

3. Treća laboratorijska vježba

3.1. Iznimke u Javi

Svrha laboratorijske vježbe je upoznavanje i različitih vrsta iznimaka koje postoje u Javi, način na koji se one obrađuju i bacaju u programskom kodu. Također će se vidjeti kako je moguće kreirati vlastite iznimke, određivanje njihovih tipova i način kreiranja objekata. Osim toga će se unutar vježbe koristiti i vanjska biblioteka „Logback“ koju će biti potrebno dohvatiti preko „Maven“ sustava, a služi za kreiranje logova o radu programa.

3.2. Zadatak za pripremu

Nastaviti razvoj aplikacije iz druge laboratorijske vježbe i implementirati koncepte korištenja, bacanja i hvatanja iznimaka kroz sljedeće korake, uz obvezno poštivanje smjernica iz napomena na kraju pripreme:

1. Iskopirati projekt iz druge laboratorijske vježbe i kopiju preimenovati u naziv koji sadrži redni broj vježbe „3“.
2. Kreirati novi paket pod nazivom „hr.java.vjezbe.iznimke“.
3. Unutar paketa „hr.java.vjezbe.iznimke“ kreirati dvije klase:
„NekompatibilnoSuceljeZaProcesorException“ koja mora označavati označenu iznimku i „NeispravanBrojMemorijskihModulaException“ koja mora označavati neoznačenu iznimku. Obje klase moraju sadržavati tri različita konstruktora koji primaju sve tri kombinacije parametara „message“ i „cause“. Nakon kreiranja klase upozorenje koje se javlja porukom npr. „The serializable class NekompatibilnoSuceljeZaProcesorException does not declare a static final serialVersionUID field of type long“ ispraviti odabirom opcije „Add generated serial version ID“.
4. Klasu „MaticnaPloca“ proširiti s dva nova parametra: tip sučelja za procesor (za koji je potrebno koristiti konstantne vrijednosti iz klase „Procesor“) i maksimalan broj memorijskih modula. Te parametre je potrebno dodati u konstruktor, generirati „getter“ i „setter“ metode te ih zatražiti od korisnika kod unošenja ostalih podataka.
5. Logiku za unos tipa sučelja procesora doraditi tako da se izdvoji u zasebnu metodu i poziva na mjestima za unos tipa sučelja za procesor na matičnoj ploči i tipa sučelja samog procesora. U slučaju unosa tipova koji nisu kompatibilni potrebno je baciti iznimku „NekompatibilnoSuceljeZaProcesorException“ i zatražiti ponovni unos podataka o procesoru od korisnika. Provjeru kompatibilnosti također izdvojiti u

zasebnu metodu koja se poziva iz „main“ metode i iz te zasebne metode bacati iznimku, a hvatati je u „main“ metodi i prema tome odlučiti treba li korisnik ponovno unositi sučelja za procesor ili ne.

6. Na sličan način kao i za provjeru kompatibilnosti sučelja procesora i matične ploče potrebno je provjeriti je li korisnik unio ispravan broj memorijskih modula za matičnu ploču. Prvo je potrebno unijeti maksimalan broj memorijskih modula koje je moguće ugraditi na matičnu ploču, a nakon toga unijeti aktualni broj koji je korisnik odabrao. Ako nije, potrebno je baciti iznimku „NeispravanBrojMemorijskihModulaException“ i zatražiti novi unos. Također je tu provjeru potrebno ubaciti u zasebnu metodu iz koje se baca iznimka, a u „main“ metodi se ona hvala i obrađuje.
7. Aplikaciju je potrebno proširiti resursima prema kojima će se ugraditi „Logback“ biblioteka pomoću „Maven“ repozitorija i omogućiti generiranje „log“ datoteka. Koristiti upute s predavanja i napomena ove laboratorijske vježbe. Kod „importanja“ klasa vezanih za „Logback“ potrebno je odabrati sljedeće stavke:

```
import org.slf4j.Logger;  
import org.slf4j.LoggerFactory;
```

8. U aplikaciji ugraditi „logiranje“ za sve važnije događaje, a posebno za ispis podataka o iznimkama.
9. Javadoc dokumentacijom dokumentirati klase i metode koje se koriste u aplikaciji.

