

Applicatieontwikkeling in Java - Spring Boot MVC & Thymeleaf

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



Structuur van packages binnen elk project:

```
fact.it.voorbeeldfietsen
> controller
> model
```

Uitzetten Spring logo en INFO msg bij opstart

In het application.properties bestand:

```
spring.main.banner-mode=off
spring.main.log-startup-info=false
```



Weergeven van tekst in HTML tags

```
<span th:text="${reservatie.getStudent().getNaam()}" />
```

```
<p th:text="${reservatie.getStudent().getNaam()}"></p>
```

Weergeven van geconcateneerde tekst in HTML tags

```
<p th:text="'Reservatie van ' + ${reservatie.getStudent().getVoornaam()} + ' ' + ${reservatie.getStudent().getNaam()}"></p>
```

Spring Boot MVC - Controller

```
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.RequestMapping;
```

```
@Controller
public class MainController {

    @RequestMapping("/")
    public String index() {
        return "index";
    }
}
```

In Controller - Uitvoeren van methode na opstart met @PostConstruct

```
private ArrayList<FietsType> fietstypes = new ArrayList<>();

@PostConstruct
public void postConstruct() {
    fietstypes.add(new FietsType(true, "small"));
    fietstypes.add(new FietsType(true, "medium"));
    fietstypes.add(new FietsType(false, "small"));
    fietstypes.add(new FietsType(false, "medium"));
}
```

In Controller - Model & request bij methodes met @RequestMapping

```
@RequestMapping("/bevestiging-fiets")
public String bevestigingFiets(Model model, HttpServletRequest request) {
    Integer grootte = Integer.parseInt(request.getParameter("grootte"));
    FietsType fietstype = fietstypes.get(grootte);
    int jaartal = Integer.parseInt(request.getParameter("jaartal"));

    Fiets fiets = new Fiets(fietstype, jaartal);

    model.addAttribute("fiets", fiets);

    return "bevestiging-fiets";
}
```

Conditie - "If" & "Switch"

```
<p th:if="${totaalReservaties.size()==0}">Geen reservaties!</p>
```

```
<table th:if="${totaalReservaties.size()>0}" border="1">
    // Inhoud table met reservaties
</table>
```

```
<div th:switch="${reservatie.getDag()}">
    <p th:case="zaterdag">Reservatie op zaterdag</p>
    <p th:case="zondag">Reservatie op zondag</p>
    <p th:case="*">Reservatie op een weekday</p>
</div>
```

"For-each" iteratie over een ArrayList uit het Model

Een optie in een drop-down element, per object in de ArrayList

```
<select name="productChoice" id="productChoice">
    <option th:each="product : ${products}" th:value="${product.id}"
            th:text="${product.name}"></option>
</select>
```

Een list-item per object in de ArrayList

```
<ul>
    <li th:each="reservatie : ${reservaties}">
        <span th:text="${reservatie.getStudent().getNaam()}" />
    </li>
</ul>
```

Een rij in de tabel per object in de ArrayList

```
<table border="1">
    <tr><td>rNummer</td><td>Naam student</td></tr>
    <tr th:each="reservatie : ${totaalReservaties}">
        <td th:text="${reservatie.getStudent().getrNummer()}"></td>
        <td th:text="${reservatie.getStudent().getNaam()}"></td>
    </tr>
</table>
```

Referenties met @{ }

Referentie naar de submit-URL van een formulier

```
<form th:action="@{/edit-fiets(id=${fiets.getId()})}" method="post"> </form>
```

Referentie naar media / css / enz. in de static folder

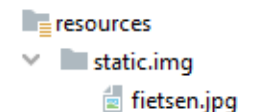
```

```

Referentie in een hyperlink dynamisch opbouwen

```
<a th:href="@{/reservatie-details(volnummer=${reservatie.getId()})}" />
```

resulteert bijvoorbeeld in:



"For" iteratie over een tijdelijke sequence van Integers

Lokaal op de Thymeleaf pagina aanmaken en gebruiken van een sequence van Integers

Als Thymeleaf: `th:each="i: ${#numbers.sequence(2015, 2020, 1)}"`

In Python: `for i in range(2015, 2021, 1)`

Als Thymeleaf: `th:each="i: ${#numbers.sequence(2, 30, 3)}"`

In Python: `for i in range(2, 31, 3)`

Een radiobutton per Integer-object in de sequence

```
<th:block th:each="i: ${#numbers.sequence(2015, 2020, 1)}">
    <input type="radio" name="jaartal" th:value="${i}" th:id="${i}">
    <label th:for="${i}" th:text="${i}"></label>
</th:block>
```

☐ 2015 ☐ 2016 ☐ 2017 ☐ 2018 ☐ 2019 ☐ 2020