

1. Osnovni elementi Java programa

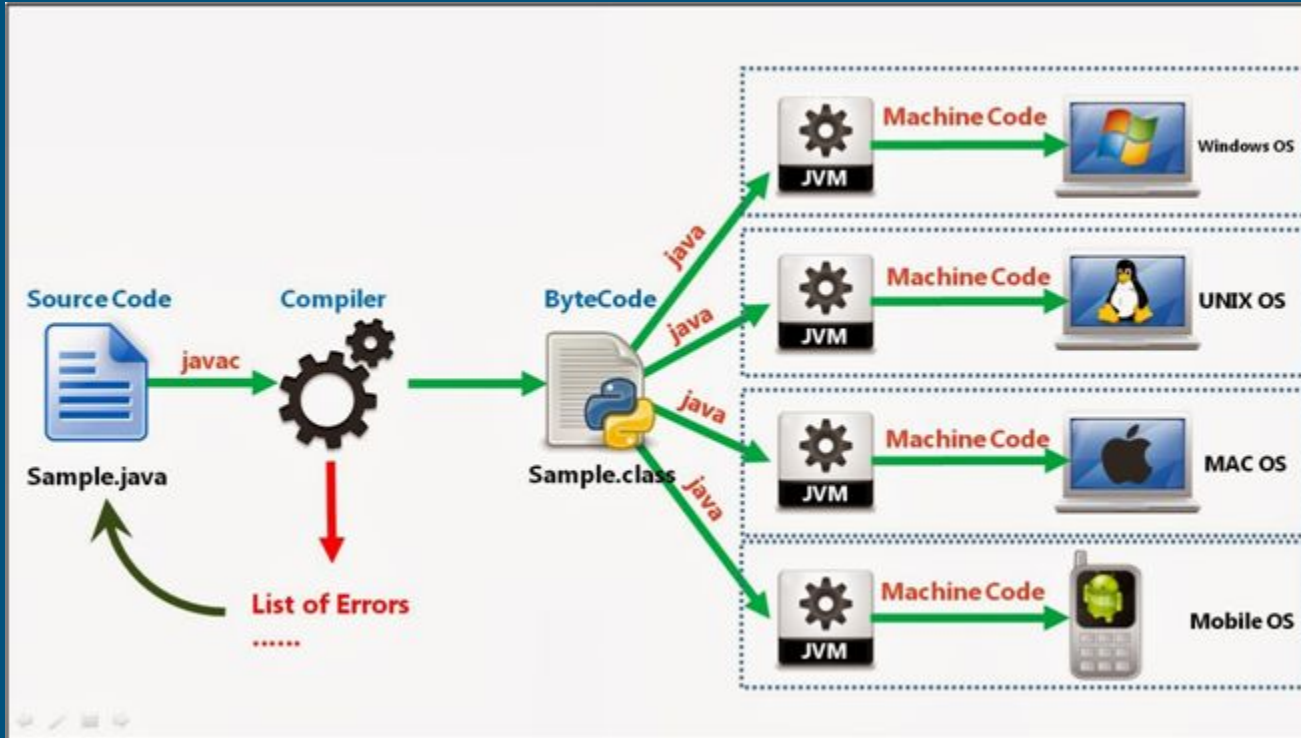
Sadržaj

- Osobine programskog jezika Java
- Struktura Java programa
- Tipovi podatka i promenljive
- Eclipse *Integrated Development Environment* (IDE)

Osobine programskoj jezika Java

- Objektno-orijentisan programski jezik
- Portabilni programi
- Bogate biblioteke klasa
- Namenjen za različite arhitekture softverskih rešenja
 - Standard Edition (SE) - desktop aplikacije
 - Enterprise Edition (EE) - višeslojne aplikacije
 - Micro Edition (ME) - aplikacije za mobilne uređaje sa slabijim hardverskim resursima

Java kompajler i interpreter



HelloWorld program u Javi

```
public class HelloWorld {  
    public static void main(String[] args) {  
        System.out.println("Hello, World!");  
    }  
}
```

Tip podatka

- Skup vrednosti koje imaju neke zajedničke karakteristike
- Skup operacija definisanih nad tim skupom vrednosti
- Primer:
 - tip podatka: celi brojevi
 - operacije: sabiranje, oduzimanje, množenje, deljenje

Primitivni tipovi podataka

- Osnovni primitivni tipovi podataka
 - `int` - celobrojne vrednosti
 - `boolean` - logičke vrednosti
 - `double` - decimalne vrednosti
 - `char` - znakovi (karakter)

