

**Q1. Create an external JS file and use it to show an alert on click of a button.**

**Solution**

**Alert.js**

```
function displayAlert()
{
    alert("Hello from external Js");
}
```

**Form.html**

```
<!DOCTYPE html>
<html lang="en" xmlns:th="http://www.thymeleaf.org">
<head>
    <meta charset="UTF-8">
    <title>Alert</title>
    <script th:src="@{/js/alert.js}"></script>
</head>
<body>
<button onclick="displayAlert()">Click to see alert!</button>
</body>
</html>
```

**ApplicationController.java**

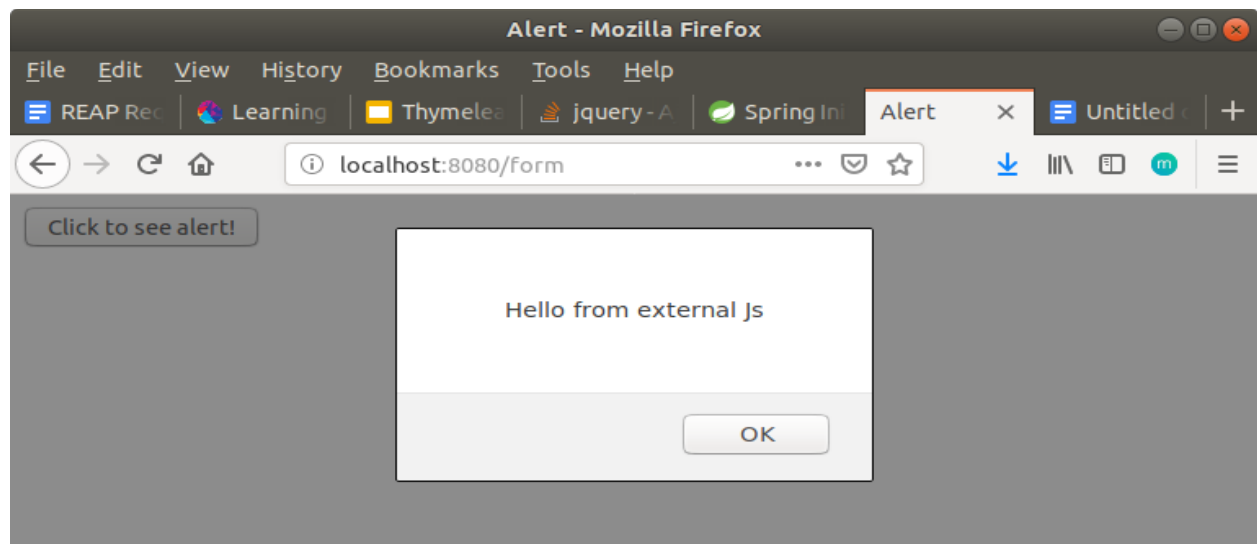
```
package com.demo.thymeleaf.controller;

import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.GetMapping;

@Controller
public class ApplicationController {

    //Question1
    @GetMapping("/form")
    public String form()
    {
        return "form";
    }
}
```

## Output



**Q2. Create an external CSS file and use it to modify hello world page background color to grey**

**Solution**

**Style.css**

```
body
{
    background-color: gray;
}
```

**Hello.html**

```
<!DOCTYPE html>
<html lang="en" xmlns:th="http://www.thymeleaf.org">
<head>
    <meta charset="UTF-8">
    <title>Hello World</title>
    <link rel="stylesheet" th:href="@{/css/style.css}">
</head>
<body>
<h1>Hello World</h1>
</body>
</html>
```

