

How **Thymeleaf** Engine Works

By Ramesh Fadatara (Java Guides)

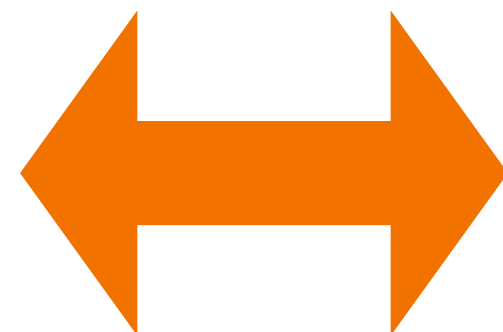
Model

```
public class StudentDto {  
    private Long id;  
    private String firstName;  
    private String lastName;  
    private String email;  
}
```

```
// getter/setter methods
```

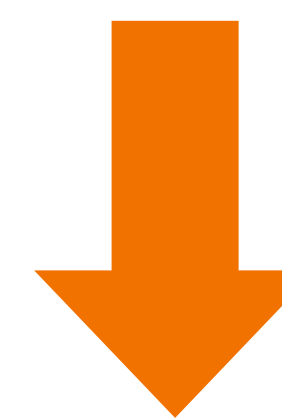
```
// handler method to handle list students request  
@GetMapping("/{students}")  
public String listStudents(Model model){  
    List<StudentDto> students = studentService.getAllStudents();  
    model.addAttribute( attributeName: "students", students);  
    return "students";  
}
```

Controller



Thymeleaf
Template

```
<!DOCTYPE html>  
<html lang="en"  
    xmlns:th="http://www.thymeleaf.org"  
>  
<head...>  
<body>  
    <table class="table table-striped table-hover table-bordered">  
        <thead class="table-dark">  
            <tr>  
                <th>Student First Name</th>  
                <th>Student Last Name</th>  
                <th>Student Email</th>  
            </tr>  
        </thead>  
        <tbody>  
            <tr th:each="student : ${students}">  
                <td th:text="${student.firstName}"></td>  
                <td th:text="${student.lastName}"></td>  
                <td th:text="${student.email}"></td>  
            </tr>  
        </tbody>  
    </table>  
</div>  
</body>  
</html>
```



Output

localhost:8080/students

Student First Name	Student Last Name	Student Email
ram	jadhav	ram@gmail.com
vikram	jadhav	vikram@gmail.com
umesh	fadatare	umesh@gmail.com

Model

```
public class Person {  
    private String firstName;  
    private String lastName;  
}
```

List<Person> persons



View (Thymeleaf Template)

```
<!DOCTYPE HTML>  
<html xmlns:th="http://www.thymeleaf.org">  
  <head>  
    <meta charset="UTF-8" />  
    <title>Person List</title>  
    <link rel="stylesheet" type="text/css"  
          th:href="@{/css/style.css}"/>  
  </head>  
  <body>  
    <h1>Person List</h1>  
    <a href="addPerson">Add Person</a>  
    <br/><br/>  
    <div>  
      <table border="1">  
        <tr>  
          <th>First Name</th>  
          <th>Last Name</th>  
        </tr>  
        <tr th:each="person : ${persons}">  
          <td th:utext="${person.firstName}">...</td>  
          <td th:utext="${person.lastName}">...</td>  
        </tr>  
      </table>  
    </div>  
  </body>  
</html>
```

Thymeleaf Engine



```
<!DOCTYPE HTML>  
<html>  
  <head>  
    <meta charset="UTF-8" />  
    <title>Person List</title>  
    <link rel="stylesheet" type="text/css"  
          href="my-context-path/css/style.css"/>  
  </head>  
  <body>  
    <h1>Person List</h1>  
    <a href="addPerson">Add Person</a>  
    <br/><br/>  
    <div>  
      <table border="1">  
        <tr>  
          <th>First Name</th>  
          <th>Last Name</th>  
        </tr>  
        <tr>  
          <td>Bill</td>  
          <td>Gates</td>  
        </tr>  
        <tr>  
          <td>Steve</td>  
          <td>Jobs</td>  
        </tr>  
      </table>  
    </div>  
  </body>  
</html>
```

The **spring-boot-starter-thymeleaf** starter Dependency

- Spring Boot provides auto-configuration for Thymeleaf by adding the **spring-boot-starter-thymeleaf** dependency
- **spring-boot-starter-thymeleaf** dependency integrate Thymeleaf with Spring integration dependencies
- Spring boot automatically configures view resolver for Thymeleaf
- Spring boot auto configure location for Thymeleaf template. By default, Spring boot will look for Thymeleaf templates in **/resources/templates** folder

Thymeleaf Hello World Example

By Ramesh Fadatara (Java Guides)

Development Steps

1. Create Handler Method to Handle the HTTP Request (GET or POST)
2. Handler Method Returns Model and View (Thymeleaf Template name)
3. Create Thymeleaf Template View and Display Model Data

Spring Boot Web MVC Architecture

