

Ushtrimi 1.

Ndertoni nje program ne java qe klonon nje linkedList ne nje linked list tjeter.

Zhgjidhja:

```
import java.util.*;
public class Exercise18 {
    public static void main(String[] args) {
        // create an empty linked list
        LinkedList <String> c1 = new LinkedList <String> ();
        c1.add("E Kuqe");
        c1.add("Jeshile");
        c1.add("E Zeze");
        c1.add("E Bardhe");
        c1.add("E Verdhe");
        System.out.println("Original linked list: " + c1);
        LinkedList <String> newc1 = new LinkedList <String> ();
        newc1 = (LinkedList)c1.clone();
        System.out.println("Cloned linked list: " + newc1);
    }
}
```

Ushtrimi 2. Ndertoni nje klase ne java me emrin Student dhe me te dhena emrin dhe kodin. Klononi objektin e kesaj klase duke perdorur metoden clone (). (object clone kujdes!!!)

```
class Student implements Cloneable{
    int kodi;
    String emri;
    Student(int kodi,String emri){
        this.kodi=kodi;
        this.emri=emri;
    }
    public Object clone()throws CloneNotSupportedException{
        return super.clone();
    }
}

public class Klonimi {
    public static void main(String args[]) throws CloneNotSupportedException{
        Student s1=new Student(101,"Anxhi");
        Student s2=(Student)s1.clone();
        System.out.println(s1.kodi+" "+s1.emri);
        System.out.println(s2.kodi+" "+s2.emri);
    }
}
```

Ushtrimi 3.

Te ndertohet klasa Punonjes me te dhena private, emri, dite active ne pune dhe site pushimi.

Te perdoret konstruktori per leximin e ketyre te dhenave, me parametra formal duke perdorur dhe objektin referent. Me pas ndertoni klasen publike Punonjes clone() per te klonual superklasen. Shtypni ne main te dhenat.

Kujdes!! Duhet ndertuar dy funksione shtese te cilet llogarisin ditet active ne pune dhe ditet e pushimit.

```
import java.util.Date;
import java.util.GregorianCalendar;

class Punonjes implements Cloneable {

    private String emri;
    private double ditepune;
    private Date pushim;

    public Punonjes(String emri, double ditepune)
    {
        this.emri = emri;
        this.ditepune = ditepune;
        pushim = new Date();
    }

    public Object clone() throws CloneNotSupportedException
    {
        // Object clone()
        Punonjes obj = (Punonjes)super.clone();

        obj.pushim = (Date)pushim.clone();

        return obj;
    }

    // I japim ditets se pushimit daten
    public void setPushim(int viti, int muaji, int dita)
    {
        Date newPushim = new GregorianCalendar(viti, muaji - 1, dita)
            .getTime();

        // instance field mutation
        pushim.setTime(newPushim.getTime());
    }

    public void Nrditepune(double byPercent)
```

```

{

    double nr=ditepune* byPercent / 100;
    ditepune += nr;
}

public String toString()
{
    return ("Punonjes[emri=" + emri
        + ", ditepune=" + ditepune
        + ", pushim=" + pushim
        + "]");
}
}

public class Vlera {
    public static void main(String[] args) throws CloneNotSupportedException
    {

        Punonjes O = new Punonjes("ABC X. YZ", 50000);

        O.setPushim(2000, 1, 1);
        Punonjes copy = (Punonjes) O.clone();

        copy.Nrditepune(10);
        copy.setPushim(2002, 12, 31);

        System.out.println("fillestari= " +O);
        System.out.println("copy= " + copy);

    }
}

```

Ushtrimi 4.

Eshte dhene vektori i tipit string si nje ArrayList. Ndertoni kopjen origjinale te tij duke perdorur metoden clone();

```

import java.util.ArrayList;
import java.util.Collections;
public class Vektor {
    public static void main(String[] args) {
        ArrayList<String> c1= new ArrayList<String>();
        c1.add("Kuq");
        c1.add("Jeshil");
    }
}

```

```

        c1.add("Zi");
        c1.add("Bardhe");
        c1.add("Verdhe");
        System.out.println("Afisho vektorin origjinal " + c1);
        ArrayList<String> newc1 = (ArrayList<String>)c1.clone();
        System.out.println("Klonimi i Array List: " + newc1);
    }
}

```

Ushtrimi 5.

Shenim: Konstruktoret e kopjimit janë konstruktorë të veçantë në një klasë e cila merr argumentin për llojin e vet të klasës. Pra, kur kaloni një shembull të klasës për të kopjuar konstruktorin, atëherë konstruktori do të kthejë një shembull të ri të klasës me vlerat e kopjuara nga instanca e argumentit. Ju ndihmon të klononi objektin me ndërfaqen Cloneable.

Te ndertohet klasa baze TestNje me te dhena private x dhe y. Klasa e derivuar TestDy permбан si te dhene private ndryshoren z. Te klonohen konstruktoreset e klasave me lart.

