1. Java Program(Command Line Argument):

Write a java program to take the number from the user from the command line and check whether the number is palindrome or not.

E.g. if the number is 12321 then its reverse i.e. 12321 equals to the actual number.

Source Code{*CommandLineArgument.java*}:

```
package JavaFileSolutions;

public class PalindromeUsingCmdArgs {
   public static void main(String[] args) {
        int num = Integer.parseInt(args[0]);

        int temp = num , palin =0 , rem;

        while(temp != 0){
            rem = temp%10;
            palin = (palin*10) + rem;
            temp = temp / 10;
        }

        if (palin == num)
            System.out.println(num+" is Palindrome");

        else
            System.out.println(num+" is not Palindrome");
   }
}
```

Output:

```
PS D:\Java\JavaFileSolutions> java .\PalindromeUsingCmdArgs.java 12321
12321 is Palindrome
PS D:\Java\JavaFileSolutions> java .\PalindromeUsingCmdArgs.java 123321
123321 is Palindrome
PS D:\Java\JavaFileSolutions> java .\PalindromeUsingCmdArgs.java 44322345
44322345 is not Palindrome
```

2. Java Simple Program(Only Main class):

Write a Java program to check if the given number is an Armstrong or not.
Definition :An Armstrong number is a positive integer that's equal to the sum of its digits, each raised to the power of the number of digits.
E.g. 153 is an Armstrong number because 1^3 + 5^3 + 3^3 = 153.

Source Code{JavaSimpleProgram.java}:

```
import java.util.Scanner;

public class Armstrong {
   public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter an integer number to check Armstrong : ");
        int armsNumber = sc.nextInt();
        int temp1 = armsNumber;
        int temp = armsNumber;
        int rem,count=0,arm=0;
        while(temp1!=0){
            temp1 = temp1/10;
            count++;
        }
        while (temp != 0){
            rem = temp%10;
            arm = (int) (arm+Math.pow(rem,count));
            temp = temp/10;
        }
        if(arm == armsNumber)
            System.out.println(armsNumber+" is Armstrong");
        else
            System.out.println(armsNumber+" is not Armstrong");
   }
}
```

Output:

```
Enter an integer number to check Armstrong : 153
153 is Armstrong
PS D:\Java\JavaFileSolutions> java .\Armstrong.java
Enter an integer number to check Armstrong : 1634
1634 is Armstrong
PS D:\Java\JavaFileSolutions> java .\Armstrong.java
Enter an integer number to check Armstrong : 1234
1234 is not Armstrong
PS D:\Java\JavaFileSolutions>
```