Tip podatka `int`

- Za memorisanje `int` vrednosti koriste se 32 bita
- Opseg vrednosti od -2 147 483 648 do 2 147 283 647
- Podrazumevana (*default*) vrednost je 0
- Java omogućava korišćenje i drugih celobrojnih tipova podataka sa drugačijim opsezima vrednosti:
 - `byte` (8 bita)
 - `short` (16 bita) i
 - `long` (64 bita)
- Prilikom izvršenja aritmetičkih operacija u izrazima dolazi do implicitne konverzije u `int`

Tip podatka `boolean`

- Moguće dve vrednosti: `true` i `false`
- Podrazumevana vrednost je `false`
- Najčešće korišćene logičke operacije su “i” (`&&`), “ili” (`||`) i “not” (`!`)
- “Not” je unarni operator koji menja vrednost iz `true` u `false` i obrnuto

A	B	A && B	A B
false	false	false	false
false	true	false	true
true	false	false	true
true	true	true	true

Tip podatka `double`

- Za memorisanje `double` vrednosti koriste se 64 bita
- Decimalni brojevi dvostruke preciznosti sa pokretnim zarezom
- Podrazumevana (*default*) vrednost je `0.0d`
- Java omogućava korišćenje tipa podatka “jednostruke” preciznosti
 - `float` (32 bita)

Tip podatka `char`

- Za memorisanje `char` vrednosti koristi se 16 bita
- Skup vrednosti su *unicode* karakteri
- Literali se navode između apostrofa
- Primeri vrednosti:
 - `'a'`
 - `'2'`
 - `'\n'`
- Za nizove znakova se koristi klasa `String`
- Podrazumevana vrednost je `'\u0000'`

Složeni tipovi podataka

- Pored postojećih primitivnih tipova, Java omogućava definisanje složenih tipova podataka
- Nizovi - upravljanje nad više vrednosti istog tipa podatka
- Klase - složeni tip podataka čiju strukturu definiše programer

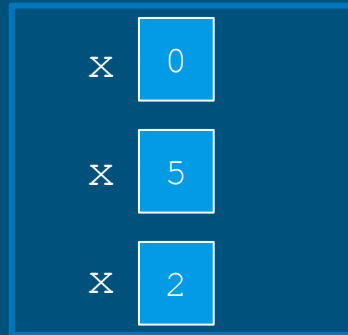
Pojam promenljive

- Imenovani deo memorije
- Predstavlja vezu sa memorijskom lokacijom
- Tip podatka promenljive određuje vrednosti koje mogu da joj se dodele

```
int x;
```

```
x = 5;
```

```
x = 2;
```



Promenljive u Javi

- Moraju se eksplicitno deklarirati
- Mora sadržati vrednost deklarisanog tipa podatka
- Primeri:

```
int visina = 186;
```



```
char pol = false;
```



```
tezina = 88;
```



Imenovanje promenljivih

- Naziv promenljive u Javi mora početi:
 - slovom,
 - znakom _ ili
 - znakom \$
- Naziv promenljive ne može biti rezervisana reč jezika
- Preporučena konvencija imenovanja:
 - počinju malim slovom
 - svaka sledeća reč u nazivu počinje velikim slovom
 - ne počinju znakom _ ili \$ iako su dozvoljeni
- Naziv promenljive treba da ukazuje na njenu namenu

Klasa `String` (1)

- Specifična klasa u programskom jeziku Java
- Omogućava rad sa nizovima znakova (prevazilazi nedostatke `char` tipa)
- Naziv klase po konvenciji imenovanja počinje velikim slovom, za razliku od naziva primitivnih tipova podataka
- Java je *case sensitive* programski jezik
- Literali se navode između navodnika:

```
String ime = "Petar";
```


Klasa `String` (2)

- Ukoliko je bar jedan operand za operator `+` tipa `String`, i rezultat će biti tipa `String`
- Primeri:

```
String rez1 = "Petar" + " Petrov";
```

```
String rez2 = "12" + 38;
```

Klasa `String` (3)

- Konverzija vrednosti primitivnog tipa u `String` je uvek bezbedna, pa se implicitno izvršava
- Primeri:
 - `int` vrednost `123` u `"123"`
 - `boolean` vrednost `true` u `"true"`
 - `double` vrednost `3.14` u `"3.14"`
- Konverzija vrednosti `String` u primitivni tip podatka nije bezbedna, pa se može pokušati izvršiti eksplicitno

Osnovni operatori

- Aritmetički:

+ , - , * , / , %

- Logički:

&& , || , ^ , !

