

Programowanie w JAVA

Lab. 5 – Testy jednostkowe

1. **Cel zadania:** Uzupełnij jeden z projektów pisanych na poprzednich zajęciach (Lab 2, 3 lub 4) zestawem testów jednostkowych oraz systemem raportowym. Sugestia: najprościej będzie rozbudować Lab 2.
2. **Przygotowanie projektu maven:**

Zainstaluj plugin Cobertura. Do pliku pom.xml dodaj następujący wpis:

```
<project>
. . .
<reporting>
  <plugins>
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-project-info-reports-plugin</artifactId>
      <version>2.7</version>
      <configuration>
        <dependencyLocationsEnabled>>false</dependencyLocationsEnabled>
      </configuration>
    </plugin>
    <plugin>
      <groupId>org.codehaus.mojo</groupId>
      <artifactId>cobertura-maven-plugin</artifactId>
      <version>2.6</version>
      <configuration>
        <formats>
          <format>html</format>
          <format>xml</format>
        </formats>
        <instrumentation>
          <excludes>
            <exclude>com/intellij/**/*.*.class</exclude>
          </excludes>
        </instrumentation>
      </configuration>
    </plugin>
  </plugins>
</reporting>
. . .
</project>
```

Zainstaluj JUnit w wersji 4>.

```
<dependency>
  <groupId>junit</groupId>
  <artifactId>junit</artifactId>
  <version>4.12</version>
  <scope>test</scope>
</dependency>
```

3. **Jak korzystać:**
 - a. Po zainstalowaniu pluginu, uruchomienie targetu mvn site generuje statyczną stronę zawierającą dokumentację projektu, w tym Coverage Report
 - b. Raport pokazuje ilość kodu pokrytą testami „Line Coverage”

- c. Testy tworzymy poprzez kliknięcie prawym przyciskiem myszy na nazwę danej klasy lub metody i wybór z menu kontekstowego „Create <nazwa>Test...”

4. Zadania:

- a. Zapewnij co najmniej 80% Line Coverage (<60% = -1pkt; <40% = -2pkt; <20% = -3pkt)
- b. Napisz co najmniej po jednym teście, który:
 - i. Sprawdza zachowanie funkcji dla poprawnych danych
 - ii. Sprawdza zachowanie funkcji dla niepoprawnych danych
 - iii. Sprawdza, czy funkcja rzuca wyjątek dla odpowiednich danych (jeśli nie zaimplementowano takiej funkcji, należy ją dodać / zmodyfikować istniejącą)
 - iv. Każdy test musi wykorzystywać conajmniej jedną z metod klasy Assert
 - v. W zadaniu należy przetestować co najmniej 10 funkcji/metod