EXPERIMENT-01

Q1. Akshay scored 45, 67, 98 and 70 marks in 4 subjects. Write a java code to calculate his Average marks and percentage using Command line Arguments.

Source Code:

```
import java.util.Scanner;

class Test

{

public static void main(String args[])

{    float eng, phy, chem, math;

     double total, average, percentage;

    Scanner op=new Scanner(System.in);


  /* Input marks of all 4 subjects */

    System.out.println("Enter marks of five subjects:");

    System.out.print("Enter marks of English subjects:");

    eng=op.nextFloat();

    System.out.print("Enter marks of Physics subjects:");

    phy=op.nextFloat();

    System.out.print("Enter marks of Chemistry subjects:");

    chem=op.nextFloat();

    System.out.print("Enter marks of Maths subjects:");

    math=op.nextFloat();


 /* Calculate total, average and percentage */

    total = eng + phy + chem + math;

    average = (total / 4.0);

    percentage = (total / 400.0) * 100 ;
```

```
/* Print all results */
   System.out.println("Total marks ="+total);
   System.out.println("Average marks = "+average);
   System.out.println("Percentage = "+percentage);
}
}
```

Output :

```
Enter marks of five subjects:
Enter marks of English subjects:65
Enter marks of Physics subjects:70
Enter marks of Chemistry subjects:88
Enter marks of Maths subjects:95
Total marks =318.0
Average marks = 79.5
Percentage = 79.5
```

EXPERIMENT-02

Q2 . Akshay wants to calculate the area of Square, Rectangle and Circle. Create three different methods having same name calculateArea(). Take input from command line.

Source Code:

```
import java.util.Scanner;

public class Main

{

    public static void main(String[] args)

   {

      Area ob = new Area();

   Scanner op=new Scanner(System.in);

   System.out.println("Enter the length of Square");

            ob.calculateArea(op.nextFloat());


  System.out.println("Enter the length and breadth of Rectangle");

            ob.calculateArea (op.nextFloat(),op.nextFloat());


  System.out.println("Enter the radius of Circle");

            ob.calculateArea(op.nextDouble());

   }

}

class Area

{

   void calculateArea(float x)

   {

      System.out.println("The area of the square is "+ x*x+" sq units");
```

```
    }
    void calculateArea(float x, float y)
    {
        System.out.println("The area of the rectangle is "+x*y+" sq units");
    }
    void calculateArea(double x)
    {
        double z = 3.14 * x * x;
        System.out.println("The area of the circle is "+z+" sq units");
    }
}
```

Output :

```
Enter the length of Square
5
The area of the square is 25.0 sq units
Enter the length and breadth of Rectangle
6  8
The area of the rectangle is 48.0 sq units
Enter the radius of Circle
7
The area of the circle is 153.86 sq units
```

EXPERIMENT-03

Q3. Write a JAVA program to compute the sum of even digits in a 5 digit number.

Source Code:

```
import java.util.Scanner;


public class SumOfEvenDigits {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        System.out.print("Enter a 5-digit number: ");
        int number = input.nextInt();


        int sum = 0;
        int digit;
        while (number > 0) {
            digit = number % 10;
            if (digit % 2 == 0) {
                sum += digit;
            }
            number /= 10;
        }


        System.out.println("The sum of even digits is: " + sum);
    }
}
```

Output :

```
Enter a 5-digit number:
45783
The sum of even digits is: 12
```

```
Enter a 5-digit number:
22736
The sum of even digits is: 10
```

```
Enter a 5-digit number:
35678
The sum of even digits is: 14
```

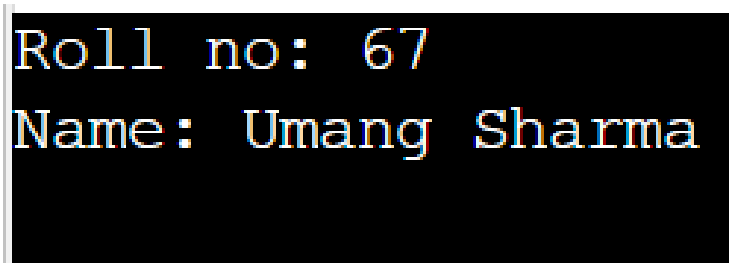
EXPERIMENT-04

Q4 . Create a Class Student having data members roll no and name and methods are getData() and display().

