

**VEŽBA 4: Petlje (WHILE, DO-WHILE, FOR-numerička)****Primer 1 – While petlja**

```
package com.asss.uup;

/*
 * Napisati program koji će omogućiti korisniku da pomoću
 * Scanner objekta unese određenu reč, a potom i jedan broj.
 * Unetu reč, pomoću WHILE petlje, odštampati navedeni broj puta.
 */

import java.util.Scanner;

public class While_1 {

    public static void main(String[] args) {

        // deklaracija promenljivih za unos sa tastature
        String rec;
        int brojPonavljanja;

        // definisanje promenljive radi kontrole petlje
        int brojac = 1;

        // pravljenje Scanner objekta
        Scanner unosSaTastature = new Scanner(System.in);

        // unos željenih vrednosti i njihova dodela promenljivama
        System.out.print("Unesite željenu reč: ");
        rec = unosSaTastature.nextLine();

        System.out.print("Koliko puta želite da je šampamo? ");
        brojPonavljanja = unosSaTastature.nextInt();

        /*
         * while petlja - dokle god je brojač manji od broja
         * ponavljanja dotle će se štampati redni
         * broj ponavljanja i uneta reč
         * (brojač u svakoj iteraciji uvećava
         * svoju vrednost za 1 - inkrement)
         */
        while (brojac <= brojPonavljanja) {
            System.out.println(brojac + ". " + rec);
            brojac++;
        }

        unosSaTastature.close();
    }
}
```

**Primer 2 – Beskonačna while petlja**

```
package com.asss.uup;

import java.util.Scanner;

/*
    Napisati program koji će, upotrebom while petlje,
    omogućiti korisniku unos proizvoljnog broja vrednosti
    koje pripadaju skupu prirodnih brojeva.
    Ukoliko dodje do situacije da korisnik unese negativan
    broj preskočiti iteraciju "kao da se ništa nije desilo",
    dok ukoliko unese 0, prekinuti unos i ispisati mu zbir.
*/

public class While_2 {

    public static void main(String[] args) {
```

```
public static void main(String[] args) {

    // deklarisanje promenljive za unos sa tastature
    int unetiBroj;

    // definisanje promenljive za čuvanje zbira
    int zbir = 0;

    // pravljenje Scanner objekta
    Scanner unosSaTastature = new Scanner(System.in);

    /*
    while petlja (beskonačna)
    Ovakvim oblikom kontrolnog mehanizma, koji
    će podrazumevano uvek biti "true" omogućavamo
    beskonačno izvršavanje petlje. Prilikom rada
    sa beskonačnom petljom treba voditi računa da
    se uvek obezbedi izlazak iz nje.
    Shodno definisanoj logici, dokle god korisnik
    bude unosio prirodne brojeve (cele i pozitivne)
    zbir će se uvećavati sa unetu vrednost.
    Ukoliko unese negativnu vrednost (else if),
    zahvaljujući ključnoj reči "continue", doći će
    do preskakanja tog kruga iteracije i petlja
    će se izvršavati od početka.
    S druge strane, ukoliko unese 0 (else),
    zahvaljujući ključnoj reči "break", doći će do
    prekida izvršavanja petlje i tok izvršavanja
    kod biće prenet na prvi iskaz nakon petlje.
    */
    while (true) {
        System.out.print("Unesite broj: ");
        unetiBroj = unosSaTastature.nextInt();

        if (unetiBroj > 0) {
            zbir += unetiBroj;
        } else if (unetiBroj < 0){
            continue;
        } else {
            break;
        }
    }

    System.out.println("Zbir unetih prirodnih " +
        " brojeva je " + zbir);

    unosSaTastature.close();
}
```

