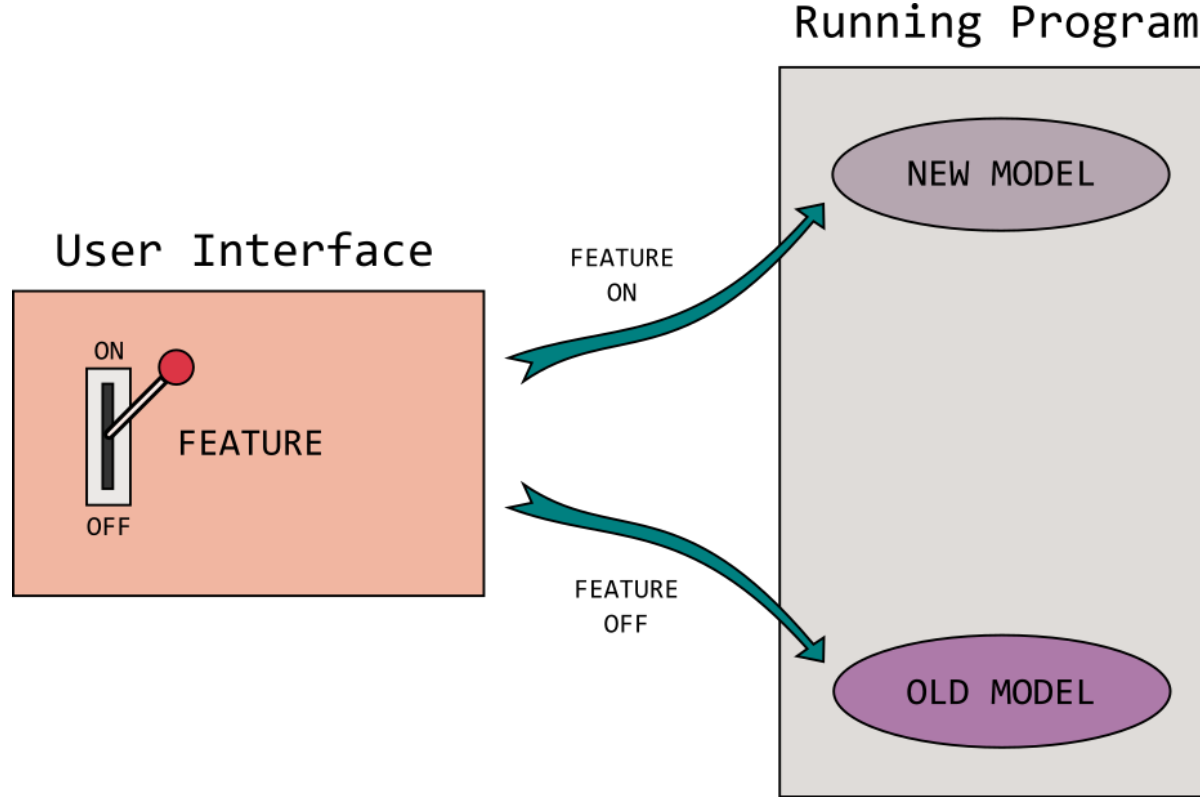


Wielkie możliwości małej
biblioteki Togglz

Opis przypadku

Czym jest wzorzec Feature Toggle?



<https://martinfowler.com/articles/feature-toggles.html>

Po co wymyślać koło na nowo...



