

## TP4 : Thymeleaf

Créer le package :

```
package com.example.appweb.entity;
```

Créer la classe Client.java

```
package com.example.appweb.entity;

public class Client {

    private Long id;

    private String name;

    private Long age;

    private String location;

    public Long getId() {
        return id;
    }

    public void setId(Long id) {
        this.id = id;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public Long getAge() {
        return age;
    }

    public void setAge(Long age) {
        this.age = age;
    }

    public String getLocation() {
        return location;
    }
}
```

```

        public void setLocation(String location) {
            this.location = location;
        }
    }
}

```

Créer le package :

```
package com.example.appweb.service;
```

Créer la classe ServiceClient.java

```

package com.example.appweb.service;

import java.util.ArrayList;
import java.util.List;

import org.springframework.stereotype.Service;

import com.example.appweb.entity.Client;

@Service
public class ServiceClient {

    private static List<Client> list = new ArrayList<Client>();
    static {
        Client c = new Client();
        c.setId(1L);
        c.setName("Client 1");
        c.setAge(28L);
        c.setLocation("Tunisia");
        list.add(c);

        c = new Client();
        c.setId(2L);
        c.setName("Client2");
        c.setAge(30L);
        c.setLocation("Algeria");
        list.add(c);

        c = new Client();
        c.setId(3L);
        c.setName("Client 3");
        c.setAge(31L);
        c.setLocation("Maroc");
        list.add(c);
    }
}

```

```

    public List<Client> getClientList() {
        return list;
    }

    public Client findClientById(long id) {
        Client client = new Client();
        for(Client cl: list)
        {
            if(cl.getId() == id)
            {
                client = cl;
            }
        }
        return client ;
    }

    public void saveClient(Client Cl)
    {
        Client c = new Client();
        c.setId(Long.valueOf(list.size() + 1));
        c.setName(Cl.getName());
        c.setAge(Cl.getAge());
        c.setLocation(Cl.getLocation());
        list.add(c);
    }

    public void deleteClientById(long id)
    {
        System.out.println("id = " + id);
        for(Client cl: list)
        {
            if(cl.getId() == id)
            {
                list.remove(cl);
            }
        }
    }
}

```

Créer le package :

```
package com.example.appweb.controller;
```

Créer la classe ClientController.java

```

package com.example.appweb.controller;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.ModelAttribute;
import org.springframework.web.bind.annotation.PathVariable;
import org.springframework.web.bind.annotation.PostMapping;

import com.example.appweb.entity.Client;
import com.example.appweb.service.ServiceClient;

@Controller
public class ClientController {

    @Autowired
    ServiceClient SCL;

    @GetMapping("/")
    public String viewHomePage(Model model) {
        model.addAttribute("allemplist", SCL.getClientList());
        return "index";
    }

    @GetMapping("/addnew")
    public String addNewClient(Model model) {
        Client client = new Client();
        model.addAttribute("client", client);
        return "newclient";
    }

    @PostMapping("/save")
    public String saveEmployee(@ModelAttribute("employee") Client
client) {
        SCL.saveClient(client);
        return "redirect:/";
    }

    @GetMapping("/showFormForUpdate/{id}")
    public String updateForm(@PathVariable(value = "id") long id,
Model model) {
        Client client = SCL.findClientById(id);
        model.addAttribute("client", client);
        return "update";
    }

    @GetMapping("/deleteClient/{id}")

```

```

        public String deleteThroughId(@PathVariable(value = "id") long
id) {
            SCL.deleteClientById(id);
            return "redirect:/";

        }

    }
}

```

Ajouter dans le dossier : **src/main/ressources/templates**, les pages suivantes :

**index.html** ;

```

<!DOCTYPE html>
<html lang="en" xmlns:th="http://www.thymeleaf.org">
<head>
<meta charset="ISO-8859-1">
<title>Client</title>
<link rel="stylesheet"

href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstr
ap.min.css"
        integrity="sha384-
MCw98/SFnGE8fJT3GXwEOngsV7Zt27NXFoaoApmYm81iuXoPkFOJwJ8ERdknLPMO"
        crossorigin="anonymous">
</head>
<body>
<div class="container my-2" align="center">

<h3>Client List</h3>
<a th:href="@{/addnew}" class="btn btn-primary btn-sm mb-3" >Add
Client</a>
    <table style="width:80%" border="1"
        class = "table table-striped table-responsive-md">
        <thead>
        <tr>
            <th>Name</th>
            <th>Age</th>
            <th>Location</th>
            <th>Action</th>
        </tr>
        </thead>
        <tbody>
        <tr th:each="client:${allempelist}">
            <td th:text="${client.name}"></td>
            <td th:text="${client.age}"></td>
            <td th:text="${client.location}"></td>
            <td> <a
th:href="@{/showFormForUpdate/{id}}(id=${client.id})"
                class="btn btn-primary">Update</a>

```

```

        <a th:href="@{/deleteClient/{id}(id=${client.id})}"
        class="btn btn-danger">Delete</a>
    </td>
</tr>
</tbody>
</table>
</div>
</body>
</html>

```

***newclient.html ;***

```

<!DOCTYPE html>
<html lang="en" xmlns:th="http://www.thymeleaf.org">
<head>
<meta charset="ISO-8859-1">
<title>Client Management System</title>
<link rel="stylesheet"

href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstr
ap.min.css"
    integrity="sha384-
MCw98/SFnGE8fJT3GXwEOngsV7Zt27NXFoaoApmYm81iuXoPkFOJwJ8ERdknLPMO"
    crossorigin="anonymous">
</head>
<body>
    <div class="container">
        <h1>Client Management System</h1>
        <hr>
        <h2>Save Client</h2>

        <form action="#" th:action="@{/save}" th:object="${client}"
        method="POST">
            <input type="text" th:field="*{name}"
placeholder="Client Name"
                class="form-control mb-4 col-4"> <input type="text"
                th:field="*{age}" placeholder="Employee age"
                class="form-control mb-4 col-4">
                <input type="text"
                th:field="*{location}" placeholder="Employee
location"
                class="form-control mb-4 col-4">

                <button type="submit" class="btn btn-info col-2">Save
                Employee</button>
            </form>

            <hr>

```

```

        <a th:href="@{/}"> Back to Client List</a>
    </div>
</body>
</html>

```

### *Update.html*

```

<!DOCTYPE html>
<html lang="en" xmlns:th="http://www.thymeleaf.org">
<head>
<meta charset="ISO-8859-1">
<title>Client Management System</title>

<link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstr
ap.min.css">
</head>
<body>
    <div class="container">
        <h1>Client Management System</h1>
        <hr>
        <h2>Update Client</h2>

        <form action="#" th:action="@{/save}" th:object="${client}"
            method="POST">

            <!-- Add hidden form field to handle update -->
            <input type="hidden" th:field="*{id}" />

            <input type="text" th:field="*{Name}" class="form-
control mb-4 col-4">

                <input type="text" th:field="*{age}" class="form-
control mb-4 col-4">

                <input type="text" th:field="*{location}"
class="form-control mb-4 col-4">

                <button type="submit" class="btn btn-info col-2">
Update Client</button>
        </form>

        <hr>

        <a th:href = "@{/}"> Back to Client List</a>
    </div>
</body>
</html>

```