## Primer 3 – Do-while petlja (modifikovani Craps)

```
package com.asss.uup;

import java.util.Scanner;

/*
Napisati program koji predstavlja modifikaciju igre Craps.
Modifikovana varijanta igre treba da omogući korisniku unos
određene sume novca kako bi mogao da učestvuje u igri, a potom
i da izvrši "bacanje" dve kockice (vrednosti od 1 do 6).
Za svako bacanje korisnik mora da uloži 100 poena iznosa koji
je uplatio. Ukoliko rezultat bacanja bude 7 korisnik dobija
nagradu 3 puta veću od uložених sredstava, dok ukoliko je
rezultat bacanja bude 12 uložena sredstva treba pomnožiti sa 9.
Tokom igre omogućiti ispis svih dešavanja u konzoli (vrednosti
kockica, njihov zbir, rezultat igre i stanje sredstava korisnika).
Na kraju svakog bacanja, pitati korisnika da li želi da nastavi
igru i na osnovu odgovora mu omogućiti novo bacanje ili izlaz.
Ukoliko igrač ostane bez sredstava, onemogućiti mu dalju igru i
ispisati mu o tome poruku sa podatkom o ukupnom broju bacanja.
Logiku zadatka implementirati upotrebom do-while petlje.
*/

public class DoWhile {

    public static void main(String[] args) {
```

1. Deklarisanje promenljivih, definisanje potrebnih konstanti, pravljenje Scanner objekta i njegova upotreba.

```
public static void main(String[] args) {  
  
    /*  
        deklarisanje promenljivih za praćenje vrednosti  
        kockica, njihov zbir i stanje sredstava korisnika  
    */  
    int kockica1, kockica2, zbirKockica, stanjeSredstava;  
  
    // deklaracija promenljive za odabir nastavka igre  
    int izbor;  
  
    // definisanje promenljive za praćenje broja bacanja  
    int brojBacanja = 0;  
  
    /*  
        fiksno definisana vrednost uloga, dobitaka  
        i koeficijenata dobitka (konstante)  
    */  
    final int ULOG = 100;  
    final int DOBITAK_7 = 7;  
    final int DOBITAK_12 = 12;  
    final int KOEF_7 = 3;  
    final int KOEF_12 = 9;  
  
    //pravljenje objekta tipa Scanner  
    Scanner unosSaTastature = new Scanner(System.in);  
  
    //dodela vrednosti promenljivoj stanjeSredstava  
    System.out.print("Unesite pocetno stanje sredstava : ");  
    stanjeSredstava = unosSaTastature.nextInt();  
}
```

## 2. Implementacija logike programa.

- a. Provera ispunjenosti uslova za učešće u igri i implementacija logike bacanja i obrade rezultata bacanja.

```
//obrada igre i prikaz vrednosti kockica i stanja sredstava
if (stanjeSredstava >= ULOG) {
    do {
        // odbijanje uloga pred svako bacanje
        stanjeSredstava -= ULOG;

        // bacanje (vrednosti se generišu upotrebom klase Math)
        kockica1 = (int) (6 * Math.random() + 1);
        kockica2 = (int) (6 * Math.random() + 1);

        // evidentiranje broja bacanja (inkrement)
        brojBacanja++;

        // poruka o bacanju (praćenje toka igre)
        System.out.println("\n" + brojBacanja + ". BACANJE:"
            + "\nVrednost kockica je " + kockica1
            + " i " + kockica2 + ".");

        zbirKockica = kockica1 + kockica2;

        if (zbirKockica == DOBITAK_7) {
            stanjeSredstava += KOEF_7 * ULOG;
            System.out.println("Zbir kockica je 7! " +
                "Vase stanje se uvecava za "
                + KOEF_7 * ULOG +
                " i ono sada iznosi "
                + stanjeSredstava + ".");
        } else if (zbirKockica == DOBITAK_12) {
            stanjeSredstava += KOEF_12 * ULOG;
            System.out.println("Zbir kockica je 12! " +
                "Vase stanje se uvecava za "
                + KOEF_12 * ULOG +
                " i ono sada iznosi "
                + stanjeSredstava + ".");
        } else {
            System.out.println("Zbir kockica je " + zbirKockica
                + ". Trenutno sredstava je "
                + stanjeSredstava + ".");
        }
    }
}
```