Source Code:

```
class Student {
    int rollno;
    String name;
 public void getData(int r, String n) {
      this.rollno = r;
        this.name = n;
 }
  public void display() {
        System.out.println("Roll no: " + rollno);
        System.out.println("Name: " + name);
    }
}


public class Main {
    public static void main(String[] args) {
     Student s = new Student();


        s.getData( 67, "Prashant");
        s.display();
    }
}
```

Output :



EXPERIMENT-05

Q5 . Create a Class Student having data members roll no and name and parametrized constructor.

Source Code:

```
class Student {
   int rollNo;
   String name;



   public Student(int rollNo, String name) {
      this.rollNo = rollNo;
      this.name = name;
   }



   public int getRollNo() {
      return rollNo;
   }


   public String getName() {
      return name;
   }
}


public class Main {
    public static void main(String[] args) {
```

```
        Student st= new Student(4,"Arjun");


      String n=st.getName();
        int r=st.getRollNo();


        System.out.println("Name: " + n );
        System.out.println("Roll no.: " + r );


    }
}
```

Output :

```
Name: Arjun
Roll no.: 4
```

EXPERIMENT-06

Q6 . Create a Class “Person” and declare the following variable name (string) and save it in a file called “Person.java”. Create a Class called Employee that will inherit the Person Class , the other data members of Employee Class are AnnualSalary (double), YearOfJoininging (int) and MobileNumber (long). Your Class Should have the necessary constructors and the getter and the setter methods. Do create Class Test having main method to fully check your class definition.

Source Code:

```
class Person {
  protected String name;

  public Person(String name) {
    this.name = name;
  }

  public String getName() {
    return name;
  }

  public void setName(String name) {
    this.name = name;
  }
}

 class Employee extends Person {
  private double annualSalary;
  private int yearOfJoining;
```

```
  private long mobileNumber;

  public Employee(String name, double annualSalary, int yearOfJoining, long
mobileNumber) {
    super(name);
    this.annualSalary = annualSalary;
    this.yearOfJoining = yearOfJoining;
    this.mobileNumber = mobileNumber;
  }

  public double getAnnualSalary() {
    return annualSalary;
  }

  public void setAnnualSalary(double annualSalary) {
    this.annualSalary = annualSalary;
  }

  public int getYearOfJoining() {
    return yearOfJoining;
  }

  public void setYearOfJoining(int yearOfJoining) {
    this.yearOfJoining = yearOfJoining;
  }

  public long getMobileNumber() {
```

```
    return mobileNumber;
  }

  public void setMobileNumber(long mobileNumber) {
    this.mobileNumber = mobileNumber;
  }
}

public class Main{
  public static void main(String[] args) {
    Employee e = new Employee("Arjun Pandit", 50000.00, 2020, 1234567890);
    System.out.println("Employee Name: " + e.getName());
    System.out.println("Annual Salary: " + e.getAnnualSalary());
    System.out.println("Year of Joining: " + e.getYearOfJoining());
    System.out.println("Mobile Number: " + e.getMobileNumber());
  }
}
```

Output :

```
Employee Name: Arjun Pandit
Annual Salary: 50000.0
Year of Joining: 2020
Mobile Number: 1234567890
```

EXPERIMENT-07

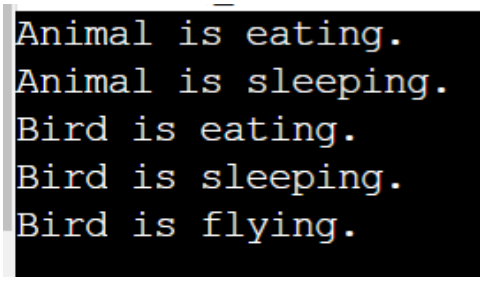
Q7 . Create a class named as “Animal” which has methods eat() and sleep(). Create a child class of animal named as “Bird” and override the parent class methods such that define a new method named as fly() in the same class. Create an instance of Animal Class and invoke eat() and sleep(). Create an instance of Bird Class and invoke eat(), sleep() and fly().

