Slip-1

Q 1) Write a Program to print all Prime numbers in an array of 'n' elements. (use command line arguments).

[10 Marks]

```
import java.util.*;
public class slippract
{
public static void main(String[] args)
{
int n[]=new int[args.length];
for(int i=0;i<args.length;i++)
{
n[i]=Integer.parseInt(args[i]);
}
for(int i=0;i<args.length;i++)
{
int num=n[i];
if(num<2)
continue;
int flag=1;
for(int j=2;j<num;j++)
{
if(num%j==0)
{
flag=0;
break;
}
}
if(flag==1)
        System.out.println(num);
}
}
}
```

Q 2) Define a class CricketPlayer (name,no_of_innings,no_of_times_notout, totatruns, bat_avg). Create an array of n player objects .Calculate the batting average for each player using static method avg(). Define a static sort method which sorts the array on the basis of average. Display the player details in sorted order. [15 Marks]

[1 Marks]

```
import java.util.*;
class cricketplayer
{
int no_of_innings,no_of_times_notout, totalruns,batavg;
String name;
Scanner sc=new Scanner(System.in);
void accept()
{
System.out.println("Enter cricket player name no of innings no of times not out total run= ");
name=sc.next();
no_of_innings=sc.nextInt();
no_of_times_notout=sc.nextInt();
totalruns=sc.nextInt();
}
void disp()
{
System.out.println("CricketPlayer name= "+name);
System.out.println("no_of_innings= "+no_of_innings);
System.out.println("no_of_times_notout= "+no_of_times_notout);
System.out.println("Totalrunt= "+totalruns);
System.out.println("bat average= "+batavg);
}
static void avg(cricketplayer c[],int n)
{
int i;
for(i=0;i<n;i++)
c[i].batavg=c[i].totalruns/(c[i].no_of_innings-c[i].no_of_times_notout);
}
static void sort(cricketplayer c[],int n)
{
int pass,i;
for(pass=1;pass<n;pass++)
{
for(i=0;i<n-pass;i++)
{
if(c[i].batavg>c[i+1].batavg)
{
cricketplayer t=c[i];
c[i]=c[i+1];
c[i+1]=t;
}
}
}
}
}
```

[2 Marks]

```
public class slippract
{
public static void main(String[] args)
{
int n,i;
cricketplayer c[]=new cricketplayer[20];
Scanner sc=new Scanner(System.in);
System.out.println("Enter no of players= ");
n=sc.nextInt();
for(i=0;i<n;i++)
{
c[i]=new cricketplayer();
c[i].accept();
}
cricketplayer.avg(c,n);
cricketplayer.sort(c, n);
for(i=0;i<n;i++)
{
 c[i].disp();
}
}
}
```

Slip-2

Q 1) Write a program for multilevel inheritance such that Country is inherited from Continent. State is inherited from Country. Display the place, State, Country and Continent. [10 Marks]

```
import java.util.*;
class continent
{
Scanner sc=new Scanner(System.in);
String cname;
continent()
{
System.out.println("Enter continennt name= ");
cname=sc.next();
}
```

[3 Marks]

```
}
class country extends continent
{
String cn;
country()
{
System.out.println("Enter country name= ");
cn=sc.next();
}
}
class state extends country
{
String st;
state()
{
System.out.println("Enter state name= ");
st=sc.next();
}
}
class place extends state
{
String pn;
place()
{
System.out.println("Enter place name= ");
pn=sc.next();
}
void disp()
{
System.out.println("Continent name= "+cname);
System.out.println("Countryu name= "+cn);
System.out.println("State name= "+st);
System.out.println("place name= "+pn);
}
}
public class slippract
{
public static void main(String[] args)
{
place p=new place();
p.disp();
}
}
```

[4 Marks]

Q 2) Create an abstract class “order” having members id, description. Create two subclasses “PurchaseOrder” and “Sales Order” having members customer name and Vendor name respectively. Definemethods accept and display in all cases. Create 3 objects each of Purchase
Order and Sales Order and accept and display details [15 Marks]

```
import java.util.*;
abstract class order
{
int id;
String des;
abstract void accept();
abstract void disp();
Scanner sc=new Scanner(System.in);
class purchaseorder extends order
{
String cname;
void accept()
{
System.out.println("Enter member id and des= ");
id=sc.nextInt();
des=sc.next();
System.out.println("Enter customer name= ");
cname=sc.next();
}
void disp()
{
System.out.println("Order id= "+id);
System.out.println("Orderdescription= "+des);
System.out.println("Customer name= "+cname);
}
}
class salesorder extends order
{
String vname;
void accept()
{
System.out.println("Enter member id and des= ");
id=sc.nextInt();
des=sc.next();
System.out.println("Enter customer name= ");
vname=sc.next();
}
```

[5 Marks]

```
void disp()
{
System.out.println("Order id= "+id);
System.out.println("Orderdescription= "+des);
System.out.println("Vendor name= "+vname);
}
}
}
    public class slippract
{
public static void main(String[] args)
{
purchaseorder p[]=new purchaseorder[3];
salesorder s[]=new salesorder[3];
for(int i=0;i<3;i++)
{
p[i]=new purchaseorder();
p[i].accept();
}
for(int i=0;i<3;i++)
{
s[i]=new salesorder();
s[i].accept();
}
for(int i=0;i<3;i++)
{
p[i].disp();
}
for(int i=0;i<3;i++)
{
s[i].disp();
}
    }
    }
```

SLIP -03

Q 1) Write a program to display the Employee(Empid, Empname, Empdesignation, Empsal) information using toString(). [10 Marks]

```
import java.util.*;
class Employee
{
private int empid;
private String empname;
```

[6 Marks]

```
private String empdesignation;
private double empsal;
public Employee(int empid, String empname, String empdesignation, double empsal) {
this.empid = empid;
this.empname = empname;
this.empdesignation = empdesignation;
this.empsal = empsal;
}
public String toString() {
return "Employee ID: " + empid + "\n" +
"Employee Name: " + empname + "\n" +
"Employee Designation: " + empdesignation + "\n" +
"Employee Salary: " + empsal;
}
}
public class slippract
{
public static void main(String[] args)
{
Scanner sc=new Scanner(System.in);
int empid;
String empname;
String empdes;
double empsal;
System.out.println("Enter emp id name des sal= ");
empid=sc.nextInt();
empname=sc.next();
empdes=sc.next();
empsal=sc.nextDouble();
Employee emp1 = new Employee(101, "John Doe", "Software Developer", 75000.50);
System.out.println(emp1.toString());
}
                    }
```

Q 2) Write a program to create a class Product (product_id, product_name, product_cost, product_quantity) default and parameterized constructor. Create objectsof class product and display the contents of each object and Also display the object count. [15 Marks]

```
 import java.util.*;
class product
{
int pid,pcost, pqu;
```

[7 Marks]

```
static int cnt=0;
String pname;
product()
{
pid=0;
pname="";
pcost=0;
pqu=0;
}
product(int pid,String pname,int pcost,int pqu)
{
this.pid=pid;
this.pname=pname;
this.pcost=pcost;
this.pqu=pqu;
cnt++;
}
void disp()
{
System.out.println("Product id= "+pid);
System.out.println("Product name= "+pname);
System.out.println("Product cost= "+pcost);
System.out.println("Product quantity= "+pqu);
}
}
public class slippract
{
public static void main(String[] args)
{
Scanner sc=new Scanner(System.in);
int pid,pcost, pqu;
String pname;
product p[]=new product[20];
int n,i;
System.out.println("Enter no of product= ");
n=sc.nextInt();
for(i=0;i<n;i++)
{
System.out.println("Enter product id name cost and quantity= ");
pid=sc.nextInt();
pname=sc.next();
pcost=sc.nextInt();
pqu=sc.nextInt();
p[i]=new product(pid,pname,pcost,pqu);
}
```

[8 Marks]

```
for(i=0;i<n;i++)
{
p[i].disp();
}
System.out.println("No of object count= "+product.cnt);
}
}
```

SLIP -4

Q 1) Write a Program to print all even numbers in an array of 'n' elements. (use command line arguments). [10 Marks]

```
import java.util.*;
public class slippract
{
public static void main(String[] args)
{
int n[]=new int[args.length];
for(int i=0;i<args.length;i++)
{
n[i]=Integer.parseInt(args[i]);
}
for(int i=0;i<args.length;i++)
{
if(n[i]%2==0)
System.out.println(n[i]);
}
}
}
```

Q 2) Define an abstract class Staff with protected members id and name. Define a parameterized constructor. Define one subclass OfficeStaff with member department. Create n objects of OfficeStaff and display all details. [15 Marks]

```
import java.util.*;
abstract class staff
{
protected int id;
protected String name;
```