Primjer izvođenja programskog koda:

```
Unos podataka prvog računala:  
Unesite proizvođača matične ploče računala: Asrock  
Unesite tip matične ploče računala:  
Unesite tip sučelja procesora (odaberite broj ispred željenog sučelja):  
1) SOCKET_AM2  
2) SOCKET_AM3  
3) SOCKET_LGA_1151  
4) SOCKET_G3  
3  
Unesite maksimalni broj memorijskih modula: 4  
Unesite broj memorijskih modula koji želite ugraditi na matičnu ploču: 10  
Pogreška! Unijeli ste neispravan broj memorijskih modula. Matična ploča  
podržava između 1 i 4 memorijskih modula!  
Unesite broj memorijskih modula koji želite ugraditi na matičnu ploču: 4  
Unesite proizvođača procesora računala: Intel  
Unesite tip procesora računala: Celeron G1840  
1) SOCKET_AM2  
2) SOCKET_AM3  
3) SOCKET_LGA_1151
```

```
4) SOCKET_G3
10
Pogreška kod odabira, molimo pokušajte ponovno!
1) SOCKET_AM2
2) SOCKET_AM3
3) SOCKET_LGA_1151
4) SOCKET_G3
4
Pogreška! Tip sučelja procesora mora se podudarati s tipom sučeljaza
procesor na matičnoj ploči!
1) SOCKET_AM2
2) SOCKET_AM3
3) SOCKET_LGA_1151
4) SOCKET_G3
1
Pogreška! Tip sučelja procesora mora se podudarati s tipom sučeljaza
procesor na matičnoj ploči!
1) SOCKET_AM2
2) SOCKET_AM3
3) SOCKET_LGA_1151
4) SOCKET_G3
2
Pogreška! Tip sučelja procesora mora se podudarati s tipom sučeljaza
procesor na matičnoj ploči!
1) SOCKET_AM2
2) SOCKET_AM3
3) SOCKET_LGA_1151
4) SOCKET_G3
3
Unesite brzinu procesora računala (u GHz): 2,8
Unesite proizvođača radne memorije računala: Corsair
Unesite tip radne memorije računala: DDR3 1600 MHz
Unesite kapacitet radne memorije računala (u GB): 4
Unesite proizvođača tvrdog diska računala: Seagate
Unesite tip tvrdog diska računala: Barracuda
Unesite kapacitet tvrdog diska računala (u TB): 4
Unos podataka drugog računala:
Unesite proizvođača matične ploče računala: Asrock
Unesite tip matične ploče računala: Z97 KILLER
Unesite tip sučelja procesora (odaberite broj ispred željenog sučelja):
1) SOCKET_AM2
2) SOCKET_AM3
3) SOCKET_LGA_1151
4) SOCKET_G3
4
Unesite maksimalni broj memorijskih modula: 4
Unesite broj memorijskih modula koji želite ugraditi na matičnu ploču: 4
Unesite proizvođača procesora računala: Intel
Unesite tip procesora računala: i5-4690K
1) SOCKET_AM2
2) SOCKET_AM3
3) SOCKET_LGA_1151
4) SOCKET_G3
4
Unesite brzinu procesora računala (u GHz): 3,9
Unesite proizvođača radne memorije računala: Kingston
Unesite tip radne memorije računala: HyperX Savage DDR3 2400MHz
Unesite kapacitet radne memorije računala (u GB): 8
Unesite proizvođača tvrdog diska računala: Western Digital
Unesite tip tvrdog diska računala: Caviar Green WD20EZR
```

