Set

“Set” este o colectie ce nu poate contine elemente duplicate, modeland abstractia setului matematic.

Interfata “Set” contine doar metodele mostenite din Collection adaugand si restrictia ca elementele duplicate sa fie interzise.

“Set” adauga si un contract mai puternic intre comportamentul operatiilor “equals” si “hashCode”, permitand instantelor de ‘Set’ sa fie comparate corect indiferent de modul lor de implementare.

O parte din metodele declarate de set :

* add()

- Adauga un obiect

* clear()

- Sterge toate obiectele

* contains()

- Returneaza ‘true’ daca un anumit obiect este element al unei colectii

* isEmpty()

- Returneaza ‘true’ dacavcolectia nu are elemente

* iterator()

- Returneaza un obiect “Iterator” pentru colectie

* remove()

- Sterge obiectul specificat

* size()

- Returneaza numarul de elemente

Set are in implementarea sa diverse clase cum ar fi HashSet, TreeSet, LinkedHashSet.

import java.util.\*;

public class SetDemo {

public static void main(String args[]) {

int count[] = {34,22,10,60,30,22};

Set<Integer> set = new HashSet<Integer>();

try {

for(int i = 0; i < 6; i++) {

set.add(count[i]);

}

System.out.println(set);

TreeSet sortedSet = new TreeSet<Integer>(set);

System.out.println("The sorted list is:");

System.out.println(sortedSet);

System.out.println("The First element of the set is: "+ (Integer)sortedSet.first());

System.out.println("The last element of the set is: "+ (Integer)sortedSet.last());

}

catch(Exception e) {}

}

}

Output :

[34, 22, 10, 60, 30]

The sorted list is:

[10, 22, 30, 34, 60]

The First element of the set is: 10

The last element of the set is: 60