3. Java Program With Class And Object:

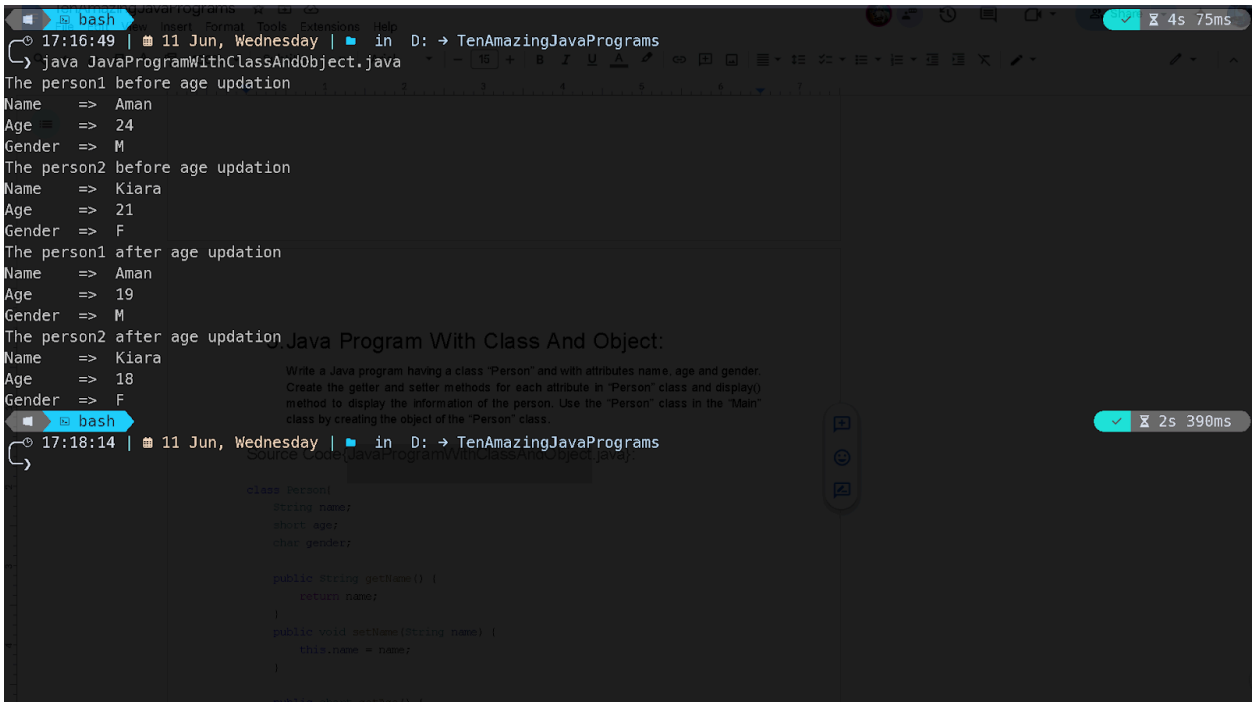
Write a Java program having a class “Person” and with attributes name, age and gender. Create the getter and setter methods for each attribute in “Person” class and display() method to display the information of the person. Use the “Person” class in the “Main” class by creating the object of the “Person” class.

Source Code{JavaProgramWithClassAndObject.java}:

```
class Person{
    String name;
    short age;
    char gender;

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

    public short getAge() {
        return age;
    }
    public void setAge(short age) {
        this.age = age;
    }

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

    public void display(){
        System.out.printf("""
                Name    =>  %s
                Age     =>  %d
                Gender  =>  %c
```

```
                """, name, age, gender);
    }
}
class JavaProgramWithClassAndObject{
    public static void main(String[] arg){
        Person person1 = new Person();

        person1.setName("Aman");
        person1.setAge((short)24);
        person1.setGender('M');
        System.out.println("The person1 before age updation");
        person1.display();

        Person person2 = new Person();

        person2.setName("Kiara");
        person2.setAge((short)21);
        person2.setGender('F');
        System.out.println("The person2 before age updation");
        person2.display();

        person1.setAge((short)19);
        System.out.println("The person1 after age updation");
        person1.display();

        person2.setAge((short)18);
        System.out.println("The person2 after age updation");
        person2.display();
    }
}
```

Output:


bash
4s 75ms
17:16:49 | 11 Jun, Wednesday | in D: → TenAmazingJavaPrograms
java JavaProgramWithClassAndObject.java
The person1 before age updation
Name => Aman
Age => 24
Gender => M
The person2 before age updation
Name => Kiara
Age => 21
Gender => F
The person1 after age updation
Name => Aman
Age => 19
Gender => M
The person2 after age updation
Name => Kiara
Age => 18
Gender => F
bash
2s 390ms
17:18:14 | 11 Jun, Wednesday | in D: → TenAmazingJavaPrograms
Java Program With Class And Object:
Write a Java program having a class "Person" and with attributes name, age and gender.
Create the getter and setter methods for each attribute in "Person" class and display()
method to display the information of the person. Use the "Person" class in the "Main"
class by creating the object of the "Person" class.

4. Java Program With Inheritance and Polymorphism:

Write a Java program to create a class called "Reservation"" with attributes for reservation_id, customer_name, and reservation_date. Create subclasses "ResortReservation" and "RailwayReservation" that add specific attributes like room_no, room_type, room_charge for hotels and seat_no, coach_type, ticket_price. Implement methods as given in the chart below and can add some extra attributes and methods as per your choice.

Note:

a. Implement method Override the display() method from the superclass Reservation.

