1. **COMPARATOR PROGRAM**

**package** comparator;

**import** java.util.\*;

**class** Student **implements** Comparator<Student>, Comparable<Student> {

**private** String name;

**private** **int** rollno;

Student() {

}

Student(String name, **int** rollno) {

**this**.name = name;

**this**.rollno = rollno;

}

**public** String getName() {

**return** name;

}

**public** **int** getRollno() {

**return** rollno;

}

**public** **int** compareTo(Student s) {

**return** (**this**.name).compareTo(s.name);

}

**public** **int** compare(Student s, Student s1) {

**return** s.rollno - s1.rollno;

}

}

**public** **class** ComparatorDemo

{

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

List<Student> list = **new** ArrayList<Student>();

list.add(**new** Student("Cherry",20));

list.add(**new** Student("Sahasra",25));

Collections.*sort*(list);

**for**(Student s3: list)

System.***out***.print(s3.getName() + ", ");

Collections.*sort*(list, **new** Student());

System.***out***.println(" ");

**for**(Student s3: list)

System.***out***.print(s3.getName() +" : "+ s3.getRollno() + ", ");

}

}

OUTPUT:

Cherry, Sahasra,

Cherry : 20, Sahasra : 25,

**2)STACK PROGRAM**

**package** stackprogram;

**public** **class** Stack {

**private** **int** maxSize;

**private** **long**[] stackArray;

**private** **int** top;

**public** MyStack(**int** s) {

maxSize = s;

stackArray = **new** **long**[maxSize];

top = -1;

}

**public** **void** push(**long** j) {

stackArray[++top] = j;

}

**public** **long** pop() {

**return** stackArray[top--];

}

**public** **long** peek() {

**return** stackArray[top];

}

**public** **boolean** isEmpty() {

**return** (top == -1);

}

**public** **boolean** isFull() {

**return** (top == maxSize - 1);

}

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

MyStack theStack = **new** MyStack(100);

theStack.push(100);

theStack.push(200);

theStack.push(300);

theStack.push(400);

theStack.push(500);

**while** (!theStack.isEmpty()) {

**long** value = theStack.pop();

System.***out***.print(value);

System.***out***.print(" ");

}

System.***out***.println("");

}

}

OUTPUT: 500 400 300 200 100

1. **CONCURRENT HASHMAP**

package concurrent;

import java.util.Map;

import java.util.concurrent.ConcurrentHashMap;

public class ConcurrentHashMapExample {

public static void main(String[] args) {

ConcurrentHashMap concurrentHashMap = new ConcurrentHashMap();

concurrentHashMap.put("N","Nokia");

concurrentHashMap.put("S","Samsung");

for (Map.Entry e : concurrentHashMap.entrySet()) {

System.out.println(e.getKey() + " = " + e.getValue());

}

}

}

1. **COMPARABLE PROGRAM**

**package** comparable;

**class** Employee **implements** Comparable<Employee>{

**int** empno;

String empname;

**int** empage;

Employee(String empname,**int** empno,**int** empage){

**this**.empno=empno;

**this**.empname=empname;

**this**.empage=empage;

}

**public** **int** compareTo(Employee ey){

**if**(empage==ey.empage)

**return** 0;

**else** **if**(empage>ey.empage)

**return** 1;

**else**

**return** -1;

}

}

**package** comparable;

**import** java.util.\*;

**import** java.io.\*;

**public** **class** Sort{

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

ArrayList<Employee> al=**new** ArrayList<Employee>();

al.add(**new** Employee("Rahul",213,45));

al.add(**new** Employee("Rohan",427,48));

al.add(**new** Employee("Roshan",659,47));

Collections.*sort*(al);

**for**(Employee ey:al){

System.***out***.println(ey.empname+" "+ey.empno+" "+ey.empage);

}

}

}

OUTPUT:

Rahul 213 45

Roshan 659 47

Rohan 427 48

1. **QUEUE PROGRAM**

import java.util.LinkedList.\*;

import java.util.Queue;

public class Queue{

public static void main(String[] args)

{

Queue queue=new LinkedList();

queue.add("first");

queue.offer("second");

queue.offer("third");

system.out.println("Queue Print::" +que);

String head=que.element();

system.out.println("Head element::" +head);

String element1=que.poll();

system.out.println("removed element::" +element1);

system.out.println("queue print after poll:: +que");

}

}

1. **Try with resource**

**private** **static** **void** printFileJava7() **throws** IOException {

**try**(FileInputStream input = **new** FileInputStream("file.txt")) {

**int** data = input.read();

**while**(data != -1){

System.out.print((**char**) data);

data = input.read();

}

}

}

**package** userdefined;

**class** MyException **extends** Exception {

String s1;

MyException(String s2) {

s1 = s2;

}

@Override

**public** String toString() {

**return** ("Output String = "+s1);

}

}

**public** **class** Programs{

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

**try** {

**throw** **new** MyException("user defined message");

} **catch**(MyException exp) {

System.***out***.println(exp);

}

}

}