[9 Marks]

```
staff(int id,String name)
{
this.id=id;
this.name=name;
}
}
class officestaff extends staff
{
String dep;
officestaff(int id,String name,String dep)
{
super(id,name);
this.dep=dep;
}
void disp()
{
System.out.println("Staff id= "+id);
System.out.println("Staff name= "+name);
System.out.println("Staff department= "+dep);
}
}
public class slippract
{
public static void main(String[] args)
{
int id;
String name;
String dep;
officestaff ob[]=new officestaff[20];
int n,i;
Scanner sc=new Scanner(System.in);
System.out.println("Enter no of staff members= ");
n=sc.nextInt();
for(i=0;i<n;i++)
{
System.out.println("Enter staff id name and department=");
id=sc.nextInt();
name=sc.next();
dep=sc.next();
ob[i]=new officestaff(id,name,dep);
}
for(i=0;i<n;i++)
{
ob[i].disp();
}
```

[10 Marks]

```
}
        }
```

SLIP 5

Q 1) Create a class Sphere, to calculate the volume and surface area of sphere. (Hint : Surface area=4*3.14(r*r), Volume=(4/3)3.14(r*r*r)) [10 Marks]

```
import java.util.*;
public class slippract
{
public static void main(String[] args)
{
float r;
Scanner sc=new Scanner(System.in);
System.out.println("Enter ardius of sphere= ");
r=sc.nextFloat();
float a=4*3.14f*(r*r);
float v=(4/3)3.14f(r*r*r);
System.out.println("Area of sphere= "+a);
System.out.println("Volume of sphere= "+v);
}
        }
```

Q 2) Define Student class(roll_no, name, percentage) to create n objects of the Student class. Accept details from the user for each object. Define a static method “sortStudent” which sorts the array on the basis of percentage. [15 Marks]

```
import java.util.*;
class student
{
int sid,per;
String name;
Scanner sc=new Scanner(System.in);
void accept()
{
System.out.println("Enter stud sid name per= ");
sid=sc.nextInt();
name=sc.next();
per=sc.nextInt();
}
void disp()
```

[11 Marks]

```
{
System.out.println("Student id= "+sid);
System.out.println("Student name= "+name);
System.out.println("Student per= "+per);
}
static void sort(student s[],int n)
{
int i,j;
for(i=1;i<n;i++)
{
for(j=0;j<n-i;j++)
{
if(s[j].per>s[j+1].per)
{
student t=s[j];
s[j]=s[j+1];
s[j+1]=t;
}
}
}
}
}
public class slippract
{
public static void main(String[] args)
{
Scanner sc=new Scanner(System.in);
student s[]=new student[20];
int n,i;
System.out.println("Enter no student= ");
n=sc.nextInt();
for(i=0;i<n;i++)
{
s[i]=new student();
s[i].accept();
}
student.sort(s, n);
for(i=0;i<n;i++)
{
s[i].disp();
}
}
}
```

[12 Marks]

Slip-6

Q 1) Write a program to find the cube of given number using functional interface. [10

```
import java.util.*;
interface cube
{
public void cube(int n);
}
public class slippract
{
public static void main(String[] args)
{
cube c=x->System.out.println("Cube ="+x*x*x);
c.cube(2);
}
}
```

Q 2) Write a menu driven program to perform the following operations

a. Calculate the volume of cylinder. (hint : Volume: $\pi \times r^2 \times h$)
b. Find the factorial of given number.
c. Check the number is Armstrong or not.
d. Exit [15 Marks]

```
import java.util.*;
public class slippract
{
public static void main(String[] args)
{
int i,ch;
Scanner sc=new Scanner(System.in);
do
{
System.out.println("Enter your chioce= 1-cylinder 2-factorial 3-Armstrong 3-Exit= ");
ch=sc.nextInt();
switch(ch)
{
case 1:float v;
```

[13 Marks]

```
int r,h;
System.out.println("Enter radius and volumne of cylinder= ");
r=sc.nextInt();
h=sc.nextInt();
v=3.14f*r*r*h;
System.out.println("Volumne of cylinder= "+v);
break;
case 2:int fact=1;
int num;
System.out.println("Enter a number= ");
num=sc.nextInt();
for(i=1;i<=num;i++)
{
fact=fact*i;
}
System.out.println("Factorial= "+fact);
break;
case 3:int num1,no1,rem,sum=0;
System.out.println("Enter a number= ");
num1=sc.nextInt();
no1=num1;
while(no1!=0)
{
rem=no1%10;
sum=sum+(rem*rem*rem);
no1=no1/10;
}
if(sum==num1)
System.out.println("Armstrong no");
else
System.out.println("Not Armstrong no");
}
}while(ch<4);
}
}
```

Slip-7

Q 1) Write a program to accept the array element and display in reverse order. [10 Marks]

```
import java.util.*;
public class slippract
{
```

[14 Marks]

```
public static void main(String[] args)
{
Scanner sc=new Scanner(System.in);
int a[]=new int[20];
int n,i;
System.out.println("Enter no of numbers= ");
n=sc.nextInt();
System.out.println("Enter array elements= ");
for(i=0;i<n;i++)
{
a[i]=sc.nextInt();
}
System.out.println("Reverse array elements= ");
for(i=n-1;i>=0;i--)
{
System.out.println(a[i]);
}
}
```

}

Q 2) Create an employee class(id,name,deptname,salary). Define a default and parameterized constructor. Use 'this' keyword to initialize instance variables. Keep a count of objects created. Create objects using parameterized constructor and display the object count after each object is created.(Use static member and method). Also display the contents of each object.

[15 Marks]

```
import java.util.*;
class employee
{
int id,sal;
String dname,name;
static int cnt=0;
employee()
{
id=0;
name="";
dname="";
sal=0;
}
employee(int id,String name,String dname,int sal)
{
this.id=id;
this.name=name;
```

[15 Marks]

```
this.dname=dname;
this.sal=sal;
cnt++;
}
static int count()
{
return cnt++;
}
void disp()
{
System.out.println("Emp id= "+id);
System.out.println("Emp name= "+name);
System.out.println("Emp dname= "+dname);
System.out.println("Emp sal= "+sal);
}
}
public class slippract
{
public static void main(String[] args)
{
Scanner sc=new Scanner(System.in);
int n,i;
int id,sal;
String dname,name;
employee e[]=new employee[20];
System.out.println("Enter no of emp= ");
n=sc.nextInt();
for(i=0;i<n;i++)
{
System.out.println("Enter emp id name dep sal= ");
id=sc.nextInt();
name=sc.next();
dname=sc.next();
sal=sc.nextInt();
e[i]=new employee(id,name,dname,sal);
}
for(i=0;i<n;i++)
{
e[i].disp();
}
System.out.println("No of emp count= "+employee.count());
}
}
```

[16 Marks]

Slip-8

Q 1) Write a Program to print all odd numbers in an array of 'n' elements. (use command line arguments). [10 Marks]

```
import java.util.*;
public class slippract
{
public static void main(String[] args)
{
int n[]=new int[args.length];
for(int i=0;i<args.length;i++)
{
n[i]=Integer.parseInt(args[i]);
}
for(int i=0;i<args.length;i++)
{
if(n[i]%2!=0)
System.out.println(n[i]);
}
}
}
```

Q 2) Create an abstract class "order" having members id, description. Create two subclasses "PurchaseOrder" and "Sales Order" having members customer name and Vendor name respectively. Definemethods accept and display in all cases. Create 3 objects each of Purchase
Order and Sales Order and accept and display details [15 Marks]

```
import java.util.*;
abstract class order
{
int id;
String des;
abstract void accept();
abstract void disp();
Scanner sc=new Scanner(System.in);
class purchaseorder extends order
{
```

[17 Marks]

```
String cname;
void accept()
{
System.out.println("Enter member id and des= ");
id=sc.nextInt();
des=sc.next();
System.out.println("Enter customer name= ");
cname=sc.next();
}
void disp()
{
System.out.println("Order id= "+id);
System.out.println("Orderdescription= "+des);
System.out.println("Customer name= "+cname);
}
}
class salesorder extends order
{
String vname;
void accept()
{
System.out.println("Enter member id and des= ");
id=sc.nextInt();
des=sc.next();
System.out.println("Enter customer name= ");
vname=sc.next();
}
void disp()
{
System.out.println("Order id= "+id);
System.out.println("Orderdescription= "+des);
System.out.println("Vendor name= "+vname);
}
}
}
    public class slippract
{
public static void main(String[] args)
{
purchaseorder p[]=new purchaseorder[3];
salesorder s[]=new salesorder[3];
for(int i=0;i<3;i++)
{
p[i]=new purchaseorder();
p[i].accept();
```

[18 Marks]

```
}
for(int i=0;i<3;i++)
{
s[i]=new salesorder();
s[i].accept();
}
for(int i=0;i<3;i++)
{
p[i].disp();
}
for(int i=0;i<3;i++)
{
s[i].disp();
}
    }
    }
```

Slip-9

Q 1 Write a program to accept 'n' name of cities from the user and sort them in ascending order.