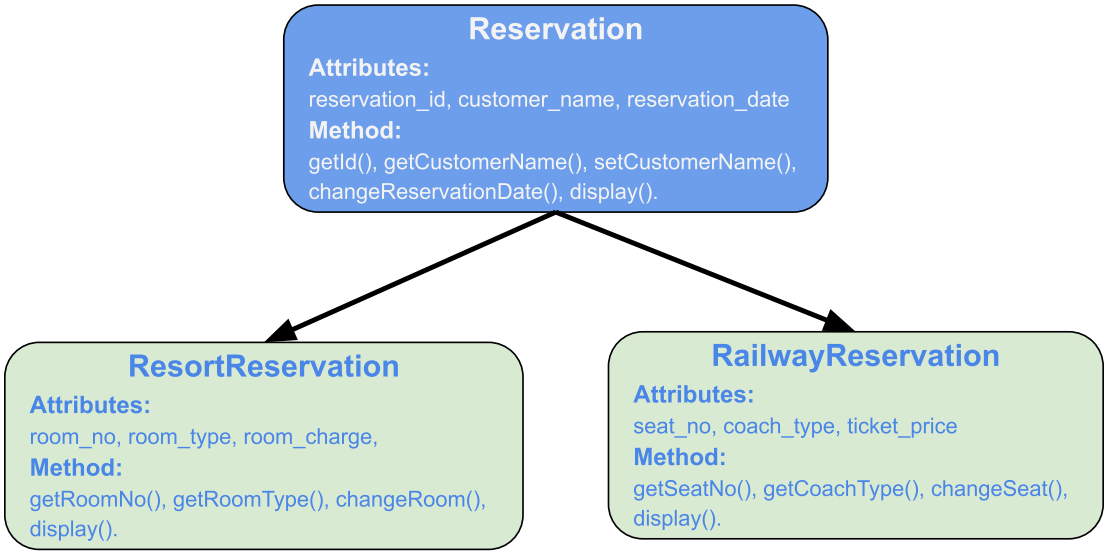
b. Implement method overloading for

   i. changeRoom(<int>, <int>) => change Room number

   ii. changeRoom(<String>, <String>) => change Room Type

   iii. changeSeat(<int>, <int>) => change Seat Number

   iv. changeSeat(<String>, <String>) =>change Coach Type



Source Code:

```
import java.util.Objects;
import java.util.Scanner;
import java.util.Random;
import java.time.LocalDate;
```

```
class Reservation{
    Random r = new Random();
    Scanner sc = new Scanner(System.in);
    LocalDate date = LocalDate.now();

    int reservation_id;
    String customer_name, reservation_date;

    int getId(){
        reservation_id = r.nextInt(100);
        return reservation_id;
    }

    void setCustomerName(String customer_name){

        this.customer_name = customer_name;
    }

    String getCustomerName(){

        return customer_name;
    }

    void setReservation_date(){

        System.out.println("Reservation Date = "+date);
    }

    void changeReservationDate(){
```

```
        System.out.println("Enter a date for reservation in format (YYYY-MM-DD)
: ");
        reservation_date = sc.nextLine();
    }


    void display(){
        System.out.println("Name : "+getCustomerName());
        System.out.println("Reservation ID : "+getId());
        setReservation_date();
    }
}


class ResortReservation extends Reservation{
    Random r = new Random();


    int room_no;
    String room_type;
    double room_charge = r.nextDouble(2000);


    int getRoomNo(){
        room_no = r.nextInt(10);
        return room_no;
    }


    void setRoomType(String room_type){
        this.room_type = room_type;
    }


    String getRoomType(){


        return room_type;
    }
```

