

```
/* primer definisanja klase Cudo da tri atributa,  
   u aplikaciji instanciranje objekta i pozivi metoda instance */
```

```
class Cudo {  
    String boja;  
    boolean sitost;  
    char pol;  
  
    void hrani(){  
        if (sitost)  
            System.out.println("hvala, jelo sam");  
        else {  
            System.out.println("Njam");  
            sitost=true;  
        }  
    }  
  
    void prikaz(){  
        System.out.print("Ja sam " + boja + " boje, " + pol + " pola, ");  
        if (sitost)  
            System.out.println( " sito sam");  
        else  
            System.out.println( " gladno sam");  
    }  
  
    public static void main( String [] args){  
        Cudo j = new Cudo();  
        j.prikaz();  
        j.boja="crvena";  
        j.sitost=false;  
        j.pol='M';  
        j.hrani();  
        j.prikaz();  
        j.hrani();  
        j.prikaz();  
  
    } // kraj main  
} // kraj class
```

```
/* Definicija klase Krug
```

```
- krug je odredjen prvom i drugom koordinatom centra u ravni (x,y) i  
  poluprecnikom r */
```

```
class Krug{  
    int x;  
    int y;  
    int r;  
  
    // konstruktor krugova zadatog poluprecnika u zadatom poloazaju  
    Krug(int x, int y, int r){
```

```

    this.x=x;
    this.y=y;
    this.r=r;
}

//konstruktor krugova jedinicnog poluprecnika u zadatom poloazju
Krug(int a, int b){
    this(a,b,1);
}
Krug(){
    this(1,1,1);
}
void predstaviKrug(){
    System.out.println("Krug ("+x+", "+y+") poluprecnika "+r);
}

public static void main (String []args){

    /* ova klasa je i aplikacija, pošto sadrži i izvršni deo
       - main metod. Ako klase nemaju main metod, tada nisu aplikacije,
       već se mogu pozivati i koristiti u okviru drugih Java klasa -
       aplikacija */

    Krug k1=new Krug(5,5,2);
    Krug k2=new Krug(-3,3);
    Krug k3=new Krug();
    k1.predstaviKrug();
    k2.predstaviKrug();
    k3.predstaviKrug();
}
}

/* primer definisanja klase Avion, ciji su objekti opisani
   identifikatorom tipa aviona i sa tri koordinate koje odredjuju
   poziciju aviona u prostoru */

class Avion{

    // promenljive koje opisuju stanje objekata klase Avion
    String tip;
    int x;    // promenljive instance
    int y;
    int z;

    // metode koje opisuju ponasanje objekata klase Avion
    Avion pomeri(int a,int b, int c){
        x=a;
        y=b;
        z=c;
        return this;
    }
}

```

```

// isto ali na drugi nacin
void pomeriAvion(int x, int y, int z){
    this.x=x;
    this.y=y;
    this.z=z;
}

void prizemlji(){
    x=0;
    y=0;
    z=0;
}

void vratiPos(){
    System.out.println
        ("Avion " + tip + " je na poziciji " + x + ", " + y + ", " + z);
}

boolean proveripos(int a,int b,int c){
    boolean pom;
    pom =(x==a) &&(y==b) &&(z==c);
    return pom;
}

public static void main (String []args){
    Avion o1=new Avion();
    Avion o2=new Avion();
    o1.vratiPos();
    o1.tip="boing727 JAT1";
    o2.tip="AIRBUS France33";
    o1.pomeri(2,4,5);
    o1.vratiPos();
    o2.pomeriAvion(10,20,25);
    o2.vratiPos();
    o1.prizemlji();
    o1.vratiPos();
    if (o2.proveripos(10,20,30))
        System.out.println
            ("Avion " + o2.tip + " se nalazi na datoj poziciji");
    else
        System.out.println
            ("Avion " + o2.tip + " se ne nalazi na datoj poziciji");
}
}

```

```

/* primer klase Robot koja opisana je samo
   polozajem robota u ravni zadatim sa dve koordinate */

class Robot{
  int x;
  int y;

  // overload konstruktori

  Robot(){};      // default konstruktor

  Robot(int x1,int y1){
    x=x1; y=y1;
  }

  void idiGore(){
    y++;
  }

  void idiGore(int a){ // a je duzina pomeranja
    y +=a;
  }

  void idiDole(){
    y--;
  }

  void idiLevo(){
    x--;
  }

  void idiDesno(){
    x++;
  }

  void idiGoreLevo(){ // moguće je pozvati metode iste klase
    idiGore();        // iz nekog drugog metoda
    idiLevo();
  }

  void idiGoreDesno(){
    y++; x++;
  }
  void idiDoleDesno(){
    y--; x++;
  }
  void idiDoleLevo(){
    y--; x--;
  }

  void promeniPos(int x1,int y1) {
    x=x1; y=y1;
  }
}