[10 Marks]

```
import java.util.*;
public class slippract
{
public static void main(String[] args)
{
Scanner sc=new Scanner(System.in);
String name[]=new String[20];
int i,n;
System.out.println("Enter no of cities= ");
n=sc.nextInt();
System.out.println("Enter name of cities= ");
for(i=0;i<n;i++)
{
name[i]=sc.next();
}
for(int pass=1;pass<n;pass++)
{
for(i=0;i<n-pass;i++)
{
if(name[i].compareTo(name[i+1])>0)
{
```

[19 Marks]

```
String t=name[i];
name[i]=name[i+1];
name[i+1]=t;
}
}
}
System.out.println("Sorted names=");
for(i=0;i<n;i++)
{
System.out.println(name[i]);
}
}
}
```

Q 2) Define a class patient (patient_name, patient_age, patient_oxy_level,patient_HRCT_report). Create an object of patient. Handle appropriate exception while patient oxygen level less than 95% and HRCT scan report greater than 10, then throw user defined Exception “Patient is CovidPositive(+) and Need to Hospitalized” otherwise display its information. [15 Marks]

```
import java.util.*;
class covid extends Exception
{
public covid()
{
System.out.println("Patient is Covid Positive(+) and Need to Hospitalized");
}
}
class patient
{
String patient_name;
int patient_age, patient_oxy_level,patient_HRCT_report;
patient(String patient_name,int patient_age, int patient_oxy_level,int patient_HRCT_report)
{
this.patient_name=patient_name;
this.patient_age=patient_age;
this.patient_oxy_level=patient_oxy_level;
this.patient_HRCT_report=patient_HRCT_report;
}
}
public class slippract
{
public static void main(String[] args)
{
Scanner sc=new Scanner(System.in);
```

[20 Marks]

```
String patient_name;
int patient_age, patient_oxy_level,patient_HRCT_report;
System.out.println("Enter patient name,int patient_age, int patient_oxy_level,int patient_HRCT_report=");
patient_name=sc.next();
patient_age=sc.nextInt();
patient_oxy_level=sc.nextInt();
patient_HRCT_report=sc.nextInt();
patient p=new patient(patient_name,patient_age, patient_oxy_level,patient_HRCT_report);
try
{
if(patient_oxy_level<95 || patient_HRCT_report>10)
throw new covid();
else
{
System.out.println("Patient name= "+patient_name);
System.out.println("patient_age= "+patient_age);
System.out.println("patient_oxy_level= "+patient_oxy_level);
System.out.println("patient_HRCT_report= "+patient_HRCT_report);
}
}catch(covid e2)
{
}
}
}
```

Slip-10

Q 1) Create a text file and display the file information such as name,directory name,path,absolute path,lastmodified time,length of file using methods. [10 Marks]

```
import java.io.File;

import java.io.IOException;

import java.text.SimpleDateFormat;

import java.util.Date;


public class FileInfoDisplay {
```

[21 Marks]

```
public static void main(String[] args) {
    String fileName = "myTextFile.txt"; // Name of the file to create

    // 1. Create a File object
    File file = new File(fileName);

    try {
        // 2. Create the new file
        if (file.createNewFile()) {
            System.out.println("File created successfully: " + file.getName());
        } else {
            System.out.println("File already exists: " + file.getName());
        }

        // 3. Display file information
        System.out.println("\n--- File Information ---");
        System.out.println("File Name: " + file.getName());
        System.out.println("Directory Name: " + file.getParent()); // Returns null if in current directory
        System.out.println("Path: " + file.getPath());
        System.out.println("Absolute Path: " + file.getAbsolutePath());

        // Get last modified time and format it
```

[22 Marks]

```
        long lastModifiedMillis = file.lastModified();
        Date lastModifiedDate = new Date(lastModifiedMillis);
        SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");
        System.out.println("Last Modified Time: " + sdf.format(lastModifiedDate));

        System.out.println("Length of File (bytes): " + file.length());

      } catch (IOException e) {
        System.err.println("An error occurred: " + e.getMessage());
      }
    }
}
```

Q 2) Define a Item class (item_number, item_name, item_price). Define a default and parameterized constructor. Keep a count of objects created. Create objects using parameterized constructor and display the object count after each object is created. (Use static member and method). Also display the contents of each object. [15 Marks]

```
import java.util.*;
class item
{
int ino,iprice;
String iname;
static int cnt=0;
item()
{
ino=0;
iname="";
iprice=0;
}
item(int ino,String iname,int iprice)
```

[23 Marks]

```
{
this.ino=ino;
this.iname=iname;
this.iprice=iprice;
cnt++;
      }
static int count()
{
return cnt;
}
void disp()
{
System.out.println("Item no= "+ino);
System.out.println("Item name= "+iname);
System.out.println("Item price= "+iprice);
}
}
public class slippract
{
public static void main(String[] args)
{
item ob[]=new item[20];
int n,i;
int ino,iprice;
String iname;
Scanner sc=new Scanner(System.in);
System.out.println("Enter no of items= ");
n=sc.nextInt();
for(i=0;i<n;i++)
{
System.out.println("Enter item no name price= ");
ino=sc.nextInt();
iname=sc.next();
iprice=sc.nextInt();
ob[i]=new item(ino,iname,iprice);
}
for(i=0;i<n;i++)
{
ob[i].disp();
}
System.out.println("Count of object="+item.count());
}

}
```

[24 Marks]

Slip-11

Q 1) Accept the names of two files and copy the contents of the first to the second. First file having Book name and Author name in file. [10 Marks]

```
import java.io.*;
import java.util.*;
class copy{
public static void main(String arg[]) throws Exception {
Scanner sc = new Scanner(System.in);
System.out.print("source file name :\n");
String file1 = sc.next();
System.out.print("destination file name :\n");
String file2 = sc.next();
FileReader fin = new FileReader(file1);
FileWriter fout = new FileWriter(file2, true);
int c;
while ((c = fin.read()) != -1) {
fout.write(c);
}
System.out.println("Copy finish...");

fin.close();
fout.close();
}
}
```

Q 2) Define a class CricketPlayer (name,no_of_innings,no_of_times_notout, totatruns, bat_avg).

Create an array of n player objects .Calculate the batting average for each player using static method avg(). Define a static sort method which sorts the array on the basis of average. Display the player details in sorted order.

```
import java.util.*;
class cricketplayer
{
int no_of_innings,no_of_times_notout, totalruns,batavg;
String name;
Scanner sc=new Scanner(System.in);
void accept()
{
```

[25 Marks]

```
System.out.println("Enter cricket player name no of innings no of times not out total run= ");
name=sc.next();
no_of_innings=sc.nextInt();
no_of_times_notout=sc.nextInt();
totalruns=sc.nextInt();
}
void disp()
{
System.out.println("CricketPlayer name= "+name);
System.out.println("no_of_innings= "+no_of_innings);
System.out.println("no_of_times_notout= "+no_of_times_notout);
System.out.println("Totalrunt= "+totalruns);
System.out.println("bat average= "+batavg);
}
static void avg(cricketplayer c[],int n)
{
int i;
for(i=0;i<n;i++)
c[i].batavg=c[i].totalruns/(c[i].no_of_innings-c[i].no_of_times_notout);
}
static void sort(cricketplayer c[],int n)
{
int pass,i;
for(pass=1;pass<n;pass++)
{
for(i=0;i<n-pass;i++)
{
if(c[i].batavg>c[i+1].batavg)
{
cricketplayer t=c[i];
c[i]=c[i+1];
c[i+1]=t;
}
}
}
}
}
public class slippract
{
public static void main(String[] args)
{
int n,i;
cricketplayer c[]=new cricketplayer[20];
Scanner sc=new Scanner(System.in);
System.out.println("Enter no of players= ");
```

[26 Marks]

```
n=sc.nextInt();
for(i=0;i<n;i++)
{
c[i]=new cricketplayer();
c[i].accept();
}
cricketplayer.avg(c,n);
cricketplayer.sort(c, n);
for(i=0;i<n;i++)
{
 c[i].disp();
}
}
}
```

SLIP 12

Q 1) Write a program for multilevel inheritance such that Country is inherited from Continent. State is inherited from Country. Display the place, State, Country and Continent. [10 Marks]

```
import java.util.*;
class continent
{
Scanner sc=new Scanner(System.in);
String cname;
continent()
{
System.out.println("Enter continennt name= ");
cname=sc.next();
}
}
class country extends continent
{
String cn;
country()
{
System.out.println("Enter country name= ");
cn=sc.next();
}
}
class state extends country
```

[27 Marks]

```
{
String st;
state()
{
System.out.println("Enter state name= ");
st=sc.next();
}
}
class place extends state
{
String pn;
place()
{
System.out.println("Enter place name= ");
pn=sc.next();
}
void disp()
{
System.out.println("Continent name= "+cname);
System.out.println("Countryu name= "+cn);
System.out.println("State name= "+st);
System.out.println("place name= "+pn);
}
}
public class slippract
{
public static void main(String[] args)
{
place p=new place();
p.disp();
}
}
```

Q 2) Write a menu driven program to perform the following operations on multidimensional array ie matrices :

- Addition
- Multiplication
- Exit

```
import java.util.*;
public class slippract
{
public static void main(String[] args)
```