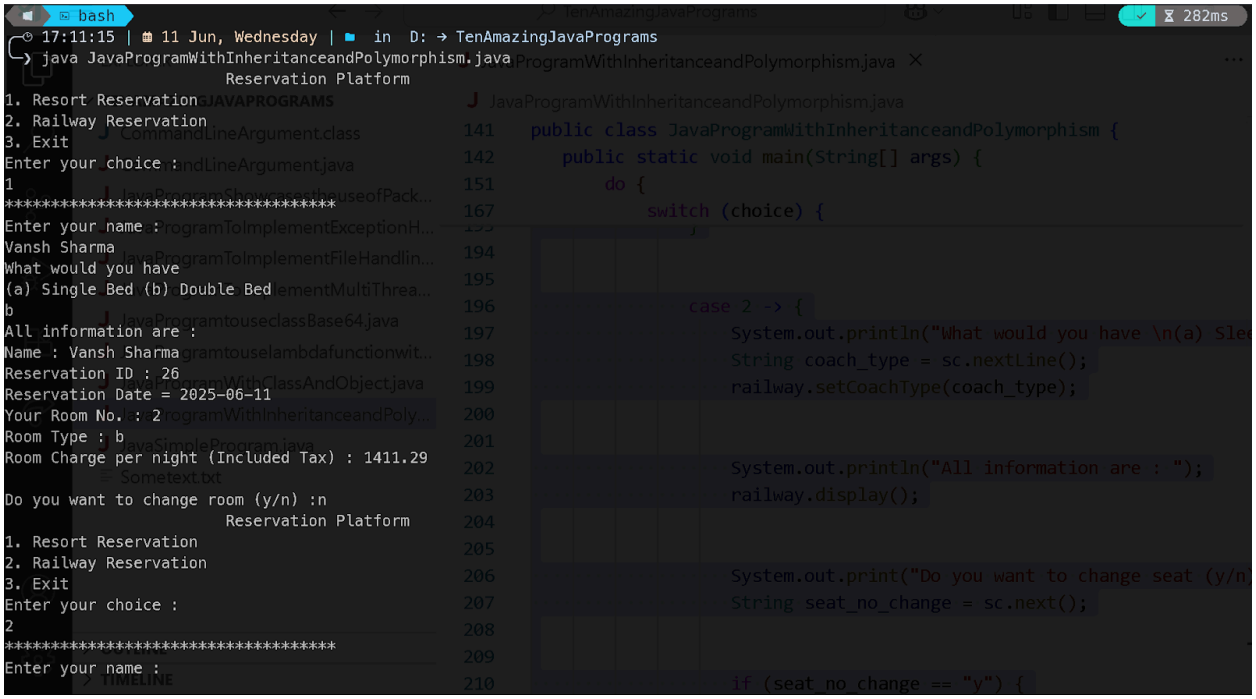
```
    void changeRoom(String room_type){

        this.room_type = room_type;
    }

    @Override
    void display() {
        super.display();
        System.out.println("Your Room No. : "+getRoomNo());
        System.out.println("Room Type : "+getRoomType());
        System.out.println("Room Charge per night (Included Tax) :
"+String.format("%.2f",room_charge));
        System.out.println(" ");
    }
}

class RailwayReservation extends Reservation{
    int seat_no,ticket_price;
    String coach_type;

    int getSeatNo(){
        seat_no = r.nextInt(50);
        return seat_no;
    }

    String getCoachType(){
        return coach_type;
    }

    void setCoachType(String coach_type){
        this.coach_type = coach_type;
    }
```

```
    void changeSeat(){
        //No need of this method.
    }


    @Override
    void display() {
        super.display();
        System.out.println("Your Seat No. : "+getSeatNo());
        System.out.println("Coach Type : "+getCoachType());
        System.out.println(" ");
    }
}


public class JavaProgramWithInheritanceandPolymorphism {
    public static void main(String[] args) {
        ResortReservation resort = new ResortReservation();
        RailwayReservation railway = new RailwayReservation();
        Scanner sc = new Scanner(System.in);


        int choice;


        do {
            System.out.println("\t\t\tReservation Platform");
            System.out.println("1. Resort Reservation\n2. Railway
Reservation\n3. Exit");
            System.out.println("Enter your choice : ");
            choice = sc.nextInt();
            if (choice != 3)
                sc.nextLine();
            System.out.println("************************************");
            System.out.println("Enter your name : ");
            String customer_name = sc.nextLine();
            resort.setCustomerName(customer_name);
            railway.setCustomerName((customer_name));
```

```
            switch (choice) {
                case 1 -> {
                    System.out.println("What would you have \n(a) Single Bed
(b) Double Bed");
                    String room_type = sc.nextLine();
                    resort.setRoomType(room_type);


                    System.out.println("All information are : ");
                    resort.display();


                    System.out.print("Do you want to change room (y/n) :");
                    String room_change = sc.nextLine();




                    if (Objects.equals(room_change, "y")) {
                        System.out.println("What would you have \n(a) Single
Bed (b) Double Bed");
                        room_type = sc.nextLine();
                        resort.setRoomType(room_type);
                    } else {
                        break;
                    }


                }


                case 2 -> {
                    System.out.println("What would you have \n(a) Sleeper Class
(b) AC Coach (c) 2S Coach");
                    String coach_type = sc.nextLine();
                    railway.setCoachType(coach_type);
```

```
                    System.out.println("All information are : ");
                    railway.display();


                    System.out.print("Do you want to change seat (y/n) : ");
                    String seat_no_change = sc.next();


                    if (seat_no_change == "y") {
                        System.out.println(railway.getSeatNo());
                    } else {
                        break;
                    }


                }


                case 3 -> {
                    System.exit(0);
                }
            }
        }while (choice!=3);
    }
}
```