- Relacioni:

> , >= , < , <= , == , !=

- Znakovni:

+

Izrazi

- Aritmetički izraz:
`visina - 100`
- Logički izraz sa logičkim operatorom:
`jare && pare`
- Logički izraz sa relacionim operatorom:
`visina > sirina`
- Znakovni izraz:
`"abc" + ime`

Operator dodele

- Vrednost izraza sa svoje desne strane dodeljuje promenljivoj sa leve strane

```
int prviBroj = 5;  
int drugiBroj = 4;  
int zbir = prviBroj + drugiBroj;  
zbir = zbir + 1;
```

Instrukcije

- Deklaracije i inicijalizacije promenljive:

```
int idealnaTezina = -1;
```

- Dodela vrednosti:

```
idealnaTezina = visina - 100;
```

- Kontrola toka izvršenja:

```
if (pol == 'm')  
    idealnaTezina = visina - 100;  
else  
    idealnaTezina = visina - 110;
```

Operatori inkrementiranja i dekrementiranja

- Unarni operatori koji uvećavaju odnosno umanjuju vrednost promenljive, respektivno
- Mogu se koristiti kao prefiks ili kao sufiks:

```
int i = 5;
```

```
i++;
```

```
int j = i++;
```

```
int k = ++i;
```

i dobija vrednost 5

i se inkrementira na 6

j dobija vrednost 6, zatim se i inkrementira na 7

i se inkrementira na 8, zatim k dobija vrednost i

Komentari

- U jednom redu:
 - počinju sa //
- U više redova:
 - počinju sa /*
 - završavaju sa */
- JavaDoc komentari:
 - Počinju sa /**
 - Završavaju sa */
 - Koriste se prilikom generisanje JavaDoc html dokumentacije

Elementi programa (1)

```
public class Sabiranje {  
    public static void main(String[] args) {  
        int prviBroj = 5;  
        int drugiBroj = 3;  
        int zbir = prviBroj + drugiBroj;  
        System.out.println(zbir);  
    }  
}
```

rezervisana reč

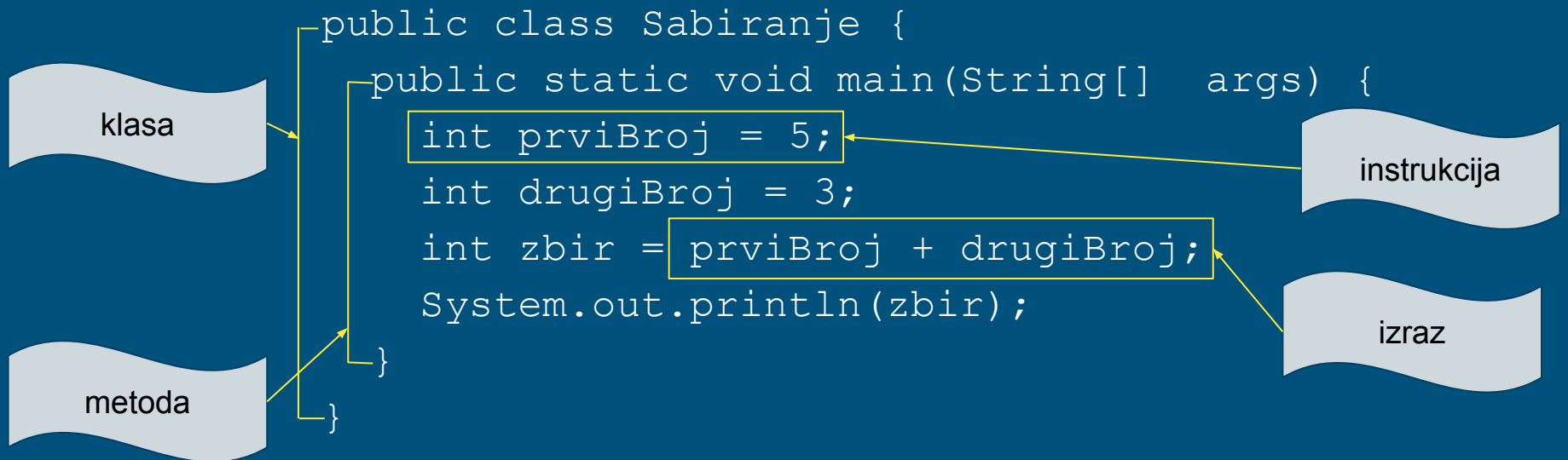
identifikator

literal

separator

operator

Elementi programa (2)



