

# Rad u Javi

Kada kucamo program, često pravimo greške...

Na primer, ako u prvom redu programa umesto

```
class Pozdrav
```

```
otkucamo
```

```
Class Pozdrav
```

u pitanju je sintaksna (“gramatička”) greška – kompajler na početku programa očekuje rezervisanu reč class (sa malo “c” na početku).

- Kada kompajlirate program sa greškom, kompajler javlja grešku, sa kratkim opisom gde u programu je greška primećena.

- Ako u programu umesto

```
System.out.println("Pozdrav svima!");
```

napišemo

```
System.out.println("Pozdrav ##ima!");
```

program nema sintaksnu grešku, kompajliranje i izvršavanje će proći bez javljanja greške, ali program i dalje ne radi ono što hoćemo.

Ovakve greške se nazivaju i semantičke greške, ili bagovi (engl. bug).

- Proces programiranja prirodno podrazumeva višestruki prolazak kroz ciklus “*editovanje programa – kompajliranje – izvršavanje*” mnogo puta, sve dok program ne bude zadovoljavajući, bez sintaksnih i semantičkih grešaka.

# Osnovni pojmovi objektno-orijentisanog programiranja

- Tipovi podataka u Javi: Apstraktno govoreći, *tip podataka* je način prikaza podataka određene vrste (tipa).
- Osnovni (primitivni) tipovi obuhvataju podatke fundamentalnog (osnovnog) tipa.
- Podaci u Javi mogu biti *primitivni podaci* (*podaci primitivnog tipa*) ili *objekti*.
- Tip objekta se naziva *klasa*.
- Analogija – šraf, matica, ekser (primitivni), mašina (objekat).

# Osnovni pojmovi objektno-orijentisanog programiranja

- Postoji samo osam primitivnih tipova podataka, ali i hiljade klasa objekata koje su standardno predefinisane u Javi.
- Takođe, programer sam može da definiše nove klase.
- Rekli smo, klasa je opis objekta. Ona opisuje od čega se objekat sastoji, koje podatke sadrži, i spisak *metoda* objekta.
- Najčešće ćemo kreirati više objekata jedne klase...
- Analogija sa primitivnim tipovima.

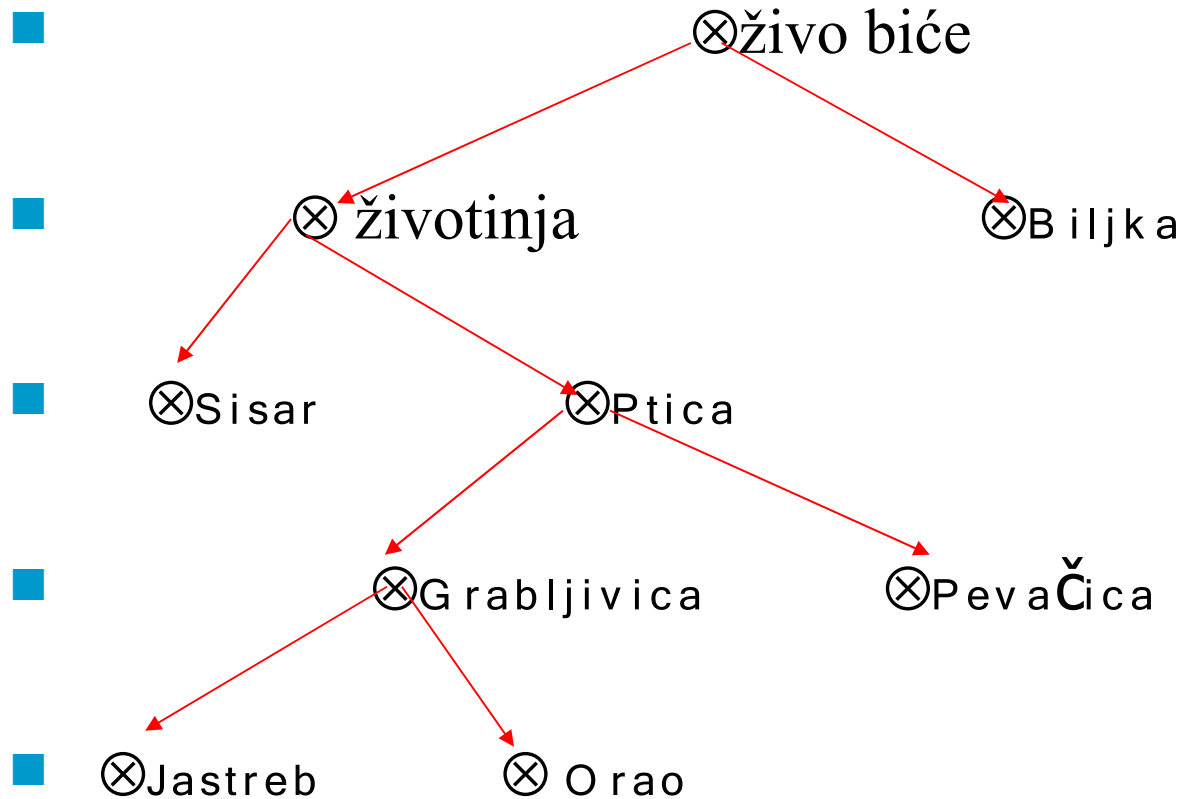
# Osnovni pojmovi objektno-orijentisanog programiranja