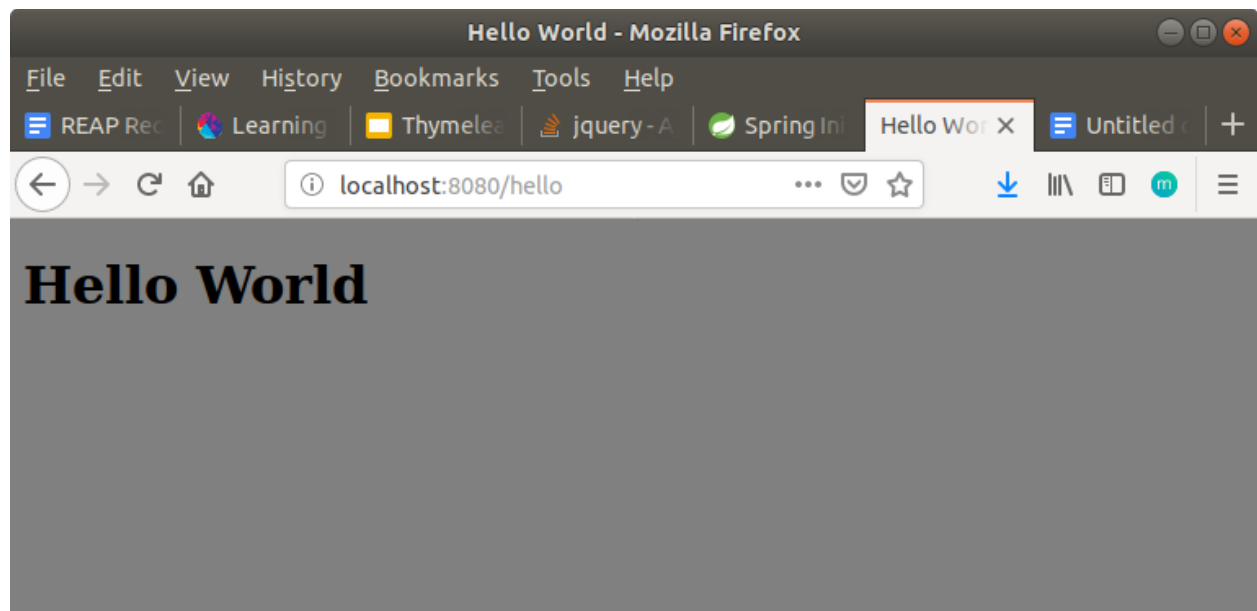
**ApplicationController.java**

```
package com.demo.thymeleaf.controller;

import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.GetMapping;

@Controller
public class ApplicationController {
    //Question2
    @GetMapping("/hello")
    public String hello()
    {
        return "hello";
    }
}
```

## Output



**Q3. Create an Employee registration form and bind it's requested values on the controller.**

**Solution**

**registrationForm.html**

```
<!DOCTYPE html>
<html lang="en" xmlns:th="http://www.thymeleaf.org">
<head>
  <meta charset="UTF-8">
  <title>Employee Registration</title>
</head>
<body>
<form method="post" th:object="${employee}">
  <label for="firstName">FirstName</label>
  <input type="text" th:field="**{firstName}" id="firstName">
  <label for="lastName">LastName</label>
  <input type="text" th:field="**{lastName}" id="lastName">
  <input type="submit" value="Register">
</form>

</body>
</html>
```

**Registered.html**

```
<!DOCTYPE html>
<html lang="en" xmlns:th="http://www.w3.org/1999/xhtml">
<head>
  <meta charset="UTF-8">
  <title>Title</title>
</head>
<body>
<h2>You have been successfully registered</h2>
FirstName: <span th:text="${firstName}"></span>
LastName: <span th:text="${lastName}"></span>
</body>
</html>
```

**EmployeeController.java**

```
package com.demo.thymeleaf.employee.Controller;

import com.demo.thymeleaf.employee.Employee;
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.PostMapping;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.servlet.ModelAndView;

@Controller
@RequestMapping("/employee")
public class EmployeeController {
```

```

    @GetMapping("/registrationForm")
    public String registrationForm(Model model)
    {
        model.addAttribute("employee",new Employee());
        return "/employee/registrationForm";
    }
    @PostMapping("/registrationForm")
    public ModelAndView registeredDetails(@ModelAttribute Employee employee)
    {
        ModelAndView modelAndView=new ModelAndView("/employee/registered");
        modelAndView.addObject("firstName",employee.getFirstName());
        modelAndView.addObject("lastName",employee.getLastName());
        return modelAndView;
    }
}