Integrirano razvojno okruženje

- Integrirano razvojno okruženje (engl. Integrated Development Environment - IDE)
 - Sadrži alate za brži efikasniji razvoj softvera
 - Podržava jedan ili više programskih jezika
 - Pojednostavljeno
 - kompajlera, pokretanje, *debugger*-a, generatora dokumentacije...
 - pisanje izvornog koda - generisanje, *autocomplete*...
 - integracija sa sistemima za upravljanje verzijama (svn, git...)
 - integracija sa sistemima za upravljanje projektom

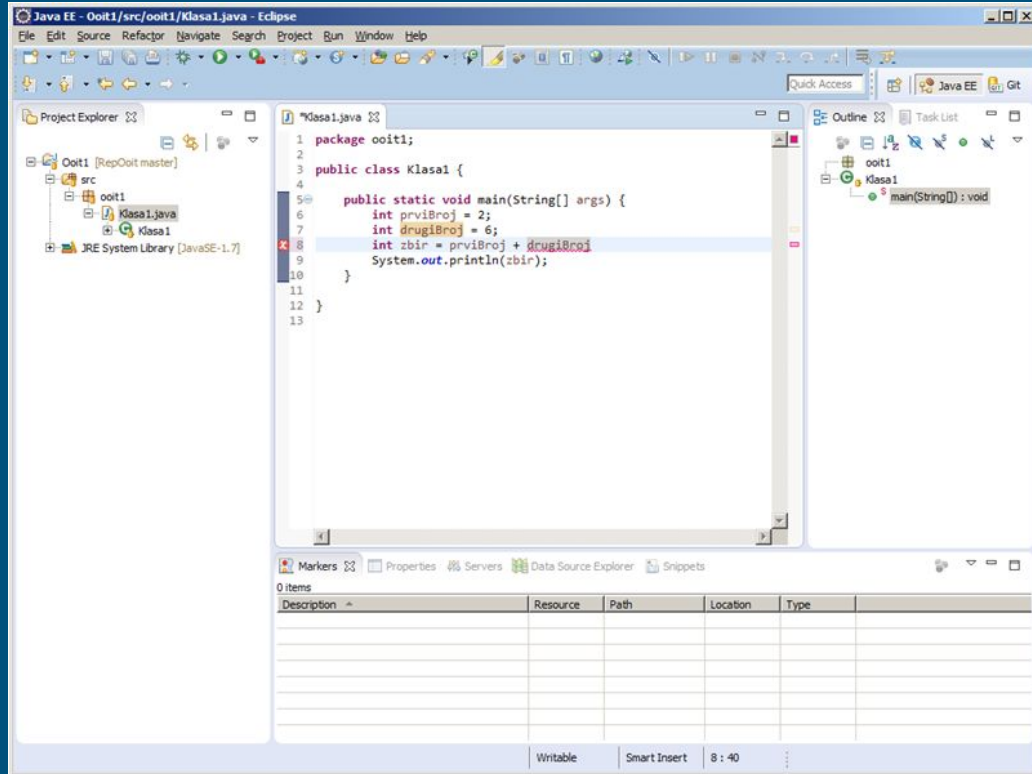
Eclipse IDE

- *Open source* proizvod
- Podržava razvoj Java aplikacija
- Zasnovan na *plug-in* arhitekturi
- Koristi koncepte:
 - *Workspace* (radni prostor)
 - *Project* (projekat)
 - *Perspective* (perspektiva)
- Verzije se imenuju po abecedi
 - ...
 - Neon
 - Oxygen
 - Photon

Instalacija Eclipse IDE

- Preuzeti željenu verziju za odgovarajući operativni sistem sa <http://www.eclipse.org/downloads/eclipse-packages/>
- Raspakovati arhivu u željeni direktorijum
- **Ukoliko je potrebno**, preuzeti i instalirati Java Development Kit (JDK) sa <http://www.oracle.com/technetwork/java/javase/downloads/index.html>
- Podesiti korišćenje JDK u Eclipse IDE

Izgled Eclipse IDE okruženja



Rezime

- Java programski jezik podržava objektno-orijentisane koncepte
- Java programi su portabilni jer se izvršavaju na JVM
- Java definiše primitivne tipove podataka i podržava definisanje složenih tipova podataka
- Eclipse IDE je open source razvojno okruženje koje omogućava efikasan razvoj Java aplikacija