

# Tutorijal 3

## Zadatak 1

Cilj vježbe je napraviti aplikaciju "**ETF Telefonski imenik**". Aplikacija se treba sastojati od sljedećih klasa:

- Apstraktna klasa **TelefonskiBroj** sadrži *javnu apstraktnu* metodu **String ispisi()** koja vraća broj u tekstualnoj formi pogodnoj za ispis, te *apstraktnu metodu* **int hashCode()** objašnjenu na predavanjima. Iz ove klase izvedene su sljedeće klase:
  - **FiksniBroj** sadrži *konstruktor* **FiksniBroj(Grad grad, String broj)**. Parametar broj je dio telefonskog broja bez pozivnog broja npr. "123-456", a grad je promjenljiva pobrojanog tipa koji označava pozivni broj koji treba koristiti npr.: **enum Grad { SARAJEVO, TUZLA, ZENICA...}** Spisak pozivnih brojeva se može naći [ovdje](#). Umjesto imena kantona koristite ime glavnog grada kantona (kao u primjeru iznad), a umjesto Brčko distrikta stavite BRCKO. Metoda ispisi() treba vratiti broj oblika "033/123-456".
  - **MobilniBroj** ima *konstruktor* **MobilniBroj(int mobilnaMreza, String broj)** pri čemu oznaka mobilne mreže predstavlja drugu i treću cifru pozivnog broja tj. 60-67 (detaljnije informacije možete naći na [istoj stranici](#)). Metoda ispisi vraća broj istog oblika "061/987-654".
  - **MedunarodniBroj** ima *konstruktor* **MedunarodniBroj(String drzava, String broj)** pri čemu string drzava sadrži kompletan pozivni broj za državu npr. "+387".
- Klasa **Imenik** omogućuje da se pretražuju brojevi koristeći klasu **HashMap**. Ova klasa treba sadržavati sljedeće javne metode:
  - **void dodaj(String ime, TelefonskiBroj broj)**
  - **String dajBroj(String ime)** - vraća telefonski broj osobe pod imenom ime u formi stringa pozivajući metodu **ispisi()** klase **TelefonskiBroj**,
  - **String dajIme(TelefonskiBroj broj)** - vraća ime osobe čiji telefonski broj je broj - i ova operacija treba koristiti **HashMap**,
  - **String naSlovo(char s)** - vraća sve brojeve u telefonskom imeniku za osobe čije ime počinje na slovo s u formatu:

1. Ime Prezime - broj
2. Ime Prezime - broj

...

- **Set<String> izGrada(Grad g)** - vraća skup Stringova koji sadrži imena i prezimena svih osoba koje žive u gradu g (istog pobrojanog tipa koji je dat u klasi FiksniBroj), što se može saznati ako osoba ima fiksni broj telefona u tom gradu. Pri tome skup treba biti sortiran abecedno.
- **Set<TelefonskiBroj> izGradaBrojevi(Grad g)** - vraća brojeve za osobe iz grada g. Ovaj skup treba biti sortiran po stringu koji se dobije metodom ispisi().
- Klasa **Program** treba sadržavati main metodu koja korisniku omogućuje da unosom naredbi pomoću tastature pozove sve metode klasa navedenih iznad.

**Potrebno je kreirati vlastiti tip izuzetka i iskoristiti ga.**

Prilikom rješavanja zadatka obavezno se pridržavati pravila rada koja su navedena na predavanjima, između ostalog:

- ispravno imenovanje paketa klasa, metoda i promjenljivih
- primjenu komentara
- atomic commits.

## Zadatak 2

Napisati minimalno 3 JUnit testa za prethodni zadatak. U jednom od ta 3 testa ubaciti i testirati neku metodu sa izuzetkom kojeg ste vi napravili.

## Zadatak 3

Napisati test vezan za Mocking testiranje. U prvom testu koristit ćete Mapu čiji će i ključ i vrijednost biti tipa **String**.

**Pokušati napisati drugi test tako da imate 2 klase, baznu i izvedenu. Bazna klasa bi trebala imati privatni atribut tipa izvedene klase, a izvedena klasa treba imati**

**jednu public metodu koja će se testirati. Nađite način testiranja metode izvedene klase preko metode bazne klase.**