[28 Marks]

```
{
Scanner sc=new Scanner(System.in);
int ch;
do
{
System.out.println("Enter your chioce=1-addition 2-multipilcation 3-Exit= ");

ch=sc.nextInt();
switch(ch)
{
case 1:int i,j;
int a[][]=new int[20][20];
int b[][]=new int[20][20];
int c[][]=new int[20][20];
System.out.println("Enter no of rows and cols= ");
int n=sc.nextInt();
System.out.println("Enter elements matrix1= ");
for(i=0;i<n;i++)
{
for(j=0;j<n;j++)
{
a[i][j]=sc.nextInt();
}
}
System.out.println("Enter elements matrix2= ");
for(i=0;i<n;i++)
{
for(j=0;j<n;j++)
{
b[i][j]=sc.nextInt();
}
}
for(i=0;i<n;i++)
{
for(j=0;j<n;j++)
{
c[i][j]=a[i][j]+b[i][j];
}
}
System.out.println("Addition= ");
for(i=0;i<n;i++)
{
for(j=0;j<n;j++)
{
```

[29 Marks]

```
System.out.println(c[i][j]);
}
}
break;
case 2:
int d[][]=new int[20][20];
int e[][]=new int[20][20];
int f[][]=new int[20][20];
System.out.println("Enter no of rows and cols= ");
int m=sc.nextInt();
System.out.println("Enter elements matrix1= ");
for(i=0;i<m;i++)
{
for(j=0;j<m;j++)
{
e[i][j]=sc.nextInt();
}
}
System.out.println("Enter elements matrix2= ");
for(i=0;i<m;i++)
{
for(j=0;j<m;j++)
{
f[i][j]=sc.nextInt();
}
}
for(i=0;i<m;i++)
{
for(j=0;j<m;j++)
{
d[i][j]=0;
for(int k=0;k<m;k++)
{
d[i][j]=d[i][j]+e[i][j]*f[k][j];
}
}
}
System.out.println("Multiplication of matrix= ");
for(i=0;i<m;i++)
{
for(j=0;j<m;j++)
{
System.out.println(d[i][j]);
}
}
```

[30 Marks]

```
}
}while(ch<4);
}
}
```

[31 Marks]

SLIP 13

Q 1) Write a program to calculate perimeter and area of rectangle.
(hint : area = length * breadth , perimeter=2*(length+breadth))

```
import java.util.*;
public class slippract
{
public static void main(String[] args)
{
int l,b;
Scanner sc=new Scanner(System.in);
System.out.println("Enter length and breadth= ");
l=sc.nextInt();
b=sc.nextInt();
int a=l*b;
int p=2*(l+b);
System.out.println("Area of rectangle= "+a);
System.out.println("Perimeter of rectangle= "+p);
}
}
```

Q 2) Write a java program to display the system date and time in various formats shown below: Current date is : 31/08/2021

Current date is : 08-31-2021
Current date is : Tuesday August 31 2021
Current date and time is : Fri August 31 15:25:59 IST 2021
Current date and time is : 31/08/21 15:25:59 PM +0530
Current time is : 15:25:59
Current week of year is : 35
Current week of month : 5
Current day of the year is : 243
Note: Use java.util.Date and java.text.SimpleDateFormat class

```
import java.text.SimpleDateFormat;
public class slippract
```

[32 Marks]

```
{
public static void main(String[] args)
{
Date d=new Date();
SimpleDateFormat sd=new SimpleDateFormat("dd/mm/yyyy");
String date=sd.format(d);
System.out.println("Current date is:"+date);
sd=new SimpleDateFormat("MM-dd-yyyy");
date=sd.format(d);
System.out.println("Current date is:"+date);
sd=new SimpleDateFormat("EEE MMM-dd-yyyy");
date=sd.format(d);
System.out.println("Current date is:"+date);
sd=new SimpleDateFormat("EEE MMM-dd-yyyy HH:mm:ss z");
        date=sd.format(d);
System.out.println("Current date and time is:"+date);
sd=new SimpleDateFormat("dd/MM/yyyy HH:mm:ss z");
date=sd.format(d);
System.out.println("Current date and time is:"+date);
sd=new SimpleDateFormat("HH:mm:ss");
date=sd.format(d);
System.out.println("Current time is:"+date);
sd=new SimpleDateFormat("w");
date=sd.format(d);
System.out.println("Current week of year is:"+date);
sd=new SimpleDateFormat("W");
date=sd.format(d);
System.out.println("Current week of month is:"+date);
sd=new SimpleDateFormat("d");
date=sd.format(d);
System.out.println("Current day of the year is:"+date);
}
        }
```

Slip-14

Q 1) Write a program to find the square of given number using functional

[33 Marks]

interface. [10 Marks]

```
import java.util.*;
interface square
{
public void square(int n);
}
public class slippract
{
public static void main(String[] args)
{
square s=x->System.out.println("Square ="+x*x);
s.square(2);
}
}
```

Q 2) Define a class MyNumber having one private int data member. Write a default constructor to initialize it to 0 and another constructor to initialize it to a value (Use this). Write methods
isNegative, isPositive, isZero, isOdd, isEven. Create an object in main. Use command line
arguments to pass a value to the object . [15 Marks]

```
 import java.util.*;
class Mynumber
{
private int a;
Mynumber()
{
a=0;
}
Mynumber(int a)
{
this.a=a;
}
void isNegative()
{
if(a<0)
System.out.println("Number is negative");
```

[34 Marks]

```
}
void isPositive()
{
if(a>0)
System.out.println("Number is positive");
}
void isZero()
{
if(a==0)
System.out.println("Number is zero");
}
    void isEven()
{
if(a%2==0)
System.out.println("Number is even");
}
void isOdd()
{
if(a%2!=0)
System.out.println("Number is odd");
}
}
public class slippract
{
public static void main(String[] args)
{
int no=Integer.parseInt(args[0]);
Mynumber ob=new Mynumber(no);
ob.isNegative();
ob.isPositive();
ob.isEven();
ob.isOdd();
ob.isZero();
}
    }
```

Slip-15

Q 1) Write a java program to accept 5 numbers using command line arguments sort and display

[35 Marks]

them. [10 Marks]

```
public class slippract
{
public static void main(String[] args)
{
int t;
int a[]=new int[5];
a[0]=Integer.parseInt(args[0]);
a[1]=Integer.parseInt(args[1]);
a[2]=Integer.parseInt(args[2]);
a[3]=Integer.parseInt(args[3]);
a[4]=Integer.parseInt(args[4]);
System.out.println("Numbers before sorting= ");
for(int i=0;i<5;i++)
{
System.out.println(a[i]);
}
for(int i=0;i<5;i++)
{
for(int j=0;j<i;j++)
{
if(a[i]<a[j])
{
t=a[i];
a[i]=a[j];
a[j]=t;
}
}
}
System.out.println("Numbers after sorting= ");
for(int i=0;i<5;i++)
{
System.out.println(a[i]);
}
}
}
```

Q 2) Write a Java program to create a Package “SY” which has a class SYMarks (members – ComputerTotal, MathsTotal, and ElectronicsTotal). Create another package TY which has a class TYMarks (members – Theory,

[36 Marks]

Practicals). Create n objects of Student class (having rollNumber, name, SYMarks and TYMarks). Add the marks of SY and TY computer subjects and calculate the Grade ('A' for >= 70, 'B' for >= 60 'C' for >= 50 , Pass Class for > =40 else 'FAIL') and display the result of the student in proper format

```
import java.util.*;
public class symarks
{
public int ct,mt,et;
public void get()
{
Scanner sc = new Scanner(System.in);
ct=sc.nextInt();
mt=sc.nextInt();
et=sc.nextInt();
}
}
package ty;
import java.util.*;
public class tymarks
{
public int tm,pm;
public void get()
{
Scanner sc = new Scanner(System.in);
tm=sc.nextInt();
pm=sc.nextInt();
}
}
package Operation;
import sy.*;
import ty.*;
import java.util.*;
public class student
{
int rno,total;
String name,grade;
float per;
public void get()
{
```

[37 Marks]

```
Scanner sc=new Scanner("System.in");
rno=sc.nextInt();
name=sc.next();
}
public static void main(String[] args)
{
Scanner sc=new Scanner("System.in");
System.out.println("Enter limit= ");
int n=sc.nextInt();
student s[]=new student[n];
symarks s1[]=new symarks[n];
tymarks t[]=new tymarks[n];
for(int i=0;i<n;i++)
{
System.out.println("Enter rno and name= ");
s[i]=new student();
s[i].get();
System.out.println("Enter marks of computer,math and electronics= ");
s1[i]=new symarks();
s1[i].get();
System.out.println("Enter marks of theory and practical = ");
t[i]=new tymarks();
t[i].get();
s[i].total=s1[i].ct+t[i].pm+t[i].tm;
}
System.out.println("Rno\tName\tPer\tGrade");
System.out.println("=============================================================");
for(int i=0;i<n;i++)
{
s[i].per=s[i].total/3;
if(s[i].per>=70)
s[i].grade="A";
else if(s[i].per>=60)
s[i].grade="B";
else if(s[i].per>=50)
s[i].grade="C";
else if(s[i].per>=40)
s[i].grade="PASS";
else
s[i].grade="FAIL";
```

[38 Marks]

```
System.out.println(+s[i].rno+"\t"+s[i].name+"\t"+s[i].per+"\t"+s[i].grade);
}
}
}
```