```

## Employee.java

```

package com.demo.thymeleaf.employee;

import org.springframework.web.bind.annotation.ModelAttribute;

public class Employee {
    private String firstName;
    private String lastName;

    public Employee() {
    }

    public Employee(String firstName, String lastName) {
        this.firstName = firstName;
        this.lastName = lastName;
    }

    public String getFirstName() {
        return firstName;
    }

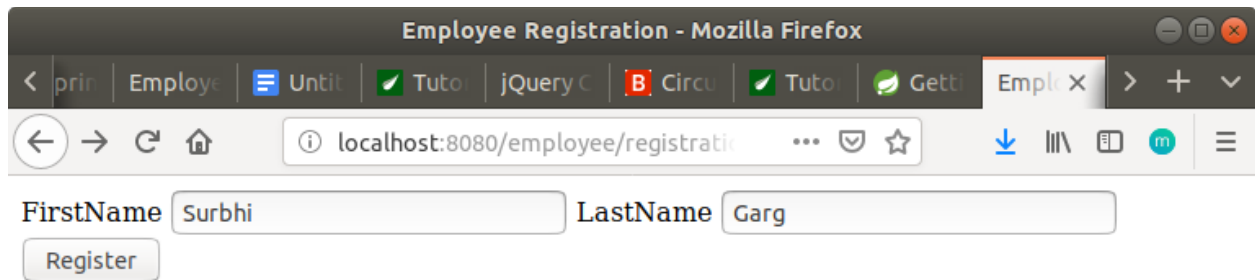
    public void setFirstName(String firstName) {
        this.firstName = firstName;
    }

    public String getLastName() {
        return lastName;
    }

    public void setLastName(String lastName) {
        this.lastName = lastName;
    }
}

```

## Output



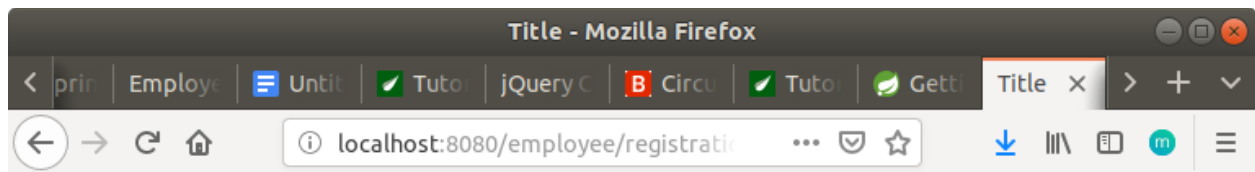
A screenshot of a Mozilla Firefox browser window titled "Employee Registration - Mozilla Firefox". The address bar shows "localhost:8080/employee/registrati". The form contains two input fields: "FirstName" with the value "Surbhi" and "LastName" with the value "Garg". Below these fields is a "Register" button. The browser's tab bar shows several open tabs, including "prim", "Employee", "Until", "Tuto", "jQuery C", "B Circu", "Tuto", "Getti", and "Empl x".

Employee Registration - Mozilla Firefox

localhost:8080/employee/registrati

FirstName Surbhi LastName Garg

Register



**Q4. Create a dynamic HTML page and render an Employee table with emp object return from the controller.**

**Solution**

**EmployeeController.java**

```
package com.demo.thymeleaf.employee.Controller;
```

```
import com.demo.thymeleaf.employee.Employee;
```

```
import org.springframework.stereotype.Controller;
```

```
import org.springframework.ui.Model;
```

```
import org.springframework.web.bind.annotation.*;
```

```
import org.springframework.web.servlet.ModelAndView;
```

```
import java.util.Arrays;
```

```
import java.util.List;
```

```
@Controller
```

```
@RequestMapping("/employee")
```

```
public class EmployeeController {
```

```
    //Question4
```

```
    @GetMapping("/table")
```

```
    public String listEmployees(Model model)
```

```
    {
```

```
        model.addAttribute("employeeList",Arrays.asList(new Employee("Surbhi","Garg"),new  
Employee("Vagish","Dixit")));
```

```
        return "/employee/table";
```

```
    }
```

```
}
```