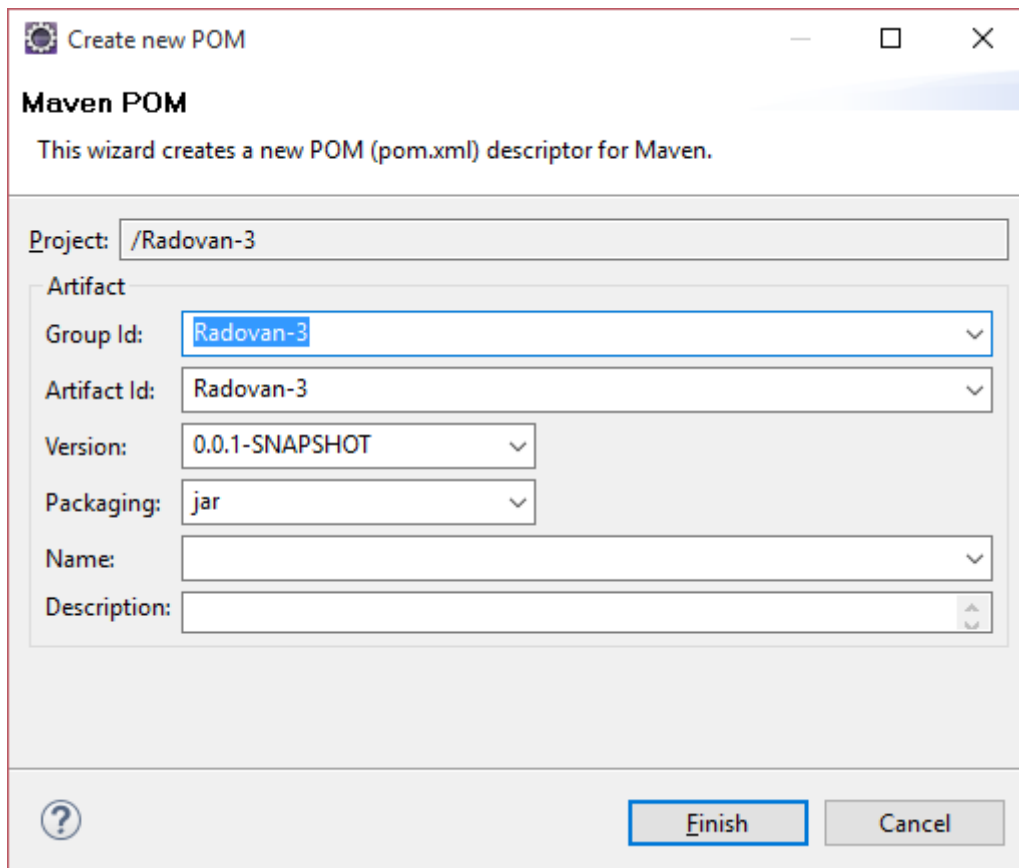
```
Unesite kapacitet tvrdog diska računala (u TB): 2
Unesena konfiguracije računala su sljedeće:
Prvo računalo:
Proizvođač matične ploče: Asrock
Tip matične ploče: scanner.nextLine();
Proizvođač procesora: Intel
Tip procesora: Celeron G1840
Tip sučelja procesora: SOCKET_LGA_1151
Brzina procesora: 2.8 GHz
Proizvođač radne memorije: Corsair
Tip radne memorije: DDR3 1600 MHz
Kapacitet radne memorije: 0.00390625 TB
Proizvođač tvrdog diska: Seagate
Tip tvrdog diska: Barracuda
Kapacitet tvrdog diska: 4 TB
Drugo računalo:
Proizvođač matične ploče: Asrock
Tip matične ploče: Z97 KILLER
Proizvođač procesora: Intel
Tip procesora: i5-4690K
Tip sučelja procesora: SOCKET_G3
Brzina procesora: 3.9 GHz
Proizvođač radne memorije: Kingston
Tip radne memorije: HyperX Savage DDR3 2400MHz
Kapacitet radne memorije: 0.0078125 TB
Proizvođač tvrdog diska: Western Digital
Tip tvrdog diska: Caviar Green WD20EZR
Kapacitet tvrdog diska: 2 TB
Nakon promjena, konfiguracije računala su sljedeće:
Prvo računalo:
Proizvođač matične ploče: Asrock
Tip matične ploče: scanner.nextLine();
Proizvođač procesora: Intel
Tip procesora: Celeron G1840
Tip sučelja procesora: SOCKET_LGA_1151
Brzina procesora: 4.20 GHz
Proizvođač radne memorije: Corsair
Tip radne memorije: DDR3 1600 MHz
Kapacitet radne memorije: 0.0078125 TB
Proizvođač tvrdog diska: Seagate
Tip tvrdog diska: Barracuda
Kapacitet tvrdog diska: 4 TB
Drugo računalo:
Proizvođač matične ploče: Asrock
Tip matične ploče: Z97 KILLER
Proizvođač procesora: Intel
Tip procesora: i5-4690K
Tip sučelja procesora: SOCKET_G3
Brzina procesora: 3.9 GHz
Proizvođač radne memorije: Kingston
Tip radne memorije: HyperX Savage DDR3 2400MHz
Kapacitet radne memorije: 0.0078125 TB
Proizvođač tvrdog diska: Western Digital
Tip tvrdog diska: Caviar Green WD20EZR
Kapacitet tvrdog diska: 2 TB
```