Slip-16

Q 1) Write a package for String operation which has two classes Con and Comp. Con class has to concatenates two strings and comp class compares two strings. Also display proper message on execution.

```
package makone;
public class compare
{
String s1,s2;
compare(String s1,String s2)
{
this.s1=s1;
this.s2=s2;
}
void compare()
{
if(s1.equals(s2))
System.out.println("String are equal");
else
System.out.println("String are not equal");
}
}
/class concate/
package makone;
public class concate
{
String s1,s2;
```

[39 Marks]

```
concate(String s1,String s2)
{
this.s1=s1;
this.s2=s2;
}
void concate()
{
String s3;
s3=s1.concat(s2);
System.out.println("Concatnate string- "+s3);
}
}

/String class/
package makone;
import java.util.*;
public class string {
public static void main(String[] args)
{
Scanner sc=new Scanner(System.in);
String s1,s2;
System.out.println("Enter string 1 and string 2= ");
s1=sc.next();
s2=sc.next();
concate c1=new concate(s1,s2);
c1.concate();
compare c2=new compare(s1,s2);
c2.compare();
}
}
```

Q 2) Define class EmailId with members ,username and password. Define default and parameterized constructors. Accept values from the command line Throw user defined exceptions –
"InvalidUsernameException" or "InvalidPasswordException" if the username and password are

[40 Marks]

Invalid

```
package makone;
import java.util.*;
class name extends Exception
{
public name()
{
System.out.println("InvalidUsernameException");
}
}
class password extends Exception
{
public password()
{
System.out.println("InvalidPasswordException");
}
}
class Emailid
{
String name;
int password;
Emailid()
{
name="";
password=0;
}
Emailid(String name,int password)
{
this.name=name;
this.password=password;
}
}
public class slippract
{
public static void main(String[] args)
{
String name;
int password;
name=args[0];
password=Integer.parseInt(args[1]);
```

[41 Marks]

```
try
{
if("ss@123".equals(name))
                System.out.println("Valid name");
else
throw new name();
}catch(name e2)
{
}
try
{
if(12==(password))
System.out.println("Valid password");
else
throw new password();
}catch(password e3)
{
}
}
                }
```

Slip-17

Q 1) Accept the names of two files and copy the contents of the first to the second. First file having
Book name and Author name in file. [10 Marks]

```
        import java.io.*;
import java.util.*;
class copy{
public static void main(String arg[]) throws Exception {
Scanner sc = new Scanner(System.in);
System.out.print("source file name :\n");
String file1 = sc.next();
System.out.print("destination file name :\n");
String file2 = sc.next();
FileReader fin = new FileReader(file1);
```

[42 Marks]

```
FileWriter fout = new FileWriter(file2, true);
int c;
while ((c = fin.read()) != -1) {
fout.write(c);
}
System.out.println("Copy finish...");

fin.close();
fout.close();
}
}
```

Q 2) Write a program to create parent class College(cno, cname, caddr) and derived class Department(dno, dname) from College. Write a necessary methods to display College details.

```
// College class
class College {
    int cno;
    String cname;
    String caddr;

    // Constructor for College class
    public College(int cno, String cname, String caddr) {
        this.cno = cno;
        this.cname = cname;
        this.caddr = caddr;
    }

    // Method to display college details
    public void displayCollegeDetails() {
        System.out.println("College Number: " + cno);
        System.out.println("College Name: " + cname);
        System.out.println("College Address: " + caddr);
    }
}

// Department class extending College
class Department extends College {
    int dno;
    String dname;
```

[43 Marks]

```
    // Constructor for Department class
    public Department(int cno, String cname, String caddr, int dno, String dname) {
        super(cno, cname, caddr);
        this.dno = dno;
        this.dname = dname;
    }

    // Method to display department details
    public void displayDepartmentDetails() {
        super.displayCollegeDetails();
        System.out.println("Department Number: " + dno);
        System.out.println("Department Name: " + dname);
    }
}

// Main class for testing
public class tybsc {
    public static void main(String[] args) {
        // Creating an instance of the Department class
        Department department = new Department(1, "Example College", "College Street",
101, "Computer Science");

        // Displaying department details
        department.displayDepartmentDetails();
    }
}
```

Slip-18

Q 1) Write a program to accept the two dimensional array from user and display sum of its diagonal Elements

```
import java.util.*;
```

[44 Marks]

```
public class slippract
{
public static void main(String[] args)
{
int a[][]=new int[20][20];
int i,j,n,sum=0;
Scanner sc=new Scanner(System.in);
System.out.println("Enter rws and cols of a mtrix =");
n=sc.nextInt();
System.out.println("Enter the elemnets of a mtrix- ");
for(i=0;i<n;i++)
{
for(j=0;j<n;j++)
{
a[i][j]=sc.nextInt();
}
}
System.out.println("Sum of diagonal elements= ");
for(i=0;i<n;i++)
{
for(j=0;j<n;j++)
{
if(i==j)
{
sum=sum+a[i][j];
}
}
}
System.out.println(sum);
}
 }
```

Q 2) Design a Super class Customer (name, phone-number). Derive a class Depositor(accno , balance) from Customer. Again, derive a class Borrower (loan-no, loan-amt) from Depositor. Write necessary member functions to read and display the details of ‘n’customers

```
import java.util.*;
```

[45 Marks]

```
class customer
{
int cno;
String cname;
Scanner sc=new Scanner(System.in);
customer()
{
System.out.println("Enter customer no name= ");
cno=sc.nextInt();
cname=sc.next();
}
}
class depositor extends customer
{
int acno,bal;
depositor()
{
System.out.println("Enter depositor accno and balance= ");
acno=sc.nextInt();
bal=sc.nextInt();
}
}
class borrower extends depositor
{
int lno, lamt;
void accept()
{
System.out.println("Enter borrower loan no and amt= ");
lno=sc.nextInt();
lamt=sc.nextInt();
}
void disp()
{
System.out.println("Customer no= "+cno);
System.out.println("Customer name= "+cname);
System.out.println("Depositor acno= "+acno);
System.out.println("Depositor bal= "+bal);
System.out.println("Borrower loan no= "+lno);
System.out.println("Borrower loan amount= "+lamt);
}
```

[46 Marks]

```
}
public class slippract
{
public static void main(String[] args)
{
borrower b[]=new borrower[20];
int n,i;
Scanner sc=new Scanner(System.in);
System.out.println("Enter no of customer= ");
n=sc.nextInt();
for(i=0;i<n;i++)
{
b[i]=new borrower();
b[i].accept();
}
for(i=0;i<n;i++)
{
b[i].disp();
}
}
}
```

SLIP 19

Q 1) Create a text file and display the file information such as name,directory name,path,absolute path,lastmodified time,length of file using methods.

```
import java.io.File;
import java.io.IOException;
import java.text.SimpleDateFormat;
import java.util.Date;

public class FileInfoDisplay {

   public static void main(String[] args) {
      String fileName = "myTextFile.txt";
```

[47 Marks]

```
File file = new File(fileName);

try {
    // Create the new file
    if (file.createNewFile()) {
        System.out.println("File created successfully: " + file.getName());
    } else {
        System.out.println("File already exists: " + file.getName());
    }

    // Display file information
    System.out.println("\n--- File Information ---");
    System.out.println("File Name: " + file.getName());
    System.out.println("Directory Name: " + file.getParent()); // Returns null if in current
            directory
    System.out.println("Path: " + file.getPath());
    System.out.println("Absolute Path: " + file.getAbsolutePath());

    // Get last modified time and format it
    long lastModifiedMillis = file.lastModified();
    Date lastModifiedDate = new Date(lastModifiedMillis);
    SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");
    System.out.println("Last Modified Time: " + sdf.format(lastModifiedDate));

    System.out.println("Length of File (bytes): " + file.length());

} catch (IOException e) {
    System.err.println("An error occurred: " + e.getMessage());
} finally {
    // Optional: Delete the file when done (uncomment if needed)
    // if (file.exists()) {
    //     file.delete();
    //     System.out.println("File deleted: " + file.getName());
    // }
```

[48 Marks]

```
    }
  }
}
```

Q 2) Write a package for Operation, which has two classes, Addition and Maximum. Addition has two methods add () and subtract (), which are used to add two integers and subtract two, float valuesrespectively. Maximum has a method max () to display the maximum of two integers

```
import java.util.*;
class Addition
{
int a,b;
Scanner sc=new Scanner(System.in);
void add()
{
System.out.println("Enter value of a and b= ");
a=sc.nextInt();
b=sc.nextInt();
int c=a+b;
System.out.println("Addition= "+c);
}
void sub()
{
System.out.println("Enter value of a and b= ");
a=sc.nextInt();
b=sc.nextInt();
int c=a-b;
System.out.println("Substraction= "+c);
}
}
class maximum
{
int a,b;
Scanner sc=new Scanner(System.in);
void max()
{
System.out.println("Enter value of a and b= ");
a=sc.nextInt();
```

[49 Marks]

```
b=sc.nextInt();
if(a>b)
System.out.println("max no is- "+a);
else
System.out.println("max no is- "+b);
}
}
public class op1
{
public static void main(String args[])
{
Addition a=new Addition();
maximum m=new maximum();
a.add();
a.sub();
m.max();
}
}
```

.