- b. Obrada slučajeva koji se odnose na: nastavak igre, prekid igre, odnosno onemogućavanje igre (ukoliko je početno stanje sredstava manje od visine uloga – prvi IF iskaz)

```
if (stanjeSredstava >= ULOG) {
    System.out.print("Zelite li da nastavite igru?" +
        "\n1 (DA)\t2 (NE)\t: ");
    izbor = unosSaTastature.nextInt();

    while (izbor != 1 && izbor != 2) {
        System.out.print("\nUnesite ispravan odgovor: ");
        izbor = unosSaTastature.nextInt();
    }
    if (izbor == 1) {
        continue;
    } else {
        System.out.println("\nNadamo se da Vam je bilo " +
            "interesantno! Nama sigurno jeste!\n" +
            "Trenutno stanje Vasih sredstava je "
            + stanjeSredstava + ".\n"
            + "Ukupan broj odigranih bacanja je "
            + brojBacanja + ".");
        break;
    }
} else {
    System.out.println("\nNemate dovoljno novca za dalju" +
        " igru.\nStanje Vasih sredstava je "
        + stanjeSredstava + ".\n"
        + "Ukupan broj odigranih bacanja je "
        + brojBacanja + ".\nNe kune majka"
        + " sina što se kladi, nego što se "
        + "vadi, pa nam Vi dodite opet!\n"
        + "PS\nKUĆA UVEK DOBIJA!!!");
}

while (stanjeSredstava > ULOG);

} else {
    System.out.println("\nPocetno stanje sredstava mora biti vece od"
        + " 100.\nPokrenite program ponovo...\n");
}

unosSaTastature.close();
}
```

## Primer 4 – For petlja

```
package com.asss.uup;

import java.util.Scanner;

/*
 * Napisati program koji će korisniku omogućiti
 * unos prirodnog broja, a potom mu, pomoću for
 * petlje izračunati faktoriјel tog broja
 * (proizvod uzastupnih brojeva) i prikazati ga.
 */
public class For_1 {

    public static void main(String[] args) {

        // deklarisanje promenljive za unetu vrednost
        int n;

        // podrazumevana vrednost faktoriјela za 0 i 1
        int faktoriјel = 1;

        // pravljenje Scanner objekta
        Scanner unosSaTastature = new Scanner(System.in);

        // unos vrednosti i njena dodela promenljivoj
        System.out.print("Unesite broj: ");
        n = unosSaTastature.nextInt();

        // ispitivanje tacnosti unete vrednosti
        if (n > 0) {
            /*
             * for petlja - počev od početne vrednosti i
             * za predviđeni broj iteracija
             * n izvršava definisanu logiku
             */
            for (int i = 1; i <= n; i++) {
                faktoriјel *= i;
            }
            System.out.println("Faktoriјel broja " + n
                               + " (" + n + "!)" je " +
                               faktoriјel + ".");
        } else if (n == 0) {
            System.out.println("Faktoriјal broja 0 (0!) "
                               + "(prema konvenciji za "
                               + "prazan proizvod) je "
                               + faktoriјel + ".");
        } else {
            System.out.println("Prema teoriji brojeva" +
                               " faktoriјel negativnih" +
                               " brojeva ne postoji!");
        }

        unosSaTastature.close();
    }
}
```



## Primer 5 – For petlja (dekrement)

```
package com.asss.uup;

import java.util.Scanner;

/*
 * Napisati program koji će korisniku omogućiti
 * unos prirodnog broja, a potom mu, pomoću for
 * petlje izračunati sve delioce tog broja i
 * prikazati ih u nerastućem redosledu.
 */
public class For_2 {

    public static void main(String[] args) {

        // deklarisanje promenljivih
        int broj, delilac;

        // pravljenje Scanner objekta
        Scanner unosSaTastature = new Scanner(System.in);

        // unos i dodela vrednosti sa tastature
        System.out.print("Unesite broj: ");
        broj = unosSaTastature.nextInt();

        //trazenje i prikaz delioca broja n
        System.out.print("\nDelioci za broj " +
            broj + " su: | ");
        for (int i = broj; i > 0; i--) {
            if(broj % i == 0){
                delilac = i;
                System.out.print(delilac + " | ");
            }
        }

        System.out.println();

        unosSaTastature.close();
    }
}
```