Source Code:

```
class Animal {
    public void eat() {
        System.out.println("Animal is eating.");
    }
    public void sleep() {
        System.out.println("Animal is sleeping.");
    }
}


 class Bird extends Animal {
    //Override
    public void eat() {
        System.out.println("Bird is eating.");
    }
    //Override
    public void sleep() {
        System.out.println("Bird is sleeping.");
    }
    public void fly() {
        System.out.println("Bird is flying.");
    }
```

```
}

public class Main{

 public static void main (String[] args) {


Animal animal = new Animal();
animal.eat();
animal.sleep();

Bird bird = new Bird();
bird.eat();
bird.sleep();
bird.fly();
  }
}
```

Output :



EXPERIMENT-08

Q8 . Write a JAVA program to calculate the cyclic sum of 5 digits number.

Source Code:

```
public class Main
{
        public static void main(String[] args)
{
System.out.println("Number=12345" + " "+ "Cyclic Sum=" + findCyclicSum(12345));
  }
  public static int findCyclicSum(int input1)
        {
                int sum=0, n, digits=0, rem, cyclicsum=0, i=0;
                n=input1;
                while(n!=0)
                {
                        n=n/10;
                        digits++;
                }
                int[] arr=new int[digits];
                n=input1;
                while(n!=0)
                {
                        rem=n%10;
                        arr[i]=rem;
                        i++;
                        n=n/10;
```

```
            }
            for(i=0;i<=digits-1;i++)
            {
                for(int j=0;j<=digits-1-i;j++)
                {
                    sum+=arr[j];
                }
                cyclicsum+=sum;
                sum=0;
            }
            return cyclicsum;
        }
}
```

Output :

```
Number=12345 Cyclic Sum=55
Number=25347 Cyclic Sum=72
Number=67895 Cyclic Sum=105
```

EXPERIMENT-09

Q9 . Create PIN using three given input numbers. Secure Assets Private Ltd", a small company that deals with digital lockers which can be locked and unlocked using PINs (password). You have been asked to work on the module that is expected to generate password using three input numbers.

Assumption: The three given input numbers will always consist of three digit i.e. each of them will be in range >=100 and <=999.

100<=input1<=999

100=Input2<-999

Look=lnput3<-999

Below are the rules for generating the PIN -

• The PIN should be made up of 4 digits

• The unit (ones) position of the PIN should be the least of the unit position of the three input numbers.

• The tens position of the PIN should be the least of the tens position of the three input numbers.

• The hundred position of the PIN should be the least of the hundreds position of the three input numbers.

• The thousand position of the PIN should be the maximum of all the digits in the three input numbers.

Source Code:

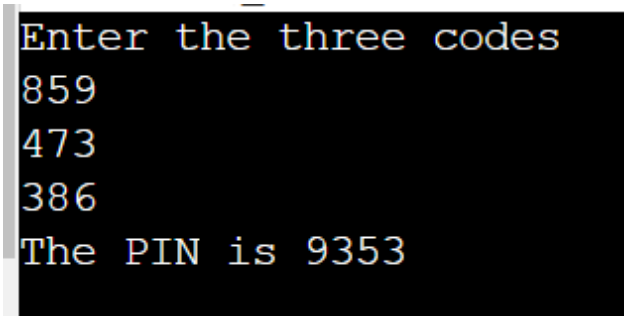
```
import java.util.Scanner;


class PIN{

public int createPin( int input1, int input2, int input3){

   int i=0,rem,u,t,h,k,j,a,pin;

   int[] A= new int[10];

   while(input1>0)

   {
```

```
    rem=input1%10;
    A[i]=rem;
    input1=input1/10;
    i++;
}

while(input2>0)
{
    rem=input2%10;
    A[i]=rem;
    input2=input2/10;
    i++;
}

while(input3>0)
{
    rem=input3%10;
    A[i]=rem;
    input3=input3/10;
    i++;
}

u=small(A[0],A[3],A[6]);
t=small(A[1],A[4],A[7]);
h=small(A[2],A[5],A[8]);

for(j=0;j<9;j++)
```

```
    {  for(k=j+1;k<9;k++)
      {  if(A[j]<A[k])
         {  a=A[j];
            A[j]=A[k];
            A[k]=a;}
      }
    }
    pin=1000*A[0]+100*h+10*t+u;
    return pin;
    }

public int small(int a, int b,int c){
    if(a<b && a<c)
    { return a;}
    else if(b<a && b<c)
    { return b;}
    else
    { return c;}

}
}

public class Main{


public static void main (String[] args) {
    int n1,n2,n3,pin;
```

```
    PIN ob=new PIN();
    Scanner s=new Scanner(System.in);
    System.out.println("Enter the three codes");
    n1=s.nextInt();
    n2=s.nextInt();
    n3=s.nextInt();
   pin= ob.createPin(n1,n2,n3);
   System.out.println("The PIN is" +" " + pin);


}
}
```

Output :



EXPERIMENT-10

Q 10. Create a class Author with the following information:

Member variables:

Name (String),e-mail(String) and gender (char)

Parametrized constructor : To initialize the variables.

Create a class Book with the following information:

Member variables:

Name(String),author (of the class Author you have just created , price(double) and qtyStock(int)

Getters and Setters method for all the member variables.

In the main method create a book object and print all details of the book.

Source Code:

```
class Author {
    private String name;
    private String email;
    private char gender;

    public Author(String name, String email, char gender) {
        this.name = name;
        this.email = email;
        this.gender = gender;
    }

    public String getName() {
        return name;
```

```
    }

    public String getEmail() {
        return email;
    }

    public char getGender() {
        return gender;
    }

    public void setName(String name) {
        this.name = name;
    }

    public void setEmail(String email) {
        this.email = email;
    }

    public void setGender(char gender) {
        this.gender = gender;
    }
}

 class Book {
    private String name;
    private Author author;
    private double price;
```

```
private int qtyStock;

public Book(String name, Author author, double price, int qtyStock) {
    this.name = name;
    this.author = author;
    this.price = price;
    this.qtyStock = qtyStock;
}

public String getName() {
    return name;
}

public Author getAuthor() {
    return author;
}

public double getPrice() {
    return price;
}

public int getQtyStock() {
    return qtyStock;
}

public void setName(String name) {
    this.name = name;
```

```
    }

    public void setAuthor(Author author) {
        this.author = author;
    }

    public void setPrice(double price) {
        this.price = price;
    }

    public void setQtyStock(int qtyStock) {
        this.qtyStock = qtyStock;
    }
}
public class Main {
    public static void main(String[] args) {
        Author author = new Author("Arundhati Roy", "arundhatiroy@email.com", 'F');
        Book book = new Book("The God of Small Things", author, 200, 50);

        System.out.println("Book Name: " + book.getName());
        System.out.println("Author Name: " + book.getAuthor().getName());
        System.out.println("Author Email: " + book.getAuthor().getEmail());
        System.out.println("Author Gender: " + book.getAuthor().getGender());
        System.out.println("Price: " + book.getPrice());
        System.out.println("Quantity in Stock: " + book.getQtyStock());
    }
}
```

Output :

```
Book Name: The God of Small Things
Author Name: Arundhati Roy
Author Email: arundhatiroy@email.com
Author Gender: F
Price: 200.0
Quantity in Stock: 50
```