SLIP 20

Q 1) Write program to define class Person with data member as Personname,Aadharno, Panno. Accept information for 5 objects and display appropriate information (use this keyword).

```
 import java.util.*;
class person
{
String pname;
int ano, panno;
person(String pname,int ano,int panno)
{
this.pname=pname;
this.ano=ano;
this.panno=panno;
}
void disp()
```

[50 Marks]

```
{
System.out.println("Person name= "+pname);
System.out.println("Person addhar no= "+ano);
System.out.println("Person panno= "+panno);
}
}
public class slippract
{
public static void main(String[] args)
{
Scanner sc=new Scanner(System.in);
String pname;
int ano, panno,i;
person p[]=new person[5];
for(i=0;i<5;i++)
{
System.out.println("Enter person name addhar no and pan no=");
pname=sc.next();
ano=sc.nextInt();
panno=sc.nextInt();
p[i]=new person(pname,ano,panno);
}
for(i=0;i<5;i++)
{
p[i].disp();
}
}
}
```

Q 2) Write a program to display the following menus and sub-menus.

```
package PS;
import java.util.*;
import javax.swing.*;
import java.awt.*;
public class Que37 extends Frame
{
    public static void main(String[] args)
    {
   JFrame j=new JFrame("Menubar,Menu,MenuItems");
   j.setVisible(true);
   j.setSize(600,400);
```

[51 Marks]

```
        JMenuBar mb=new JMenuBar();
        mb.setVisible(true);
        j.setJMenuBar(mb);

        JMenu m=new JMenu("File");
        mb.add(m);
        m.setVisible(true);

        JMenuItem m1=new JMenuItem("New");
        JMenuItem m2=new JMenuItem("Open");
        JMenuItem m3=new JMenuItem("Save");

           m.add(m1);
           m.add(m2);
           m.add(m3);

           JMenu m4=new JMenu("Save as");
           JMenuItem m5=new JMenuItem(".ppt");
           JMenuItem m6=new JMenuItem(".doc");
           JMenuItem m7=new JMenuItem(".pdf");

           m.add(m4);
           m4.add(m5);
           m4.add(m6);
           m4.add(m7);
         }

    }
```

SLIP 21

Q 1) Write a program to create a class Customer(custno,custname,contactnumber,custaddr). Write a method to accept and display the details

```
import java.util.*;
```

[52 Marks]

```
class customer
{
int cno,conum;
String cname,cadd;
Scanner sc=new Scanner(System.in);
void accept()
{
System.out.println("Enter customer no name and address= ");
cno=sc.nextInt();
cname=sc.next();
cadd=sc.next();
}
void search(int scono)
{
if(cno==scono)
{
System.out.println("The customer name with given contact number= "+cname);
}
}
void disp()
{
System.out.println("Customer no= "+cno);
System.out.println("Customer name= "+cname);
System.out.println("Customer address= "+cadd);
}
}
public class slippract
{
public static void main(String[] args)
{
int n,i,scono;
Scanner sc=new Scanner(System.in);
System.out.println("Enter no of customer= ");
n=sc.nextInt();
customer c[]=new customer[20];
for(i=0;i<n;i++)
{
c[i]=new customer();
c[i].accept();
}
```

[53 Marks]

```
System.out.println("Enter a contact no to search= ");
scono=sc.nextInt();
for(i=0;i<n;i++)
{
c[i].search(scono);
}
for(i=0;i<n;i++)
{
c[i].disp();
}
}
}
```

Q 2) Define class EmailId with members ,username and password. Define default and parameterized constructors. Accept values from the command line Throw user defined exceptions – "InvalidUsernameException" or "InvalidPasswordException" if the username and password are Invalid

```
import java.util.*;
class name extends Exception
{
public name()
{
System.out.println("InvalidUsernameException");
}
}
class password extends Exception
{
public password()
{
System.out.println("InvalidPasswordException");
}
}
```

[54 Marks]

```
class Emailid
{
String name;
int password;
Emailid()
{
name="";
password=0;
}
Emailid(String name,int password)
{
this.name=name;
this.password=password;
}
}
public class slippract
{
public static void main(String[] args)
{
String name;
int password;
name=args[0];
password=Integer.parseInt(args[1]);
try
{
if("ss@123".equals(name))
System.out.println("Valid name");
else
throw new name();
}catch(name e2)
{
}
try
{
if(12==(password))
System.out.println("Valid password");
else
throw new password();
}catch(password e3)
{
```

[55 Marks]

```
}
}
}
```

Slip-22

Q 1) Write a package for String operation which has two classes Con and Comp. Con class has to concatenates two strings and comp class compares two strings. Also display proper message on execution

```
package makone;
public class compare
{
String s1,s2;
compare(String s1,String s2)
{
this.s1=s1;
this.s2=s2;
}
void compare()
{
if(s1.equals(s2))
System.out.println("String are equal");
else
System.out.println("String are not equal");
}
}
/class concate/
package makone;
public class concate
{
String s1,s2;
concate(String s1,String s2)
{
this.s1=s1;
this.s2=s2;
```

[56 Marks]

```
}
void concate()
{
String s3;
s3=s1.concat(s2);
System.out.println("Concatnate string- "+s3);
}
}
/String class/
package makone;
import java.util.*;
public class string {
public static void main(String[] args)
{
Scanner sc=new Scanner(System.in);
String s1,s2;
System.out.println("Enter string 1 and string 2= ");
s1=sc.next();
s2=sc.next();
concate c1=new concate(s1,s2);
c1.concate();
compare c2=new compare(s1,s2);
c2.compare();
}
}
```

Q 2) Create an employee class(id,name,deptname,salary). Define a default and parameterized constructor. Use 'this' keyword to initialize instance variables. Keep a count of objects created. Create objects using parameterized constructor and display the object count after each object is created.(Use static member and method). Also display the contents of each object.

```
package makone;
import java.util.*;
class employee
{
int id,sal;
String dname,name;
static int cnt=0;
employee()
{
```

[57 Marks]

```
id=0;
name="";
dname="";
sal=0;
}
employee(int id,String name,String dname,int sal)
{
this.id=id;
this.name=name;
this.dname=dname;
this.sal=sal;
cnt++;
}
static int count()
{
return cnt++;
}
void disp()
{
System.out.println("Emp id= "+id);
System.out.println("Emp name= "+name);
System.out.println("Emp dname= "+dname);
System.out.println("Emp sal= "+sal);
}
}
public class slippract
{
public static void main(String[] args)
{
Scanner sc=new Scanner(System.in);
int n,i;
int id,sal;
String dname,name;
employee e[]=new employee[20];
System.out.println("Enter no of emp= ");
n=sc.nextInt();
for(i=0;i<n;i++)
{
System.out.println("Enter emp id name dep sal= ");
id=sc.nextInt();
```

[58 Marks]

```
name=sc.next();
dname=sc.next();
sal=sc.nextInt();
e[i]=new employee(id,name,dname,sal);
}
for(i=0;i<n;i++)
{
e[i].disp();
}
System.out.println("No of emp count= "+employee.count());
}
        }
```

Slip-23

Q 1) Write a java program to accept 5 numbers using command line arguments sort and display them.

```
import java.util.*;
public class slippract
{
public static void main(String[] args)
{
int t;
int a[]=new int[5];
a[0]=Integer.parseInt(args[0]);
a[1]=Integer.parseInt(args[1]);
a[2]=Integer.parseInt(args[2]);
a[3]=Integer.parseInt(args[3]);
a[4]=Integer.parseInt(args[4]);
System.out.println("Numbers before sorting= ");
for(int i=0;i<5;i++)
{
System.out.println(a[i]);
}
for(int i=0;i<5;i++)
{
for(int j=0;j<i;j++)
{
```

```
if(a[i]<a[j])
{
t=a[i];
a[i]=a[j];
a[j]=t;
}
}
}
System.out.println("Numbers after sorting= ");
for(int i=0;i<5;i++)
{
System.out.println(a[i]);
}
}
}
```

]

Q 2) Define a class patient (patient_name, patient_age, patient_oxy_level,patient_HRCT_report).