**Primer 6 – For petlja + uslovni (ternarni) operator**

1. Postavka zadatka, definisanje promenljivih, pravljenje Scanner objekta i njegova upotreba.

```
package com.asss.uup;

import java.util.Scanner;

/*
    Napisati program koji će korisniku omogućiti unos broja,
    a potom mu, pomoću for petlje, izračunati zbir svih parnih
    i neparnih pozitivnih brojeva na intervalu od 0 do tog broja.
    Nakon toga, izračunati aritmetičku sredinu kako parnih, tako
    i neparnih brojeva i omogućiti korisniku da odabere prikaz
    jedne od te dve.
    Prikaz izvršiti upotrebom uslovnog (ternarnog) operatora.
*/

public class For_3 {

    public static void main(String[] args) {

        // deklarisanje promenljive za unetu vrednost
        int gornjaGranica;

        // definisanje promenljivih za izračunavanje
        int brojP = 0, zbirP = 0, asP = 0;
        int brojNP = 0, zbirNP = 0, asNP = 0;

        // pravljenje Scanner objekta
        Scanner unosSaTastature = new Scanner(System.in);

        // unos i dodela vrednosti sa tastature
        System.out.print("Unesite broj: ");
        gornjaGranica = unosSaTastature.nextInt();
        System.out.println();
    }
}
```

## 2. Implementacija logike programa.

- a. Prolazak kroz dati interval, izvršenje potrebnih izračunavanja i definisanje opcionih prikaza.

```
/*  
    for petlja - prolaskom kroz dati opseg  
                vrši se provera da li je  
                trenutni broj paran ili ne,  
                shodno čemu se vrši određeno  
                sabiranje i brojanje  
*/  
  
for (int i = 0; i <= gornjaGranica; i++) {  
    if (i % 2 == 0) {  
        zbirP += i;  
        brojP++;  
    } else {  
        zbirNP += i;  
        brojNP++;  
    }  
}  
  
// izracunavanje aritmetickih sredina  
asP = zbirP / brojP;  
asNP = zbirNP / brojNP;  
  
// definisanje mogućih odgovora  
String parni = "PARNI BROJEVI\n" +  
               "Zbir: " + zbirP + "\n" +  
               "Aritmetička sredina: " + asP;  
  
String neparni = "NEPARNI BROJEVI\n" +  
                 "Zbir: " + zbirNP + "\n" +  
                 "Aritmetička sredina: " + asNP;
```

- b. Korisnički unos prikaza, osiguranje unosa ponuđenih opcija, odabir prikaza prema unosu pomoću uslovnog (ternarnog) operatora i njegovo štampanje.

Napomena:

Uslovni (ternarni) operator se može koristiti umesto prostijih konstrukcija IF-ELSE iskaza kontrole toka.

```
// unos izbora prikaza
System.out.print("Odaberite prikaz koji " +
    "želite:\t1 (parni) 2 (neparni)"
    + "\n> ");

int izbor = unosSaTastature.nextInt();
System.out.println();

while (izbor != 1 && izbor != 2) {
    System.out.print("Morate odabrati " +
        "redni broj jedne od" +
        " ponuđenih opcija: ");
    izbor = unosSaTastature.nextInt();
    System.out.println();
}

// deklaracija promenljive za prikaz poruke
String porukaZaPrikaz;

/*
    dodela vrednosti promenljivoj prema izboru
    (upotreba uslovnog - ternarnog operatora)
*/
porukaZaPrikaz = izbor == 1 ? parni : neparni;

System.out.println(porukaZaPrikaz + "\n");

unosSaTastature.close();
}
```