

Referenčna tablica za jezik Java

Podatki in tipi

| Vrsta | | Opis | | | |
|-----------------|---------------|---|-----------|---------|--|
| Vrednostni tipi | Enostavni | Številčni | | | |
| | | Vrsta | Št. bitov | Tip | Razpon/Natančnost |
| | | Predznačena števila | 8 | byte | −128...127 |
| | | | 16 | short | −32,768...32,767 |
| | | | 32 | int | −2,147,483,648...2,147,483,647 |
| | | | 64 | long | −9,223,372,036,854,775,808...9,223,372,036,854,775,807 |
| | | Števila s plavajočo vejico | 32 | float | 1.5×10^{-45} to 3.4×10^{38} , 7-digit precision |
| | | | 64 | double | 5.0×10^{-324} to 1.7×10^{308} , 15-digit precision |
| | | Logični tip | 1 | boolean | true or false |
| | | znak (unicode) | 16 | char | ‘a’ (‘\u0000’ (or 0) and a maximum value of ‘\uffff’) |
| | | | | | |
| | Naštevni tipi | Uporabniško definirani naštevni tipi oblike public enum Month {‘April’, ‘Maj’, ‘Junij’} | | | |
| | Nični tipi | Vrednost null | | | |
| Sklicni tipi | Razredi | Nadrazred vseh razredov: Object | | | |
| | | Znakovni nizi: String | | | |

| | | |
|--|----------|---|
| | | Uporabniško definirani razredi oblike class C {...} |
| | Vmesniki | Uporabniško definirani tipi v obliki interface I {...} |
| | Polja | Polja in večdimenzionalna polja. Naprimer int[] ali int[][] |

Izrazi in operatorji

| Vrsta | Izraz | Opis | |
|-------------|------------------|--|--|
| Primarni | x.m | dostop do člana | myObject.someMethod() |
| | x(...) | Klic funkcije ali metode (metoda je funkcija kot član objekta) | someMethod("parameter") |
| | x[...] | element polja | int[] arr = int[5]; //najava int[] arr1 = int[] {1,2,3,4,5}; //najava in inicializacija |
| | x++ | Povečanje po uporabi | |
| | x-- | Pomanjšanje po uporabi | |
| | new T(...) | izvod razreda | MojRazred mr = new MojRazred(); |
| | new T(...) {...} | izvod razreda z konstruktorjem | |
| | new {...} | brezimni izvod razreda | |
| | new T[...] | izvod polja razredov | |
| | instanceof(T) | povpraševanje po razredu | |
| | ++x | povečaj pred uporabo | |
| | --x | pomanjšaj pred uporabo | |
| | (T)x | Pretvorba x v razred T (cast operacija) | |
| Aritmetični | x * y | množenje | |
| | x / y | deljenje | |
| | x % y | celoštevilsko deljenje z ostankom | |

| | | | |
|--|---------|------------------------------|--|
| | $x + y$ | seštevanje, združevanje niza | |
| | $x - y$ | odštevanje | |

| | | |
|--------------|-------------------|--|
| Primerjalni | $x < y$ | manj kot |
| | $x > y$ | več kot |
| | $x \leq y$ | manj kot ali enak |
| | $x \geq y$ | večji ali enak |
| | $x \text{ is } T$ | vrni true če je x tipa T , drugače vrni false |
| Logični | $x == y$ | enak (isti tip in še dejansko isti objekt v pomnilniku) |
| | $x != y$ | ni enak |
| | $x \&\& y$ | logični in |
| | $x \parallel y$ | logični ali |
| Pogojni | $x ? y : z$ | ternarni operator. Če je x true je vrednost izraza y , sicer z . |
| Priredilni | $x = y$ | priredba |
| Kombinatorni | $x \text{ op } y$ | krajši zapis kot $a=a+b$ je $a+=b$. Ostali $*=$ $/=$ $\%=$ $+=$ $-=$ $<<=$ $>>=$ $\&=$ $\^=$ $ =$ |

Stavki

| Vrste stavkov | Primeri |
|--|---|
| Napoved lokalne spremenljivke | <pre>int a; int b = 2, c = 3; a = 1;</pre> |
| Napoved lokalne konstante | <pre>final float pi = 3.1415927f; int r = 25;</pre> |
| Izrazi (izrazi so programski stavki, ki se izvednostijo v novo vrednost) | <pre>int i; //napoved i = 123; //izraz System.out.println(i); //izraz i++; // izraz</pre> |
| pogojni stavek if else | <pre>if (args.length() == 0) { System.out.println("No arguments"); } else { System.out.println("One or more arguments"); }</pre> |
| pogojni stavek switch | <pre>int n = args.length(); switch (n) { case 0: System.out.println("No arguments"); break; //prekini case 1: System.out.println("One argument"); break; //prekini default: //če gre skozi System.out.println("{0} arguments", n); break; }</pre> |