Create an object of patient. Handle appropriate exception while patient oxygen level less than 95% and HRCT scan report greater than 10, then throw user defined Exception "Patient is Covid Positive(+) and Need to Hospitalized" otherwise display its information.

```
import java.util.*;
class covid extends Exception
{
public covid()
{
System.out.println("Patient is Covid Positive(+) and Need to Hospitalized");
}
}
class patient
{
String patient_name;
int patient_age, patient_oxy_level,patient_HRCT_report;
patient(String patient_name,int patient_age, int patient_oxy_level,int patient_HRCT_report)
{
```

[60 Marks]

```
this.patient_name=patient_name;
this.patient_age=patient_age;
this.patient_oxy_level=patient_oxy_level;
this.patient_HRCT_report=patient_HRCT_report;
}
}
public class slippract
{
public static void main(String[] args)
{
Scanner sc=new Scanner(System.in);
String patient_name;
int patient_age, patient_oxy_level,patient_HRCT_report;
System.out.println("Enter patient name,int patient_age, int patient_oxy_level,int
patient_HRCT_report=");
patient_name=sc.next();
patient_age=sc.nextInt();
patient_oxy_level=sc.nextInt();
patient_HRCT_report=sc.nextInt();
patient p=new patient(patient_name,patient_age, patient_oxy_level,patient_HRCT_report);
try
{
if(patient_oxy_level<95 || patient_HRCT_report>10)
```

[61 Marks]

```
throw new covid();
else
{
System.out.println("Patient name= "+patient_name);
System.out.println("patient_age= "+patient_age);
System.out.println("patient_oxy_level= "+patient_oxy_level);
System.out.println("patient_HRCT_report= "+patient_HRCT_report);
}
}catch(covid e2)
{
}
}
}
```

Slip-24

Q 1) Write a program to display the Employee(Empid, Empname, Empdesignation, Empsal) information using toString

```
import java.util.*;
class Employee
{
private int empid;
private String empname;
private String empdesignation;
private double empsal;
public Employee(int empid, String empname, String empdesignation, double empsal) {
this.empid = empid;
this.empname = empname;
this.empdesignation = empdesignation;
this.empsal = empsal;
}
public String toString() {
return "Employee ID: " + empid + "\n" +
"Employee Name: " + empname + "\n" +
"Employee Designation: " + empdesignation + "\n" +
"Employee Salary: " + empsal;
}
}
public class slippract
{
public static void main(String[] args)
{
Scanner sc=new Scanner(System.in);
```

[62 Marks]

```
int empid;
String empname;
String empdes;
double empsal;
System.out.println("Enter emp id name des sal= ");
empid=sc.nextInt();
empname=sc.next();
empdes=sc.next();
empsal=sc.nextDouble();
Employee emp1 = new Employee(101, "John Doe", "Software Developer", 75000.50);
System.out.println(emp1.toString());
}
}
```

Q 2) Define Student class(roll_no, name, percentage) to create n objects of the Student class.

Accept details from the user for each object. Define a static method “sortStudent” which sorts the array on the basis of percentage

```
import java.util.*;
class student
{
int sid,per;
String name;
Scanner sc=new Scanner(System.in);
void accept()
{
System.out.println("Enter stud sid name per= ");
sid=sc.nextInt();
name=sc.next();
per=sc.nextInt();
}
void disp()
{
System.out.println("Student id= "+sid);
System.out.println("Student name= "+name);
System.out.println("Student per= "+per);
}
static void sort(student s[],int n)
{
int i,j;
for(i=1;i<n;i++)
{
for(j=0;j<n-i;j++)
{
```

[63 Marks]

```
if(s[j].per>s[j+1].per)
{
student t=s[j];
s[j]=s[j+1];
s[j+1]=t;
}
}
}
}
}
public class slippract
{
public static void main(String[] args)
{
Scanner sc=new Scanner(System.in);
student s[]=new student[20];
int n,i;
System.out.println("Enter no student= ");
n=sc.nextInt();
for(i=0;i<n;i++)
{
s[i]=new student();
s[i].accept();
}
student.sort(s, n);
for(i=0;i<n;i++)
{
s[i].disp();
}
} }
```

Slip-25

Q 1) Write a Program to print all Prime numbers in an array of 'n' elements. (use command line arguments). [10 Marks]

```
 import java.util.*;
public class slippract
{
public static void main(String[] args)
{
int n[]=new int[args.length];
for(int i=0;i<args.length;i++)
```

[64 Marks]

```
{
n[i]=Integer.parseInt(args[i]);
}
for(int i=0;i<args.length;i++)
{
int num=n[i];
if(num<2)
continue;
int flag=1;
for(int j=2;j<num;j++)
{
if(num%j==0)
{
flag=0;
break;
}
}
if(flag==1)
System.out.println(num);
}
}
```

Q 2) Define a class MyNumber having one private int data member. Write a default constructor to initialize it to 0 and another constructor to initialize it to a value (Use this). Write methods isNegative, isPositive, isZero, isOdd, isEven. Create an object in main. Use command lineargumentsto pass a value to the object

```
import java.util.*;
class Mynumber
{
private int a;
Mynumber()
{
a=0;
}
Mynumber(int a)
{
this.a=a;
}
void isNegative()
{
if(a<0)
System.out.println("Number is negative");
```

[65 Marks]

```
}
void isPositive()
{
if(a>0)
System.out.println("Number is positive");
}
void isZero()
{
if(a==0)
System.out.println("Number is zero");
}
void isEven()
{
if(a%2==0)
System.out.println("Number is even");
}
void isOdd()
{
if(a%2!=0)
System.out.println("Number is odd");
}
}
public class slippract
{
public static void main(String[] args)
{
int no=Integer.parseInt(args[0]);
Mynumber ob=new Mynumber(no);
ob.isNegative();
ob.isPositive();
ob.isEven();
ob.isOdd();
ob.isZero();
}
          }
```

SLIP 26

Q 1) Create a class Sphere, to calculate the volume and surface area of sphere. (Hint : Surface area=4*3.14(r*r), Volume=(4/3)3.14(r*r*r

```
import java.util.*;
public class slippract
{
```

[66 Marks]

```
public static void main(String[] args)
{
float r;
Scanner sc=new Scanner(System.in);
System.out.println("Enter ardius of sphere= ");
r=sc.nextFloat();
float a=4*3.14f*(r*r);
float v=(4/3)3.14f(r*r*r);
System.out.println("Area of sphere= "+a);
System.out.println("Volume of sphere= "+v);
}
}
```

Q 2) Write a java program to display the system date and time in various formats show below: Current date is : 31/08/2021

Current date is : 08-31-2021

Current date is : Tuesday August 31 2021

Current date and time is : Fri August 31 15:25:59 IST 2021

Current date and time is : 31/08/21 15:25:59 PM +0530

Current time is : 15:25:59

Current week of year is : 35

Current week of month : 5

Current day of the year is : 243

Note: Use java.util.Date and java.text.SimpleDateFormat class [15 Marks]

```
import java.util.*;
import java.text.SimpleDateFormat;
public class slippract
{
public static void main(String[] args)
{
Date d=new Date();
SimpleDateFormat sd=new SimpleDateFormat("dd/mm/yyyy");
String date=sd.format(d);
System.out.println("Current date is:"+date);
sd=new SimpleDateFormat("MM-dd-yyyy");
date=sd.format(d);
System.out.println("Current date is:"+date);
sd=new SimpleDateFormat("EEE MMM-dd-yyyy");
date=sd.format(d);
```

[67 Marks]

```
System.out.println("Current date is:"+date);
sd=new SimpleDateFormat("EEE MMM-dd-yyyy HH:mm:ss z");
date=sd.format(d);
System.out.println("Current date and time is:"+date);
sd=new SimpleDateFormat("dd/MM/yyyy HH:mm:ss z");
date=sd.format(d);
System.out.println("Current date and time is:"+date);
sd=new SimpleDateFormat("HH:mm:ss");
date=sd.format(d);
System.out.println("Current time is:"+date);
sd=new SimpleDateFormat("w");
date=sd.format(d);
System.out.println("Current week of year is:"+date);
sd=new SimpleDateFormat("W");
date=sd.format(d);
System.out.println("Current week of month is:"+date);
sd=new SimpleDateFormat("d");
date=sd.format(d);
System.out.println("Current day of the year is:"+date);
}
}
```

SLIP 27

Q 1) Write a program to find the cube of given number using functional interface.

```
import java.util.*;
interface cube
{
public void cube(int n);
}
public class slippract
{
public static void main(String[] args)
{
cube c=x->System.out.println("Cube ="+x*x*x);
c.cube(2);
}
}
```

Q 2) Write a program to create a class Product (product_id, product_name, product_cost, product_quantity) default and parameterized constructor. Create objectsof class product and display the contents of each object and Also display the object count. [15 Marks]

[68 Marks]

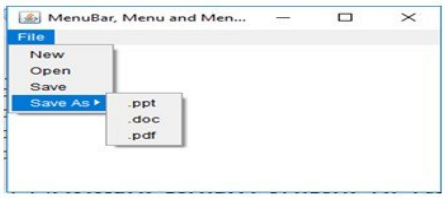
```
import java.util.*;
class product
{
int pid,pcost, pqu;
static int cnt=0;
String pname;
product()
{
pid=0;
pname="";
pcost=0;
pqu=0;
}
product(int pid,String pname,int pcost,int pqu)
{
this.pid=pid;
this.pname=pname;
this.pcost=pcost;
this.pqu=pqu;
cnt++;
}
void disp()
{
System.out.println("Product id= "+pid);
System.out.println("Product name= "+pname);
System.out.println("Product cost= "+pcost);
System.out.println("Product quantity= "+pqu);
}
}
public class slippract
{
public static void main(String[] args)
{
Scanner sc=new Scanner(System.in);
int pid,pcost, pqu;
String pname;
product p[]=new product[20];
int n,i;
System.out.println("Enter no of product= ");
n=sc.nextInt();
for(i=0;i<n;i++)
{
System.out.println("Enter product id name cost and quantity= ");
pid=sc.nextInt();
pname=sc.next();
```