- Objekat - integralna celina podataka i procedura za rad sa njima. Zbog prisustva procedura u objektima, objekti imaju mogućnost da samostalno deluju, tj. postaju dinamički.
- Objektno-orijentisano programiranje - programska paradigma zasnovana na skupu objekata koji dejstvuju međusobno. Glavne obrade zasnivaju se na manipulisanju sa objektima.
- Metod - funkcija koja je sastavni deo objekta, tj. postupak kojim se realizuje poruka upućena objektu.
- Poruka - skup informacija koji se šalje objektu. Sastoji se iz adrese (objekta primaoca poruke) i saopštenja.

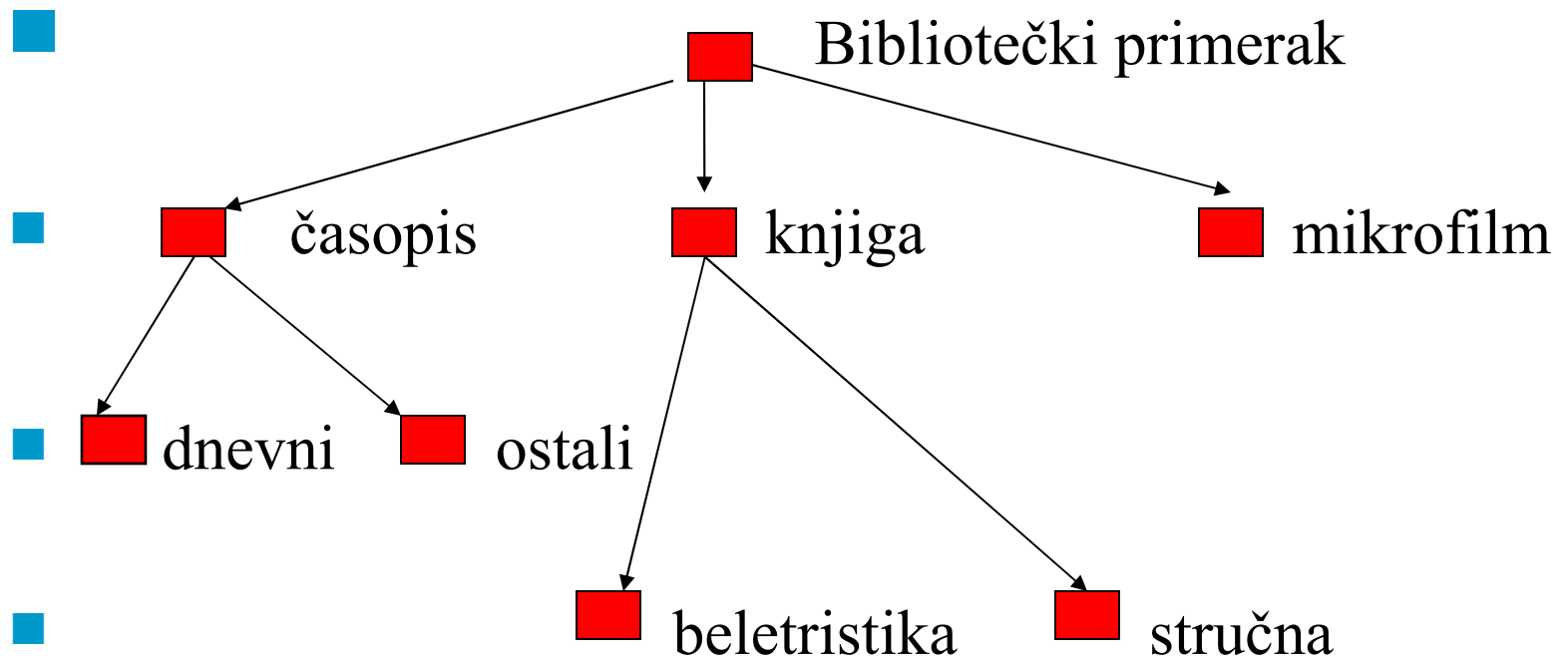
- Klasa - Tip objekata sa zajedničkim svojstvima (koji se ponašaju na isti način). Definiše šablon za kreiranje instanci.
- Primerak (instanca) - konkretan objekat iz klase. Skup instanci sa skupom metoda kreira jednu klasu.
- Klasa B je potklasa klase A ako su svi primerci klase B istovremeno i primerci klase A. Za klasu kažemo da je natklasa klase B. Potklase nastaju dodavanjem novih svojstava (metoda) postojećoj klasi.
- Nasleđivanje - mehanizam za kreiranje novih klasa iz postojećih. Nasleđivanjem se formiraju relacije između jedne i više drugih klasa.
- Polimorfizam - Mogućnost primene istog metoda (operatora) na primerke različitih klasa.