```

```

    }

    boolean proveripos(int x1,int y1) {
        return( x==x1 && y==y1 );
    }

    void prikazi(){
        System.out.println
            ("Robot je na poziciji (" + x + ", " + y + ")");
    }

    int vratiX(){
        return (x);
    }

    int vratiY(){
        return y;    // moze i bez zagrada
    }

    public static void main( String [] args){

        Robot R1 = new Robot();
        Robot R2 = new Robot(1,1);
        R1.prikazi();
        R2.prikazi();
        R1.idiGore();
        R2.idiGoreLevo();
        R1.proveriPos(1,1);    // nista se ne prikazuje!
        R1.vratiX();          // nista se ne prikazuje!
        R2.vratiX();
        System.out.println("x= " + R1.vratiX());    // ovako se prikazuje!
        System.out.println("y= " + R1.vratiY());
        R1.prikazi();
        R2.prikazi();
        System.out.println( R1.proveriPos(0,1));    // ovako se prikazuje!

        if (R2.proveriPos(0,0))
            System.out.println( "robot R2 je u tacki (0,0)");
        else
            System.out.println( "robot R2 nije u tacki (0,0)");
    }
}

```

```
/* Za potrebe inf. sistema autobuske stanice definisati klasu Linija
sa obeležjima:
    naziv destinacije, cas polaska, broj slobodnih mesta.
Omoguciti u aplikaciji:
    o unos linija,
    o prikaz svih linija izabrane destinacije,
    o rezervaciju karte sa mogucim porukama 'nema više mesta'
ili 'karta je izdata'. */
```

```
import java.util.*;
class Linija{
    String dest;
    String cas;
    int brojSl;
    Linija(String dest,String cas,int brojSl){
        this.dest=dest;
        this.cas=cas;
        this.brojSl=brojSl;
    }

    void prikazi(){
        System.out.println("Destinacija: " + dest);
        System.out.println("Cas polaska: " + cas);
        System.out.println("Broj slobodnih mesta: " + brojSl);
    }

    void rezervisi(){
        if (brojSl>0){
            brojSl--;
            System.out.println("Karta je izdata!");
        }
        else
            System.out.println("Nema više mesta!");
    } // kraj rezervisi()

    public static void main(String []args) {

        Scanner ulaz = new Scanner(System.in);

        System.out.println("Koliko ima linija?");
        int br = ulaz.nextInt();

        System.out.println("Unesite red voznje");
        for (int i=0; i<nizLin.length; i++){
            System.out.println("Destinacija: ");
            String d =ulaz.nextLine();
            System.out.println("Cas polaska: ");
            String c = ulaz.nextLine();
            System.out.println("Broj slobodnih mesta: ");
            int brSl = ulaz.nextInt();
            nizLin[i] = new Linija(d,c,brSl);
        }
    }
}
```

```

// unos datuma za koji pravimo red voznje
System.out.println ("\nUnesite datum (dd\mm\gg) koji zelite: ");
String datum=ulaz.nextLine();

System.out.println ("\nRed voznje za dan " + datum);
for (int i=0; i< br; i++){
    nizL[i].prikaz();
}
// br = nizLin.length

while (true){
System.out.print("Unesite izabranu destinaciju: ");
String d = ulaz.nextLine();
if (d.equalsIgnoreCase("kraj")) break;
System.out.println("\nLinije za destinaciju "+d+" su:");

for (int i=0; i<nizLin.length; i++){
    if ((nizLin[i].dest).equalsIgnoreCase(d))
        nizLin[i].prikazi();
}

System.out.print("Unesite cas polaska: ");
String c = ulaz.nextLine();

for (int i=0; i<nizLin.length; i++){
    if ((nizLin[i].dest).equalsIgnoreCase(d)&&
        (nizLin[i].cas).equals(c))
        nizLin[i].rezervisi();

} // kraj for
} // kraj while
} // kraj main
} // kraj class

```