[69 Marks]

```
pcost=sc.nextInt();
pqu=sc.nextInt();
p[i]=new product(pid,pname,pcost,pqu);
}
for(i=0;i<n;i++)
{
p[i].disp();
}
System.out.println("No of object count= "+product.cnt);
}
}
```

Slip-28

Q 1) Write a program to display the following menus and sub-menus. [10 Marks]




```
package PS;
import java.util.*;
import javax.swing.*;
import java.awt.*;
public class Que37 extends Frame
{
    public static void main(String[] args)
    {
    JFrame j=new JFrame("Menubar,Menu,MenuItems");
    j.setVisible(true);
    j.setSize(600,400);

    JMenuBar mb=new JMenuBar();
```

[70 Marks]

```
        mb.setVisible(true);
        j.setJMenuBar(mb);

        JMenu m=new JMenu("File");
        mb.add(m);
        m.setVisible(true);

        JMenuItem m1=new JMenuItem("New");
        JMenuItem m2=new JMenuItem("Open");
        JMenuItem m3=new JMenuItem("Save");

          m.add(m1);
          m.add(m2);
          m.add(m3);

          JMenu m4=new JMenu("Save as");
          JMenuItem m5=new JMenuItem(".ppt");
          JMenuItem m6=new JMenuItem(".doc");
          JMenuItem m7=new JMenuItem(".pdf");

          m.add(m4);
          m4.add(m5);
          m4.add(m6);
          m4.add(m7);
        }

    }
```

Q 2) Write a java program to create an interface shape with the getArea() method.Create three classes Rectangle,Circle and Triangle that implements the shape interface.Implement the getArea() method for each of the three classes. [15 Marks]

```
import java.lang.Math; // Required for Math.PI and Math.pow


// Define the Shape interface

interface Shape {
```

[71 Marks]

```
    double getArea(); // Method to calculate and return the area
}

// Implement the Shape interface for Rectangle
class Rectangle implements Shape {
    private double length;
    private double width;

    public Rectangle(double length, double width) {
        this.length = length;
        this.width = width;
    }

    @Override
    public double getArea() {
        return length * width;
    }
}

// Implement the Shape interface for Circle
class Circle implements Shape {
    private double radius;
```

[72 Marks]

```
    public Circle(double radius) {
        this.radius = radius;
    }

    @Override
    public double getArea() {
        return Math.PI * Math.pow(radius, 2); // Area of a circle: π * r^2
    }
}

// Implement the Shape interface for Triangle
class Triangle implements Shape {
    private double base;
    private double height;

    public Triangle(double base, double height) {
        this.base = base;
        this.height = height;
    }

    @Override
    public double getArea() {
        return 0.5 * base * height; // Area of a triangle: 0.5 * base * height
```

[73 Marks]

```
    }
}

// Main class to demonstrate the usage
public class ShapeDemo {
    public static void main(String[] args) {
        // Create instances of the shape classes
        Rectangle rect = new Rectangle(5, 10);
        Circle circ = new Circle(7);
        Triangle tri = new Triangle(6, 8);

        // Calculate and print the area of each shape
        System.out.println("Area of Rectangle: " + rect.getArea());
        System.out.println("Area of Circle: " + circ.getArea());
        System.out.println("Area of Triangle: " + tri.getArea());

        // Demonstrate polymorphism using the Shape interface
        Shape s1 = new Rectangle(4, 6);
        Shape s2 = new Circle(3);
        Shape s3 = new Triangle(5, 7);

        System.out.println("\nArea of Shape 1 (Rectangle): " + s1.getArea());
        System.out.println("Area of Shape 2 (Circle): " + s2.getArea());
```

[74 Marks]

```
        System.out.println("Area of Shape 3 (Triangle): " + s3.getArea());
    }
}
```

Slip-29

Q 1) Write a Program to print all even numbers in an array of 'n' elements. (use command line arguments). [10 Marks]

```
 import java.util.*;
public class slippract
{
public static void main(String[] args)
{
int n[]=new int[args.length];
for(int i=0;i<args.length;i++)
{
n[i]=Integer.parseInt(args[i]);
}
for(int i=0;i<args.length;i++)
{
int num=n[i];
if(num<2)
continue;
int flag=1;
for(int j=2;j<num;j++)
{
if(num%j==0)
{
flag=0;
break;
}
}
if(flag==1)
System.out.println(num);
}
}
```

[75 Marks]

}

Q 2) Design a Super class Customer (name, phone-number). Derive a class Depositor(accno , balance) from Customer. Again, derive a class Borrower (loan-no, loan-amt) from Depositor.

Write necessary member functions to read and display the details of 'n'customers. [15 Marks]

```
import java.util.*;
class customer
{
int cno;
String cname;
Scanner sc=new Scanner(System.in);
customer()
{
System.out.println("Enter customer no name= ");
cno=sc.nextInt();
cname=sc.next();
}
}
class depositor extends customer
{
int acno,bal;
depositor()
{
System.out.println("Enter depositor accno and balance= ");
acno=sc.nextInt();
bal=sc.nextInt();
}
}
class borrower extends depositor
{
int lno, lamt;
void accept()
{
System.out.println("Enter borrower loan no and amt= ");
lno=sc.nextInt();
lamt=sc.nextInt();
}
void disp()
{
```

[76 Marks]

```
System.out.println("Customer no= "+cno);
System.out.println("Customer name= "+cname);
System.out.println("Depositor acno= "+acno);
System.out.println("Depositor bal= "+bal);
System.out.println("Borrower loan no= "+lno);
System.out.println("Borrower loan amount= "+lamt);
}
}
public class slippract
{
public static void main(String[] args)
{
borrower b[]=new borrower[20];
int n,i;
Scanner sc=new Scanner(System.in);
System.out.println("Enter no of customer= ");
n=sc.nextInt();
for(i=0;i<n;i++)
{
b[i]=new borrower();
b[i].accept();
}
for(i=0;i<n;i++)
{
b[i].disp();
}
}
}
```

Slip-30

Q 1) Write a program to accept the two dimensional array from user and display sum of its diagonal elements. [10 Marks]

```
import java.util.*;
public class slippract
{
public static void main(String[] args)
{
int a[][]=new int[20][20];
```

[77 Marks]

```
int i,j,n,sum=0;
Scanner sc=new Scanner(System.in);
System.out.println("Enter rws and cols of a mtrix =");
n=sc.nextInt();
System.out.println("Enter the elemnets of a mtrix- ");
for(i=0;i<n;i++)
{
for(j=0;j<n;j++)
{
a[i][j]=sc.nextInt();
}
}
System.out.println("Sum of diagonal elements= ");
for(i=0;i<n;i++)
{
for(j=0;j<n;j++)
{
if(i==j)
{
sum=sum+a[i][j];
}
}
}
System.out.println(sum);
}
        }
```

Q 2) Write a java program to create an interface shape with the getArea() method.Create three classes Rectangle,Circle and Triangle that implements the shape interface.Implement the
getArea() method for each of the three classes

```
import java.lang.Math; // Required for Math.PI and Math.pow

// Define the Shape interface
interface Shape {
   double getArea(); // Method to calculate and return the area
}

// Implement the Shape interface for Rectangle
class Rectangle implements Shape {
   private double length;
```

[78 Marks]

```
    private double width;

    public Rectangle(double length, double width) {
        this.length = length;
        this.width = width;
    }

    @Override
    public double getArea() {
        return length * width;
    }
}

// Implement the Shape interface for Circle
class Circle implements Shape {
    private double radius;

    public Circle(double radius) {
        this.radius = radius;
    }

    @Override
    public double getArea() {
        return Math.PI * Math.pow(radius, 2); // Area of a circle: π * r^2
    }
}

// Implement the Shape interface for Triangle
class Triangle implements Shape {
    private double base;
    private double height;

    public Triangle(double base, double height) {
        this.base = base;
        this.height = height;
    }

    @Override
```

[79 Marks]

```
    public double getArea() {
        return 0.5 * base * height; // Area of a triangle: 0.5 * base * height
    }
}

// Main class to demonstrate the usage
public class ShapeDemo {
    public static void main(String[] args) {
        // Create instances of the shape classes
        Rectangle rect = new Rectangle(5, 10);
        Circle circ = new Circle(7);
        Triangle tri = new Triangle(6, 8);

        // Calculate and print the area of each shape
        System.out.println("Area of Rectangle: " + rect.getArea());
        System.out.println("Area of Circle: " + circ.getArea());
        System.out.println("Area of Triangle: " + tri.getArea());

        // Demonstrate polymorphism using the Shape interface
        Shape s1 = new Rectangle(4, 6);
        Shape s2 = new Circle(3);
        Shape s3 = new Triangle(5, 7);

        System.out.println("\nArea of Shape 1 (Rectangle): " + s1.getArea());
        System.out.println("Area of Shape 2 (Circle): " + s2.getArea());
        System.out.println("Area of Shape 3 (Triangle): " + s3.getArea());
    }
}
```

[80 Marks]