5.に引き続き) cart_show.php で、「カートに空にする」リンクをクリックしたときに、カートに入っている全ての商品を削除できるようにします。

cart_show.php では、カートが空のときは、「カートは空です。」と表示するようにしてください。

(表示例)

- cart_show.php

商品名	価格	注文数	小計	
みかん	300円	3	900円	削除
バナナ	150円	4	600円	削除
合計金額：1500円				
カートに空にする				
戻る				

- cart_del_all.php で、セッションに保存されている全ての商品を削除して、cart_show.php へリダイレクトします

- カートが空になったときの cart_show.php の表示です。

カートは空です。
[戻る](#)