**Table.html**

```
<!DOCTYPE html>
```

```
<html lang="en" xmlns:th="http://www.thymeleaf.org">
```

```
<head>
```

```
    <meta charset="UTF-8">
```

```
    <title>Employee List</title>
```

```
</head>
```

```
<body>
```

```
<table>
```

```
    <tr>
```

```
        <th>
```

```
            FirstName
```

```
        </th>
```

```
        <th>
```

```
            LastName
```

```
        </th>
```

```
    </tr>
```

```
    <tr th:each="employee:${employeeList}">
```

```
        <td th:text="${employee.firstName}"></td>
```

```
        <td th:text="${employee.lastName}"></td>
```

```
    </tr>
```

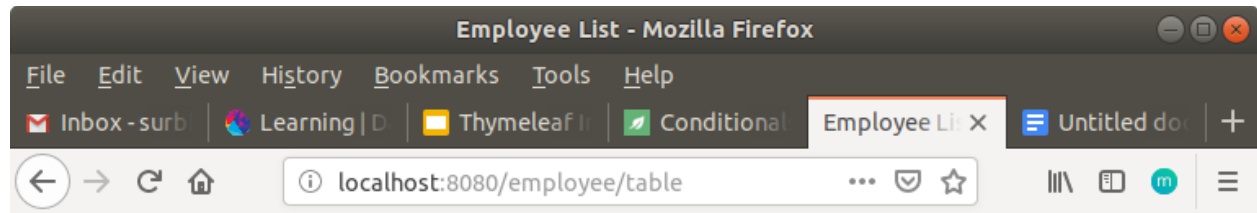
```
</table>
```



```
</body>
```

```
</html>
```

## Output



FirstName	LastName
-----------	----------

Surbhi	Garg
--------	------

Vagish	Dixit
--------	-------

**Q5. Show different custom greeting message based on the type of user. For example, if the user is of type admin=true then show "Hello Admin" message on Page else show "Hello User" basis on isAdmin attribute value passed in model data.**

**Solution**

**ApplicationController.java**

```
package com.demo.thymeleaf.controller;

import com.demo.thymeleaf.employee.Employee;
import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.ModelAttribute;
```

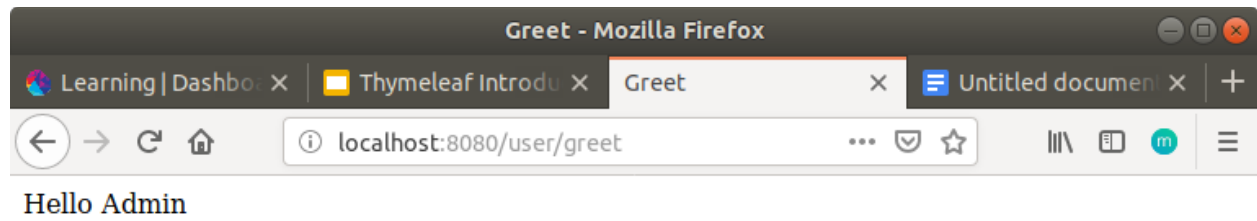
```
@Controller
public class ApplicationController {
    //Question5
    @GetMapping("/user/greet")
    public String greet(Model model)
    {
        model.addAttribute("isAdmin",true);
        return "/user/greet";
    }
}
```

**Greet.html**

```
<!DOCTYPE html>
<html lang="en" xmlns:th="http://www.thymeleaf.org">
<head>
    <meta charset="UTF-8">
    <title>Greet</title>
</head>
<body>
<div th:if="{isAdmin}">
    Hello Admin
</div>
<div th:unless="{isAdmin}">
    Hello User
</div>

</body>
</html>
```

