FreeMarker

Hello

Tomasz Odwald

Freelance Software Developer

Pasjonat aplikacji mobilnych

FreeMarker - Agenda

1. Po co nam szablony?
2. Czym jest FreeMarker i dlaczego właśnie on?
3. Składnia języka FTL
4. Połączenie FreeMarkera z Servletem

1. Szablony

Czyli co zrobić, kiedy sam HTML nie wystarczy

Statyczny HTML

```
<!DOCTYPE html>
<html>
<head>
<title>Statyczna strona HTML</title>
</head>
<body>

<h1>Nagłówek</h1>

<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin iaculis, ligula ut ultricies luctus, mi massa elementum mauris, vel ullamcorper ligula tellus ac orci. Cras pulvinar dapibus magna. Aliquam erat volutpat. Vestibulum volutpat ultricies elit. Pellentesque nulla eros, vestibulum id mollis sed, blandit ac diam. Proin id nunc justo. Praesent pulvinar iaculis velit.</p>


</body>
</html>
```

Statyczny HTML - problemy

- nie możemy skorzystać z zewnętrznych danych
- nie możemy wyświetlać różnej zawartości na jednej stronie
- nie posiada logiki wykonywanej - warunki, pętle, itd.
- nie możemy go podzielić na mniejsze elementy

Do aplikacji internetowych potrzebujemy czegoś bardziej dynamicznego ;)

Szablony

```
<!DOCTYPE html>
<html>
<head>
<title>${pageTitle}</title>
</head>
<body>

<h1>${header}</h1>

<p>${content}</p>


</body>
</html>
```

Szablony - pozwalają na

- dzielenie widoków na mniejsze elementy
- wykorzystanie zewnętrznych danych
- zastosowanie warunków i pętli
- wiele innych fajnych rzeczy :)

Powodują że HTML staje się dynamiczny.

2. FreeMarker

Czym jest i dlaczego z niego chcemy skorzystać?

FreeMarker - co to?

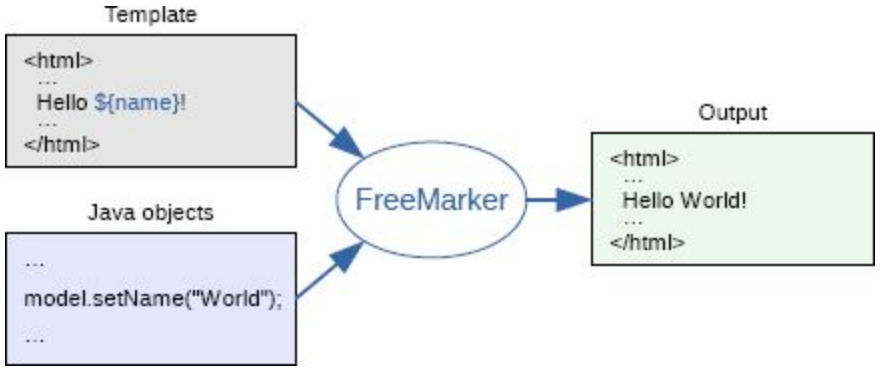
Generator szablonów (*ang. template engine)* - wykorzystywany do generowanie plików tekstowych (nie tylko HTML).

https://freemarker.apache.org/

Sandbox:

https://try.freemarker.apache.org/

FreeMarker - jak to działa?



FreeMarker - data model

Struktura drzewa

```
(root)
  |
  +- user = "Big Joe"
  |
  +- latestProduct
      |
      +- url = "products/greenmouse.html"
      |
      +- name = "green mouse"
```

FreeMarker - typy danych

- skalarne - String, Number, Boolean, Date-like,
- kontenery - Hash, Sequence, Collection,
- podprogramy - metody, funkcje, makra,
- inne - Node, Markup.

FreeMarker - dlaczego nie JSP?

- template loaders
- nie potrzebuje importów Javowych
- pozwala na użycie tagów JSP
- wywoływanie metod dla obiektów
- obsługa null i pustych string
- wsparcie dla JSON

3. FreeMarker Template Language

Składnia i podstawowe pojęcia

Interpolacja - ${...}

Wartość wstrzykiwana do kodu HTML, można wykonywać dodatkowe modyfikacje danych:

${user.name}

${product.cost / 100}

Tagi i komentarze

Tagi wbudowane we FreeMarkera:

<#tag ...>

Tagi zdefiniowane przez użytkownika:

<@tag ...>

Komentarze:

<#-- komentarz -->

Sterowanie warunkowe

<#if *value* **==** **“***expected***”>**

I’ve expected that...