## 2.1. Primeri nasleđivanja

Prilikom projektovanja programa uočavaju se veze između pojedinih klasa i način povezivanja tih klasa sa drugim klasama. Ovdje je bitna uloga nasleđivanja.

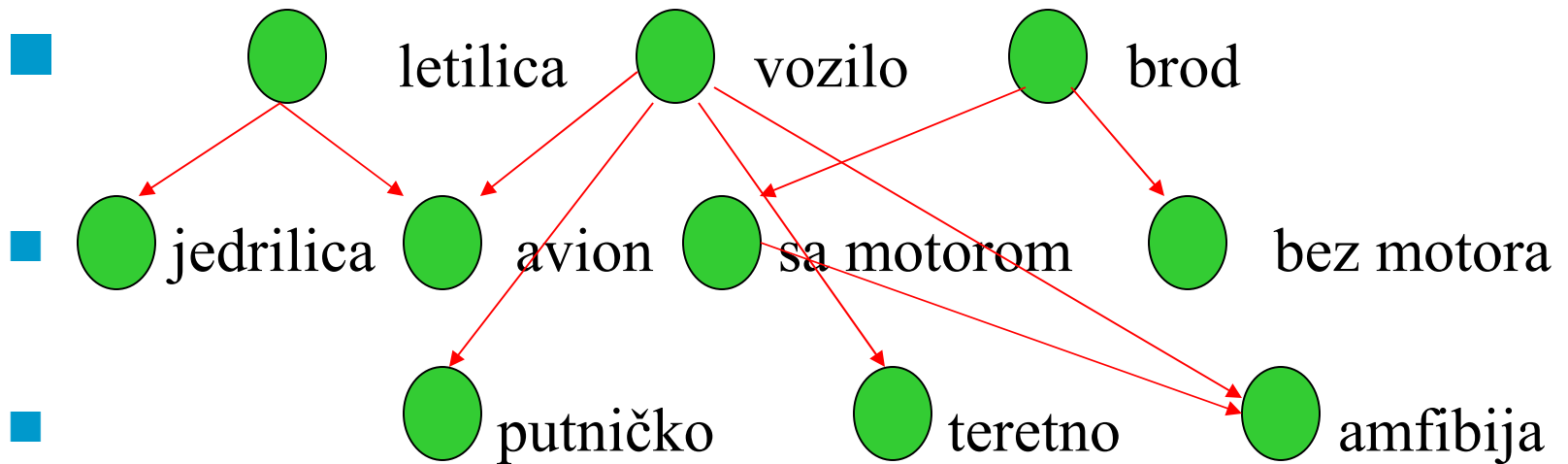


# Jos jedan primer hijerarhijskog nasleđivanja





# Primer višestrukog nasleđivanja (klasa može imati više direktnih natklasa)



V išestruko nasleđivanje nije podržano u Java-jeziku, ali jeste u C++.

## Još neka svojstva klasa

- Klase su organizovane po PAKETIMA. Osnovni paketi su: `java.lang`, `java.util`, `java.io`, `java.net`, `java.awt`, `java.applet`.
- Paket je skup klasa namenjenih jednoj vrsti posla (koje čine srodnu celinu)
- Sve klase iz paketa čine BIBLIOTEKU KLASA.
- Sve aplikacije i apleti se realizuju pomoću (preko) KLASA.
- Za svaku klasu su bitne 2 karakteristike:
  - stanje (opisuje se preko promenljivih klase I promenljivih instance)
  - ponašanje njenih instanci (određuje se preko metoda.

# Objekti u Java-jeziku

Objekat je potpuno određen svojim atributima i ponašanjem.

- Atributi objekta – promenljive instanci. Klasom je definisan njihov tip, a za svaku instancu, promenljiva instanci ima konkretnu vrednost.

Primeri:

- tacka (x,y, boja)
- automobil(broj sasijske, broj motora, boja, broj sedišta)

- Ponašanje objekta određeno je metodima u klasi koji mogu dejstvovati na taj objekat. Primeri:

- Uzeti x-koordinatu, pomeriti tačku za dx, dy, ...
- Startovati motor, promeniti brzinu, zaustaviti motor, ...

# Kontrolna pitanja

- 8. Koja je razlika između sintakasnih i semantičkih grešaka?
- 9. Šta je klasa?
- 10. Šta je objekat?
- 11. Šta su potklase?
- 12. Napišite jedan primer nasleđivanja (uključujući i sadržaj klase).
- 13. Šta su paketi i biblioteka klase?