## Output



**Q6. Create and split common header and footer in the above-created listing page.**

**Solution**

**Header.html**

```
<!DOCTYPE html>
<html lang="en" xmlns:th="http://www.w3.org/1999/xhtml">
<head>
  <meta charset="UTF-8">
  <title>Header</title>
</head>
<body>
<div th:fragment="header">
  TTN Details
</div>

</body>
</html>
```

**Footer.html**

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Footer</title>
</head>
<body>
<div th:fragment="footer">
  copyright 2019
</div>

</body>
</html>
```

**Table.html**

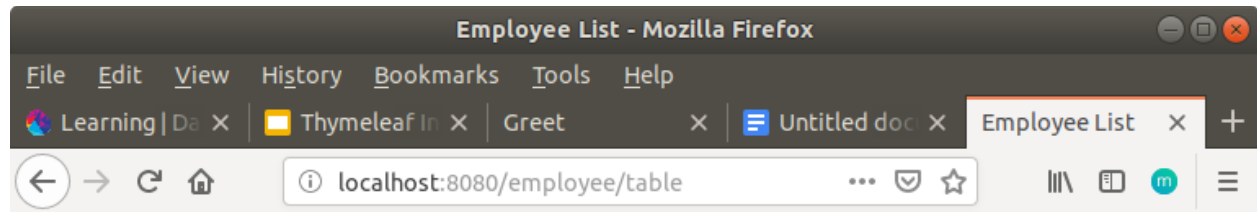
```
<!DOCTYPE html>
<html lang="en" xmlns:th="http://www.thymeleaf.org">
<head>
  <meta charset="UTF-8">
  <title>Employee List</title>
</head>
<body>
<!--ForQuestion6 header-->
<div th:replace="/fragments/header.html ::header"></div>
<table>
  <tr>
    <th>
      FirstName
    </th>
    <th>
      LastName
    </th>
  </tr>
```

```

<tr th:each="employee:${employeeList}">
  <td th:text="${employee.firstName}"></td>
  <td th:text="${employee.lastName}"></td>
</tr>
</table>
<!--ForQuestion6-footer-->
<div th:replace="/fragments/footer.html ::footer"></div>
</body>
</html>

```

## Output



TTN Details

FirstName	LastName
-----------	----------

Surbhi	Garg
--------	------

Vagish	Dixit
--------	-------

copyright 2019

## Q7. Make a web page to show server time using ajax call

### Solution

#### ApplicationController.java

```
package com.demo.thymeleaf.controller;

import com.demo.thymeleaf.employee.Employee;
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.ResponseBody;

import java.sql.Time;
import java.time.LocalDate;
import java.time.LocalDateTime;
```

```
@Controller
public class ApplicationController {
    // Question7
    @GetMapping("/getServerTime")
    @ResponseBody
    public LocalDateTime getServerTime() {
        return LocalDateTime.now();
    }
    @GetMapping("/serverTime")
    public String serverTime()
    {
        return "serverTime";
    }
}
```

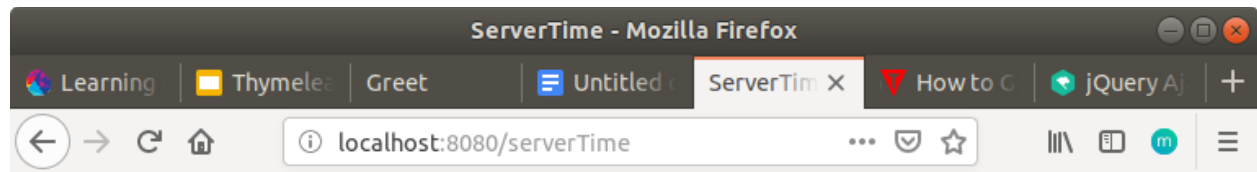
#### serverTime.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>ServerTime</title>
    <script
        src="http://code.jquery.com/jquery-2.2.4.min.js"
        integrity="sha256-BbhdlvQf/xTY9gja0Dq3HiwQF8LaCRTXxZKRutelT44="
        crossorigin="anonymous"></script>
    <script type="text/javascript">
        $(function(){
            var time=$.get({
                url:"/getServerTime"
            });
            time.done(function(data){
                if(data)
```

```
    {
      $("#serverTime").html(data);
    }
  });

  time.fail(function (jqXHR, textStatus) {
    console.log("Error in fetching time from server");
  });
})
</script>
</head>
<body>
<p id="serverTime"></p>
</body>
</html>
```