<#elseif *otherValue* **gte** *50***>**

I knew it could happen.

<#else>

That was unexpected!

</#if>

Pętle

Pętla we FreeMarkerze jest analogiczna do pętli for w Javie:

```
<ul>
  <#list users as user>
    <li>${user.name}</li>
  </#list>
</ul>
```

Pętle - obsługa pustej listy

Jeśli lista users jest pusta, to znaczniki <ul> i tak zostaną utworzone. Można tego uniknąć w ten sposób:

```
<#list users>
  <ul>
    <#items as user>
      <li>${user.name}</li>
    </#items>
  </ul>
</#list>
```

Deklarowanie zmiennych

Zmienne można deklarować też bezpośrednio w szablonie:

<#assign value **=** ”wartosc”**>**

Można też skorzystać z zakresu wartości:

<#assign digits **=** 0..<10**>**

Separatory

Separatory służą do oddzielenia wartości wypisywanych z kolekcji. Jest wstawiany za każdym razem, kiedy występuje kolejny element w kolekcji. Można też użyć w nim tagu **else**, który zostanie wykonany w przypadku pustej kolekcji.

```
<#list users as user>
  ${user.name}<#sep>, <#else>No users found</#sep>
</#list>
```

Metody on-the-fly

Metody zwracające dodatkowe wartości dla obiektu bez potrzeby jego modyfikowania:

user.name**?length** - zwraca długość string

users**?size** - zwraca rozmiar kolekcji

user**?index** - indeks obiektu w pętli (*od 0*, ***counter*** *od 1*)

Obsługa null i wartości domyślne

Sprawdzenie null:

<#if user**??>**Welcome **${**user.name**}<#else>**Welcome guest**</#if>**

Wartość domyślna:

Welcome **${(**user.name**) !** “guest”**}**

Include i import

Include pozwala na wstrzyknięcie jednego szablonu do drugiego:

<#include “/footer.ftl**”>**

Import pozwala na zapisanie innego szablonu do makra:

<#import “/footer.ftl**” as** footer**>**

<@footer date=”2019-2020”**/>**

Makra

Zdefiniowanie makra:

<#macro nazwa obiekt**>**

<h1>Nagłówek h1</h1><h3>**${**obiekt**}**</h3>

</#macro>

Użycie:

<@nazwa obiekt=”Nagłówek h3”**/>**

Format pliku - rozszerzenie .ftlh

Dla zapewnienia bezpieczeństwa szablonu HTML przed wstrzykiwaniem złośliwego kodu z zewnątrz, zaleca się używanie szablonów z rozszerzeniem **.ftlh**. Jest to automatyczna informacja dla FreeMarkera, żeby nie wykonywał żadnego kodu przekazanego jako zawartość obiektu.

Aby aktywować tą funkcję w plikach **.ftl** należy dodać:

<#ftl output_format="HTML">

4. FreeMarker & Servlet

Integracja FreeMarkera z Servletami Javowymi

Repozytorium

https://github.com/infoshareacademy/jjddr1-materialy-freemarker

Maven

```
<dependency>
  <groupId>org.freemarker</groupId>
  <artifactId>freemarker</artifactId>
  <version>2.3.30</version>
</dependency>
```

Wstępna konfiguracja

```
import freemarker.template.Configuration;

import freemarker.template.Template;


Configuration cfg = new Configuration(Configuration.VERSION_2_3_30);

cfg.setServletContextForTemplateLoading(getServletContext(), "WEB-INF/templates");

cfg.setDefaultEncoding("UTF-8");

cfg.setTemplateExceptionHandler(TemplateExceptionHandler.RETHROW_HANDLER);

cfg.setLogTemplateExceptions(false);

cfg.setWrapUncheckedExceptions(true);

cfg.setFallbackOnNullLoopVariable(false);
```

Użycie szablonu

```
Map<String, Object> root = new HashMap<String, Object>();

root.put(key, value);


Template template = cfg.getTemplate(templateName);


Writer out = response.getWriter();

template.process(root, out);
```

Dzieki