

flatMap Developer Language Skills

Klasse Developer

Eine Software Team besteht aus mehreren Entwicklern mit unterschiedlichen Kenntnissen von einer oder mehrerer Programmiersprachen. Jeder Entwickler wird über die Klasse

```
ch.std.jp2.lambda.flatMap.Developer abgebildet:package
ch.std.jp2.lambda.flatMap;&#xA;&#xA;import java.util.HashSet;&#xA;import
java.util.Set;&#xA;&#xA;public class Developer {&#xA;&#xA; private String name;&#xA; private Set
languages;&#xA;&#xA; public Developer(String name) {&#xA; this.languages = new
HashSet<&#xA;&#xA;();&#xA; this.name = name;&#xA; }&#xA;&#xA; public void add(String language)
{&#xA; this.languages.add(language);&#xA; }&#xA;&#xA; public Set getLanguages() {&#xA;
return languages;&#xA; }&#xA;&#xA; public String getName() {&#xA; return name;&#xA; }&#xA;}
```

Flatmap Unit Test

Der folgende Unit Test zeigt die flatMap Anwendung:

```
package
ch.std.jp2.lambda.flatMap.test;&#xA;&#xA;import static
org.junit.Assert.assertTrue;&#xA;&#xA;import java.util.ArrayList;&#xA;import
java.util.List;&#xA;import java.util.stream.Collectors;&#xA;&#xA;import
org.junit.Test;&#xA;&#xA;import ch.std.jp2.lambda.flatMap.Developer;&#xA;&#xA;public class
FlatMapTest {&#xA;&#xA; @Test&#xA; public void flatMap() {&#xA;
List<Developer> team = new ArrayList<&#xA;&#xA;();&#xA; Developer polyglot
= new Developer(&#34;esoteric&#34;);&#xA; polyglot.add(&#34;clojure&#34;);&#xA;
polyglot.add(&#34;scala&#34;);&#xA; polyglot.add(&#34;groovy&#34;);&#xA;
polyglot.add(&#34;go&#34;);&#xA;&#xA; Developer busy = new
Developer(&#34;pragmatic&#34;);&#xA; busy.add(&#34;java&#34;);&#xA;
busy.add(&#34;javascript&#34;);&#xA;&#xA; team.add(polyglot);&#xA;
team.add(busy);&#xA;&#xA; List<String> teamLanguages = team.stream().&#xA;
map(d -&#xA;&#xA; d.getLanguages()).&#xA; flatMap(l -&#xA;&#xA; l.stream()).&#xA;
collect(Collectors.toList());&#xA;
assertTrue(teamLanguages.containsAll(polyglot.getLanguages()));&#xA;
assertTrue(teamLanguages.containsAll(busy.getLanguages()));&#xA; }&#xA;}
```

Mit FlatMap wird die Liste der Developer und deren Language Skills auf eine einzelne Liste reduziert.

Beispiel Code

Sie finden das komplette Beispiel hier

Kontakt

Simtech AG
Finkenweg 23
3110 Münsingen
Schweiz

Impressum

Das Copyright für sämtliche Inhalte dieser Website liegt bei Simtech AG, Schweiz.
Beachten Sie auch unsere Hinweise zum Urheberrecht, Datenschutz und Haftungsausschluss.
Jeder Hinweis auf Fehler nehmen wir gerne entgegen.

Copyright

2024 Simtech AG, All rights reserved, Powered by stack.ch written in Golang by Daniel Schmutz

<https://www.simtech-ag.ch/>