## Output



22:31:40.25

**Q8. Create a user listing table show alternate row color using a conditional tag.**

**Solution**

**ApplicationController.java**

```
package com.demo.thymeleaf.controller;
```

```
import com.demo.thymeleaf.employee.Employee;
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.ResponseBody;
```

```
import java.sql.Time;
import java.time.LocalDate;
import java.time.LocalTime;
import java.util.Arrays;
import java.util.List;
```

**@Controller**

```
public class ApplicationController {
    // Question8
    @GetMapping("/user/list")
    public String userList(Model model)
    {
        List<String>users= Arrays.asList("Surbhi Garg","Vagish Dixit","Reshma Mohan","Yukti Sharma");
        model.addAttribute("userList",users);
        return "/user/list";
    }
}
```

**listStyle.css**

```
.odd
{
    background-color: deepskyblue;
}
.even
{
    background-color: bisque;
}
```

**List.html**

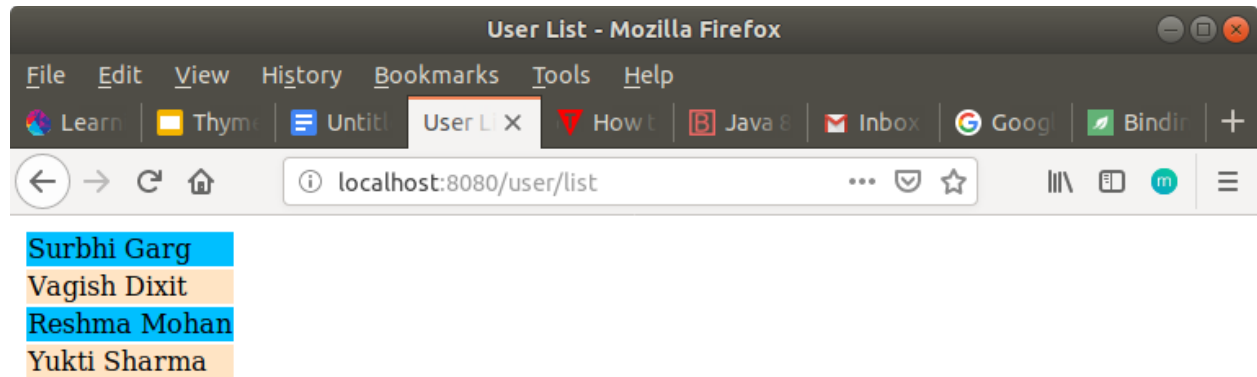
```
<!DOCTYPE html>
<html lang="en" xmlns:th="http://www.w3.org/1999/xhtml">
<head>
    <meta charset="UTF-8">
    <title>User List</title>
    <link rel="stylesheet" th:href="@{/css/listStyle.css}"/>
</head>
<body>
```



```
<table>
  <tr th:each="user,iterationStatus:${userList}" class="row"
th:classappend="${iterationStatus.odd}?'odd':'even'">
    <td th:text="${user}"></td>
  </tr>
</table>

</body>
</html>
```

## Output



**Q9. Create an ENUM with values USER, ADMIN, SUPER\_ADMIN and iterate it to show a custom message to users basis on the ENUM value**

**Solution**

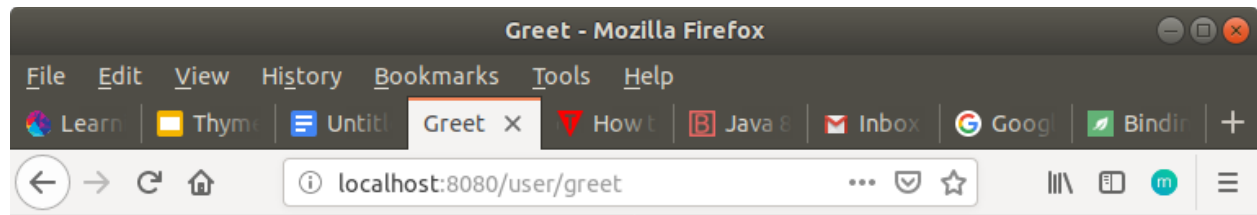
**UserType.java**

```
package com.demo.thymeleaf.enums;  
public enum UserType  
{  
    USER,ADMIN,SUPER_ADMIN;  
}
```