M1.

```

class PointOne
{
    private int x;
    private int y;

    public PointOne(int x, int y){
        this.x = x;
        this.y = y;
    }
    //public int getA() {return x;}
    //public int getAb() {return y;}
}
class PointTwo extends PointOne
{
    private int z;

    public PointTwo(int z){
        super(z,z); //Call Super class constructor here
        this.z = z ;
    }
}
public class PointNje
{
    public static void main(String[] args)
    {
        PointOne one = new PointOne(1,2);
        PointTwo two = new PointTwo(3);
    }
}

```

```

    PointOne clone1 = one;
    PointOne clone2 = two;

    //Let check for class types
    System.out.println(clone1.getClass());
    System.out.println(clone2.getClass());
}
}

M2. class PointOne implements Cloneable
{
    private Integer x;
    private Integer y;

    public PointOne(Integer x, Integer y)
    {
        this.x = x;
        this.y = y;
    }

    public PointOne copyPoint(PointOne point) throws CloneNotSupportedException
    {
        if(!(point instanceof Cloneable))
        {
            throw new CloneNotSupportedException("Invalid cloning");
        }

        //Can do multiple other things here
        return new PointOne(point.x, point.y);
    }
}

public class PointNje
{
    public static void main(String[] args)
    {
        PointOne one = new PointOne(1,2);
        // PointTwo two = new PointTwo(1,2,3);

        PointOne clone1 = one;
        // PointOne clone2 = two;

        //Let check for class types
        System.out.println(clone1.getClass());
        // System.out.println(clone2.getClass());
    }
}

```

Ushtrimi 6. D.SH

Te ndertohet klasa DogName me te dhene private dname. Ndertoni objektin origjinal dhe ate te klonuar brenda te njejtës klase. Implementoni nderfaqen Cloneable.

Ushtrimi 7. Eshte dhene vektori i tipit char A={"j","a","v","a"}

- Konvertoni char Array ne string
- Krijoni nje string te ri nepermjet nje variabli te ri. Afishoni rezultatet.

```
public class StringExample{
    public static void main(String args[]){
        char ch[]={ 's','t','r','i','n','g','s' };
        String s1=new String(ch);//converting char array to string
        String s2=new String("example");//creating java string by new keyword
        System.out.println(s1);
        System.out.println(s2);
    }
}
```

Ushtrimi 8.

Eshte dhene nje variable I tipit string dhe nje variable I tipit int. Gjeni Shumen e dy variablave duke konvertuar te dhenen e tipit string ne te dhene te tipit int.

```
public class JavaExample{
    public static void main(String args[]){
        String str="123";
        int inum = 100;

        /* Konvertimi I var string ne int
         * , vlera inum2 do te behet 123 mbas
         * konvertimit nepermjet komandes
         */
        int inum2 = Integer.parseInt(str);

        int sum = inum+inum2;
        System.out.println("Result is: "+sum);
    }
}
```

Ushtrimi 9.

Ndertoni nje variable te tipit string dhe me pas gjejn gjatesine e variablit.

```

public class LengthExample{
public static void main(String args[]){
String s1="javatpoint";
String s2="python";
System.out.println("string length is: "+s1.lengthSystem.out.println("string length is: "+s2.length(
));
}}

```

Ushtrimi 10. Te ndertohet String Array dhe nje variable e tipit string. Kontrolloni nese ky variable ndodhet tek ndonje nga vlerat e vektorit. Nese po afishoni ate dhe pozicionin se ku gjendet ne vector.

```

import java.util.*;
public class Stri {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
String[] StrArray= {"Liber", "Fletore", "Laps", "Stilolaps"};
boolean found=false;
String SearchStr="Lapsi";
int index=0;
for (int i=0;i<StrArray.length;i++) {
    if (SearchStr.equals(StrArray[i])) {
        found=true; index=i;
    }
    if(found) {
        System.out.println(SearchStr + "U gjend tek indeksi "+index);
    }
    else
        System.out.println(SearchStr + " Nuk u gjend askund");
}
}

```

Ushtrimi 11.

Konvertoni nje StringArray ne nje ListArray

```

import java.util.ArrayList;

import java.util.Arrays;

import java.util.Collections;

import java.util.List;

```

```
public class Tester {  
    public static void main(String args[]) {  
        String[] array = {"a", "b", "c", "d", "e"};  
  
        //Method 1  
  
        List<String> list = Arrays.asList(array);  
  
        System.out.println(list);  
  
        //Method 2  
  
        List<String> list1 = new ArrayList<String>();  
  
        Collections.addAll(list1, array);  
  
        System.out.println(list1);  
  
        //Method 3  
  
        List<String> list2 = new ArrayList<String>();  
  
        for(String text:array) {  
            list2.add(text); }  
  
        System.out.println(list2); } }
```