NAPOMENE:

1. Usporedbu jednakosti Stringova obavljati korištenjem metode „equals“.

2. Kod ponovnog unošenja vrijednosti nakon pogrešnog unosa koristiti metodu „nextLine“ u klasi „Scanner“ kako bi se izbjeglo iskorištavanje neispravno unesene vrijednosti.
3. Kako bi se dodala vanjska zavisnost pomoću „Maven“-a, potrebno je u projektu kreirati pom.xml datoteku.

Iz Eclipse razvojnog sučelja to je najjednostavnije napraviti desnim klikom na projekt *Configure -> Convert to Maven project* nakon čega Eclipse automatski kreira „pom.xml“ datoteku (za „Name“ i „Description“ parametre nije potrebno ništa definirati).



Slika 1. Kreiranje „pom.xml“ datoteke

Kada je datoteka kreirana, potrebno je u istu dodati novu ovisnost (engl. *Dependency*). Ovisnosti je najjednostavnije pronaći na web stranici <http://mvnrepository.com/>. Nakon što je pronađena datoteka koju je potrebno ukomponirati u projekt, s web stranice je potrebno kopirati dio pod oznakom „maven“ i prebaciti u „pom.xml“ datoteku.

Home » ch.qos.logback » logback-classic » 1.1.3

Logback Classic Module » 1.1.3

**Logback Classic Module**
ch.qos.logback » logback-classic » 1.1.3 under **Logging Frameworks**
logback-classic module

5,441 usages

Artifact	Download (JAR) (275 KB)
POM File	View
Date	(Mar 24, 2015)

MavenIvyGrapeGradleBuildrSBTLeiningen

```
<dependency>
  <groupId>ch.qos.logback</groupId>
  <artifactId>logback-classic</artifactId>
  <version>1.1.3</version>
</dependency>
```

Slika 2. Pretraga Maven repozitorija

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>Radovan-3</groupId>
  <artifactId>Radovan-3</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <dependencies>
    <dependency>
      <groupId>ch.qos.logback</groupId>
      <artifactId>logback-classic</artifactId>
      <version>1.1.3</version>
    </dependency>
  </dependencies>
  <build>
    <sourceDirectory>src</sourceDirectory>
    <plugins>
      <plugin>
        <artifactId>maven-compiler-plugin</artifactId>
        <version>3.3</version>
        <configuration>
          <source>1.8</source>
          <target>1.8</target>
        </configuration>
      </plugin>
    </plugins>
  </build>
</project>
```

Programski isječak 1. Datoteka pom.xml nakon dodavanja ovisnosti

MOGUĆNOSTI UNAPREĐENJA ZADATKA:

1. Kod svakog „try-catch“ bloka staviti i „finally“ blok koji sadrži programski kod koji ispisuje podatke o neispravnoj konfiguraciji računala koja je unesena.

2. Cijeli proces unošenja podataka o konfiguraciji računala staviti u jedan „try-catch“ blok koji pomoću „multicatch“ bloka hvata sve iznimke koje se mogu dogoditi i u slučaju bacanja iznimke ponavlja se unos cijele konfiguracije računala.