**Greet.html**

```
<!DOCTYPE html>  
<html lang="en" xmlns:th="http://www.thymeleaf.org">  
<head>  
    <meta charset="UTF-8">  
    <title>Greet</title>  
</head>  
<body>  
<div th:if="{isAdmin}">  
    Hello Admin  
</div>  
<div th:unless="{isAdmin}">  
    Hello User  
</div>  
  
<!--Question9-->  
<span th:each="userType:${T(com.demo.thymeleaf.enums.UserType).values()}">  
    <p th:if="{userType.name()}=='USER' ">Hello User</p>  
    <p th:if="{userType.name()}=='ADMIN' ">Hello ADMIN</p>  
    <p th:if="{userType.name()}=='SUPER_ADMIN' ">Hello Super admin</p>  
</span>  
  
</body>  
</html>
```

## Output



Hello Admin

Hello User

Hello ADMIN

Hello Super admin

**Q10.Add loader in registration form submission which renders load till server return success form submission response.**

**Solution**

**EmployeeController.java**

```
package com.demo.thymeleaf.employee.Controller;
```

```
import com.demo.thymeleaf.employee.Employee;
import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.*;
import org.springframework.web.servlet.ModelAndView;
```

```
import java.util.Arrays;
import java.util.List;
```

```
@Controller
@RequestMapping("/employee")
public class EmployeeController {
    // Question10
    @RequestMapping(value = "/registrationFormAjax",method = RequestMethod.POST)
    @ResponseBody
    public String register(@ModelAttribute Employee employee) {
        try {
            Thread.sleep(2000);
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
        System.out.println("Info"+employee.getFirstName());
        return employee.getFirstName();
    }
}
```

```
}
```

registrationForm.html

```
<!DOCTYPE html>
<html lang="en" xmlns:th="http://www.thymeleaf.org">
<head>
    <meta charset="UTF-8">
    <title>Employee Registration</title>
    <!--<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
integrity="sha384-q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
crossorigin="anonymous"></script>-->
    <!--<script src="https://cdn.jsdelivr.net/npm/popper.js@1.14.7/umd/popper.min.js"
integrity="sha384-UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1cIHTMGA3JDZwrnQq4sF86dIHNDz0W1"
crossorigin="anonymous"></script>-->
    <script
        src="http://code.jquery.com/jquery-2.2.4.min.js"
        integrity="sha256-BbhdlvQf/xTY9gja0Dq3HiwQF8LaCRTxXZKRutelT44="
        crossorigin="anonymous"></script>
```

```

<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"
integrity="sha384-UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dlHNDz0W1"
crossorigin="anonymous"></script>
<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"
integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T"
crossorigin="anonymous"/>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"
integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoily6OrQ6VrjlEaFf/nJGzlxFDsf4x0xIM+B07jRM"
crossorigin="anonymous"></script>
<!--Ajax for question10-->
<script type="text/javascript">
    $(function () {
        $("#wait").hide();
        $("#registrationForm").submit(
            function (event) {
                event.preventDefault();
                var register= $.post({
                    url: "/employee/registrationFormAjax",
                    data: $("#registrationForm").serialize(),
                    beforeSend: function() { $('#wait').modal('show')},
                    complete: function() { $('#wait').modal('hide'); }
                });
                register.done(function (data) {
                    if(data)
                    {
                        $('#status').html("Registered "+data)
                    }
                });
                register.fail(function (jqXHR,textStatus) {
                    console.log("Error in registration"+jqXHR+textStatus)
                });
            });
    });
</script>
</head>
<body>
<form method="post" th:object="${employee}" id="registrationForm">
    <label for="firstName">FirstName</label>
    <input type="text" th:field="**{firstName}" id="firstName">
    <label for="lastName">LastName</label>
    <input type="text" th:field="**{lastName}" id="lastName">
    <input type="submit" value="Register">
</form>
<!-- Modal -->