<https://www.togglz.org/>

```
<dependencies>
  //...
  <dependency>
    <groupId>org.togglz</groupId>
    <artifactId>togglz-spring-boot-starter</artifactId>
    <version>3.1.2</version>
  </dependency>
  <dependency>
    <groupId>org.togglz</groupId>
    <artifactId>togglz-console</artifactId>
    <version>3.1.2</version>
  </dependency>
</dependencies>
</project>
```

```
enum PoetFeature implements Feature {  
  
    @EnabledByDefault  
    @Label("First verse")  
    FIRST_VERSE("Nic dwa razy się nie zdarza"),  
  
    @EnabledByDefault  
    @Label("Second verse")  
    SECOND_VERSE("i nie zdarzy. Z tej przyczyny"),  
  
    @EnabledByDefault  
    @Label("Third verse")  
    THIRD_VERSE("zrodziliśmy się bez wprawy"),  
  
    @EnabledByDefault  
    @Label("Fourth verse")  
    FOURTH_VERSE("i pomrzemy bez rutyny.");  
  
    private final String verse;  
  
    PoetFeature(String verse) {  
        this.verse = verse;  
    }  
}
```



```
togglz.console.secured=false
```

```
togglz.feature-enums=pl.devcezz.togglz.PoetFeature
```





```
@RestController
@RequestMapping("/feature")
class FeatureToggleController {

    private final FeatureManager featureManager;

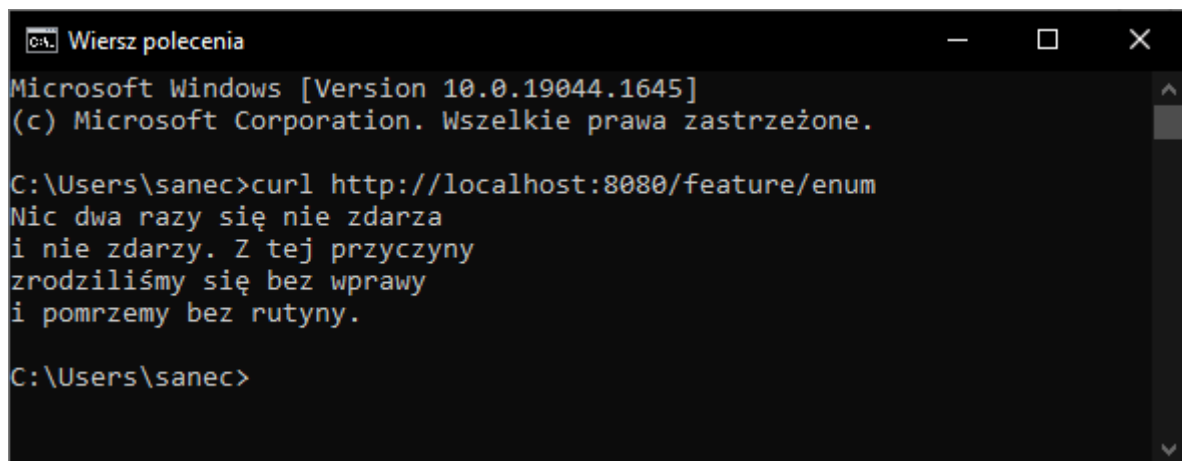
    @GetMapping("/enum")
    String checkEnumFeature() {
        StringBuilder builder = new StringBuilder();
        if (featureManager.isActive(PoetFeature.FIRST_VERSE)) {
            builder.append(PoetFeature.FIRST_VERSE.getVerse() + '\n');
        }
        if (featureManager.isActive(PoetFeature.SECOND_VERSE)) {
            builder.append(PoetFeature.SECOND_VERSE.getVerse() + '\n');
        }
        if (featureManager.isActive(PoetFeature.THIRD_VERSE)) {
            builder.append(PoetFeature.THIRD_VERSE.getVerse() + '\n');
        }
        if (featureManager.isActive(PoetFeature.FOURTH_VERSE)) {
            builder.append(PoetFeature.FOURTH_VERSE.getVerse() + '\n');
        }
        return builder.toString();
    }
}
```

Togglz application

All Features

Feature	Status	Strategy	Actions
First verse	Enabled		
Second verse	Enabled		
Third verse	Enabled		
Fourth verse	Enabled		

Togglz 3.1.2
www.togglz.org
Apache Tomcat/9.0.62



```
Microsoft Windows [Version 10.0.19044.1645]  
(c) Microsoft Corporation. Wszelkie prawa zastrzeżone.  
  
C:\Users\sanec>curl http://localhost:8080/feature/enum  
Nic dwa razy się nie zdarza  
i nie zdarzy. Z tej przyczyny  
zrodziliśmy się bez wprawy  
i pomrzemy bez rutyny.  
  
C:\Users\sanec>
```

```
<dependencies>
  //...
  <dependency>
    <groupId>org.togglz</groupId>
    <artifactId>togglz-junit</artifactId>
    <version>3.1.2</version>
    <scope>test</scope>
  </dependency>
</dependencies>
</project>
```

```
class FeatureToggleControllerTest {  
  
    private final TestFeatureManager featureManager =  
        new TestFeatureManager(PoetFeature.class);  
    private final FeatureToggleController controller =  
        new FeatureToggleController(featureManager);  
  
    @Test  
    void should_print_poem_without_first_verse() {  
        featureManager.enableAll();  
        featureManager.disable(PoetFeature.FIRST_VERSE);  
  
        String result = controller.checkEnumFeature();  
  
        assertEquals("""  
            i nie zdarzy. Z tej przyczyny  
            zrodziliśmy się bez wprawy  
            i pomrzemy bez rutyny.  
            """, result);  
    }  
}
```

Podsumowanie