| | |
|---------------------------|---|
| | |
| zanka while | <pre>int i = 0; while (i < args.length()) { System.out.println(args[i]); i++; }</pre> |
| zanka do | <pre>string s; do { s = "bla bla"; if (s != null) System.out.println(s); } while (s != null);</pre> |
| zanka for | <pre>for (int i = 0; i < args.length(); i++) { System.out.println(args[i]); }</pre> |
| zanka foreach | <pre>foreach (string s in args) { System.out.println(s); }</pre> <p>//dela nad kopijo niza; zato pa omogoča vrivanje, brisanje, dodajanje</p> |
| prekinitveni stavek break | <pre>while (true) { string s = "foo bar"; if (s == null) break; System.out.println(s); }</pre> |
| stavek continue | <pre>for (int i = 0; i < args.length(); i++) { if (args[i] != '') continue; System.out.println(args[i]); }</pre> |
| brezpogojni skok goto | <pre>int i = 0; goto check; loop:</pre> |

| | |
|-----------------------------|---|
| | <pre>System.out.println(args[i++]); check: if (i < args.length()) goto loop;</pre> |
| vraćanje iz funkcije return | <pre>static int Add(int a, int b) { return a + b; }</pre> |

Razredi

Razred (class) je vzorec po katerem se izdelujejo predmeti oz. objekti (object) oz. njegovi izvodi. Izvode prirejamo spremenljivkam sklicnega tipa.n Razred ima lahko tudi statične elemente, ki jih uporabljamo neposredno, brez poprejšnje izdelave kakega izvoda. Razred določajo njegove lastnosti in metode. Razrede lahko nadgrajujemo z **dedovanjem**.

| Člani razreda so | Opis |
|------------------------------|--|
| Stalnica / Constants | stalne vrednosti |
| Spremenljivke / Fields | razredne spremenljivke |
| Metode / Methods | kaj lahko razred dela. “pamet” razreda. |
| Lastnosti / Properties | lastnosti razreda (za branje ali pisanje) (get, set, getterji, setterji) |
| Konstruktorji / Constructors | posebna metoda, ki se pokliče ob vzpostavitvi predmeta(new) |
| Čistilci / Destructors | posebna metoda, ki se pokliče predno je predmet uničen |

Pridevniki / Modifiers (določajo dostopnost lastnosti, metod ali razreda)

| Dostopnost | Pomen |
|------------|---|
| public | dostop ni omejen |
| protected | dostop je omejen na razred ali izpeljani razred |
| private | Samo znotraj razreda ali izpeljanega razreda |

Označevanje pripadnosti znotraj razreda

| Oznaka | Pomen |
|--------|--|
| this | pripada temu razredu this.član |
| super | pripada nadrazredu super.članNadrazreda |

Primer razreda

| | |
|--|---|
| package Si.Housing | imenski prostor (prepreči imenske konflikte) |
| import java.util.Properties; | uvoz imenskega prostora (odvisnost v času prevajanja) |
| <pre>** Bla bla * @return the image at the specified URL */</pre> | javadoc komentar |
| public class Reverser extends BaseReverser { | napoved razreda, ki je izpeljan iz nadrazreda |
| int maxLength = 10000; | stalnica |
| private String txtReversed; private String txtOrigin; int count = 0; | razredna spremenljivka razredna spremenljivka z začetno prireditvijo vrednosti |
| static String greeting = "Hello from reverser"; | statična lastnost, ki je skupna vsem izpeljankam razreda |
| public Reverser() { txtReversed = ""; } | konstruktor začetna prireditve |
| public Reverser(String initial) { txtReversed = initial; | preobloženi konstruktor lahko uporabimo this .txtReversed, da z njo eksplicitno določimo pripadnost spremenljivke razredu |

| | |
|--|---|
| } | |
| <pre> public String GetTxtReversed(){ return txtReversed; } public void SetTxtOrigin(String yourInput){ txtOrigin= yourInput; } </pre> | Lastnosti kot metoda (izpostavijo razredne spremenljivke) |

| | |
|---|---|
| <pre> public void Reverse() { int x; this.txtReversed = super.SomeMethodFromBase(); } @Override public String SomeMethodFromBase() {} </pre> | <p>Metoda z telesom lokalna spremenljivka metode klic druge metode</p> <p>nadjačana metoda nadrazreda (če rabimo)</p> |
| <pre> public void Reverse(string toAppend) {} </pre> | preobložena metoda |
| <pre> } </pre> | //konec razreda |

Vaje

| | |
|--|---|
| <pre> public class Runner { public static void main(String[] args) { System.out.println("Enter Integer:"); String s = System.console().readLine(); System.out.println("You entered :" + s); } } </pre> | <p>Standardni vhod / izhod za konzolne aplikacije</p> <p>javac Runner.java java Runner</p> |
| <pre> System.out.print("You entered param :" + args[0]); </pre> | argumenti ob zagonu iz ukazne vrstice |
| <pre> import java.io.BufferedReader; import java.io.File; import java.io.FileReader; </pre> | Branje tekstovne datoteke |

```
import java.io.IOException;

public class Runner {
    public static void main(String[] args) {
        System.out.print("Enter file path:");
        String s = System.console().readLine();
        try { //blok z lovljenjem izjem
            File file = new File(s);
            FileReader fileReader = new FileReader(file);
            BufferedReader br = new BufferedReader(fileReader);
            String doc = "";
            String line;
            while((line = br.readLine()) != null) {
                doc += line + "\n";
            }
            System.out.println("Computer read : " + doc);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```