```

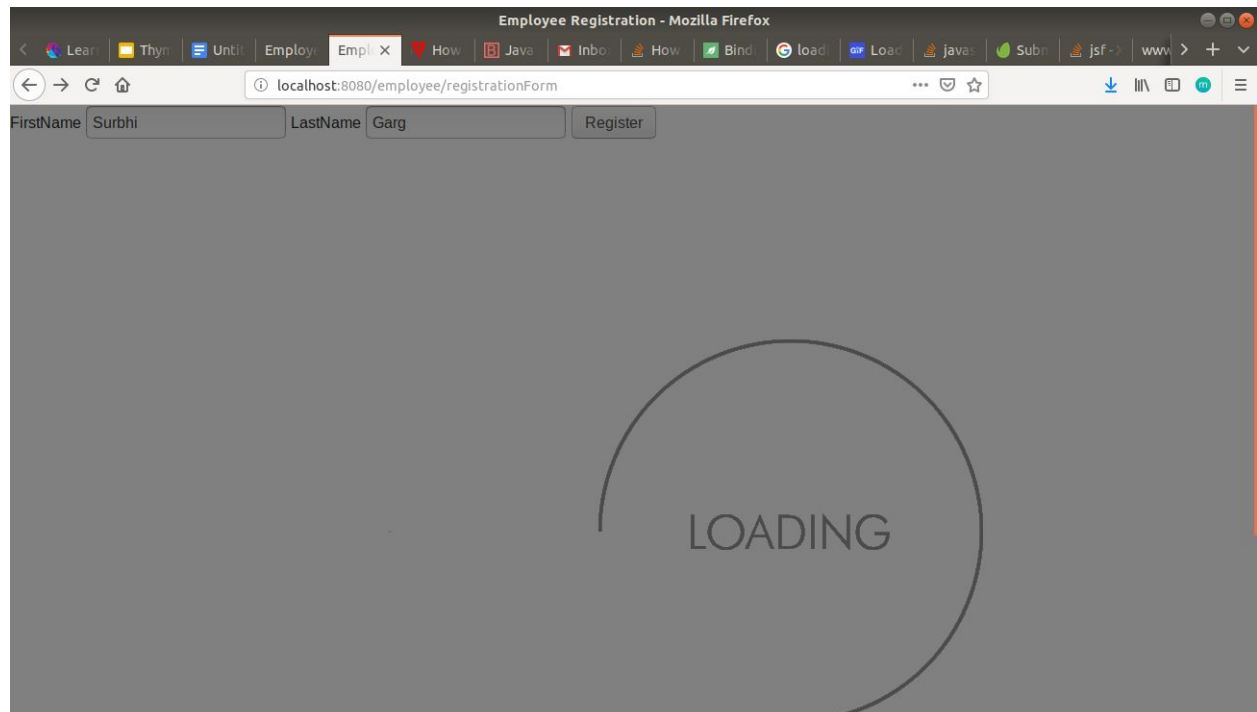
```

<div id="wait" class="modal fade" tabindex="-1" role="dialog" aria-labelledby="exampleModalCenterTitle"
aria-hidden="true">
  <div class="modal-dialog modal-dialog-centered" role="document">
    <div class="modal-content">
      </div>
      <div class="modal-body">
        
      </div>
    </div>
  </div>
</div>
<p id="status"></p>

</body>
</html>

```

## Output



Employee Registration - Mozilla Firefox

File Edit View History Bookmarks Tools Help

Learn Thym Untill Employ: Empli X How Java Inbo: How Bind load Load Load Subn jsf -> www > + v

localhost:8080/employee/registrationForm

FirstName Surbhi LastName Garg Register

Registered Surbhi