

Correction des Activités 1- 2- 3 -4 (chapitre 3)

1. Créez un nouveau projet Laravel

```
composer create-project laravel/laravel nom_projet
```

2. Lancer le serveur web et accéder à la page d'accueil de notre installation LARAVEL

```
php artisan serve
```

-----Activité 1-----

3. Créez et testez une route "/start" nommée "start" qui permet d'afficher le texte "Hello, i am your first route"

Résultat d'affichage: " Hello my name, i am your first route"

```
Route::get('start',function(){  
    echo "<h3>Hello, i am your first route</h3>";  
})->name('start');
```

4. Créez une vue nommé "start.blade.php" qui affiche le même message que la question précédente.
5. Redéfinissez la route (définie en question 3) pour qu'elle retourne la vue "start" comme réponse Http.

```
Route::get('start',function(){  
    return view('start');  
})->name('start');
```

6. Listez les routes

```
php artisan route:list
```

-----Activité 2-----

7. Créez un Contrôleur TestController:

```
php artisan make:controller TestController
```

8. Utilisez "TestController", la route "start" et la vue "start.blade.php" pour afficher le message "Hello, i am your first route", pour y faire, procédez comme suit:
 - dans le fichier web.php, écrivez :

```
use App\Http\Controllers\TestController;  
Route::get('start', [TestController::class, 'index'])->name('start');
```

- dans le contrôleur TestController.php, ajoutez la fonction index qui retourne la vue start comme réponse http

```
public function index(){  
    return view('start');    }
```

-----Activité 3-----

9. Modifiez "TestController", la route "start" et la vue "start.blade.php" pour afficher le message "Hello ****, your age is **** years"

- **dans le fichier web.php:**

- l'âge ici peut être inférieur à 10 car on a mis « ? » (0..1) après le deuxième chiffre.
- Le « name » est alphabétique et obligatoire car on a mis « + » (1..*), on met « * » (0..*) ça devient un paramètre optionnel

```
Route::get('start/{name}/{age}', [TestController::class, 'index'])
    ->where(['name'=>'[a-zA-Z]+', 'age'=>'[1-9][0-9]?'])
    ->name('start');
```

- **dans TestController, modifiez la méthode index:**

- si l'un des paramètres est optionnel il faut lui donner une valeur par défaut dans la méthode index, par exemple index(\$name, \$age=null), \$age ici est optionnel.

```
public function index($name, $age){
    return view('start', compact('name', 'age'));
}
```

- **dans la vue start.blade.php**

```
<h3>Hello {{$name}}, your age is {{$age}} years</h3>
```

-----Activité 4-----

1. Créez la vue info.blade.php, qui affiche un formulaire avec deux zones de texte nom, prenom, age, sexe (zone de sélection simple), et loisirs (liste de checkbox)

```
<form action="{{ route('users.store') }}" method="POST" target="_blank">
    @csrf
    <label for="nom">Entrez votre nom : </label>
    <input type="text" name="nom" id="nom"><br>
    <label for="prénom">Entrez votre prénom : </label>
    <input type="text" name="prenom" id="prénom"><br>
    <label for="age">Entrez votre age : </label>
    <input type="decimal" name="age" id="age"><br>
    <label for="sexe">Choisir votre sexe : </label>
    <select name="sexe" id="sexe">
        <option value="Homme">Homme</option>
        <option value="Femme">Femme</option>
    </select><br>
```

```

<label for="loisir">Quels sont vos loisirs? </label><br>
<input type="checkbox" name="loisirs[]" value="Music">Music<br>
<input type="checkbox" name="loisirs[]" value="Sport">Sport<br>
<input type="checkbox" name="loisirs[]" value="Lecture">Lecture<br>
<input type="submit" value="Envoyer !">

</form>

```

2. Associez le formulaire à un contrôleur "TestFormController", qui définit deux actions:

- Create(): permet d'afficher le formulaire
- Store(): permet de récupérer les valeurs saisies dans le formulaire et de les afficher dans la vue "afficheInfo.blade.php"



Création du contrôleur :

Php artisan make :controller TestFormController



Définition des méthodes :

```

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

class TestFormController extends Controller
{
    public function create(){
        return view("infos");
    }
    public function store(Request $req){
        return view("afficheInfo", compact("req"));
    }
}

```



Définition de la page afficheInfo.blade.php

```

<body>
    <h1> Information soumises au serveur </h1>
    <p>Nom: {{$req->nom}}</p>
    <p>Prénom: {{$req->prenom}}</p>
    <p>Nom: {{$req->age}}</p>
    <p>Nom: {{$req->sexe}}</p>
    <ul>
        @foreach($req->loisirs as $loisir)
            <li>{{$loisir}}</li>
        @endforeach
    </ul>
</body>

```

ce syntaxe n'est pas inclus dans l'examen

3. Définissez les routes nécessaires à l'affichage et la soumission du formulaire

```
use App\Http\Controllers\TestFormController;
Route::get('users/info', [TestFormController::class, 'create'])
    ->name('users.create');
Route::post('users/info', [TestFormController::class, 'store'])
    ->name('users.store');
```