Output:


bash
282ms
17:11:15 | 11 Jun, Wednesday | in D: → TenAmazingJavaPrograms
java JavaProgramWithInheritanceandPolymorphism.java
Reservation Platform
1. Resort Reservation
2. Railway Reservation
3. Exit
Enter your choice :
1
****
Enter your name :
Vansh Sharma
What would you have
(a) Single Bed (b) Double Bed
b
All information are :
Name : Vansh Sharma
Reservation ID : 26
Reservation Date = 2025-06-11
Your Room No. : 2
Room Type : b
Room Charge per night (Included Tax) : 1411.29
Do you want to change room (y/n) :n
Reservation Platform
1. Resort Reservation
2. Railway Reservation
3. Exit
Enter your choice :
2
****
Enter your name :
JavaProgramWithInheritanceandPolymorphism.java
141 public class JavaProgramWithInheritanceandPolymorphism {
142 public static void main(String[] args) {
151 do {
167 switch (choice) {
196 case 2 -> {
197 System.out.println("What would you have \n(a) Sle
198 String coach_type = sc.nextLine();
199 railway.setCoachType(coach_type);
202 System.out.println("All information are : ");
203 railway.display();
206 System.out.print("Do you want to change seat (y/n
207 String seat_no_change = sc.next();
210 if (seat_no_change == "y") {

5. Java Program To Implement Exception Handling:

Write a Java Program to create your own class “Registration” which is used to register the details of an User, with attributes user_id, user_name, password, mobile_no, email_id. Create the as follows:

a. Create a constructor which automatically creates the user_id.
b. Create the methods to validate the credentials such user_name, password, email, mobile_no.
c. Create your own Exception and throw them when an invalid credential is encountered.
    i. InvalidEmailIdException => for an invalid email format.
    ii. InvalidContactNumberException => for an invalid mobile no.
        1. Its length must be 10 and have only numbers.
    iii. InvalidUserNameException => for invalid user name
        1. User name only contains uppercase, lowercase and ‘_’ letters.
    iv. InsecurePasswordException => for an easy password
        1. Password must have at least 8 letters.
        2. Password must have at least an Uppercase and special Symbol.

Source Code{JavaProgramToImplementExceptionHandling.java}:

```
import java.util.Scanner;
import java.util.Random;
import java.util.regex.Pattern;
import java.util.regex.Matcher;


class Registration{
   Random r = new Random();


   int user_id;
   long mobile_no;
   String user_name,password,email_id;


   Registration(){
       this.user_id = r.nextInt(10000);
   }
```

```
void setUserName(String user_name){
    this.user_name = user_name;
}


String getUserName(){
    return user_name;
}


void setPassword(String password){
    this.password = password;
}


String getPassword(){
    return password;
}


void setMobileNo(long mobile_no){
    this.mobile_no = mobile_no;
}


long getMobileNo(){
    return mobile_no;
}


void setEmailID(String email_id){
    this.email_id = email_id;
}


String getEmailID(){
    return email_id;
}
```

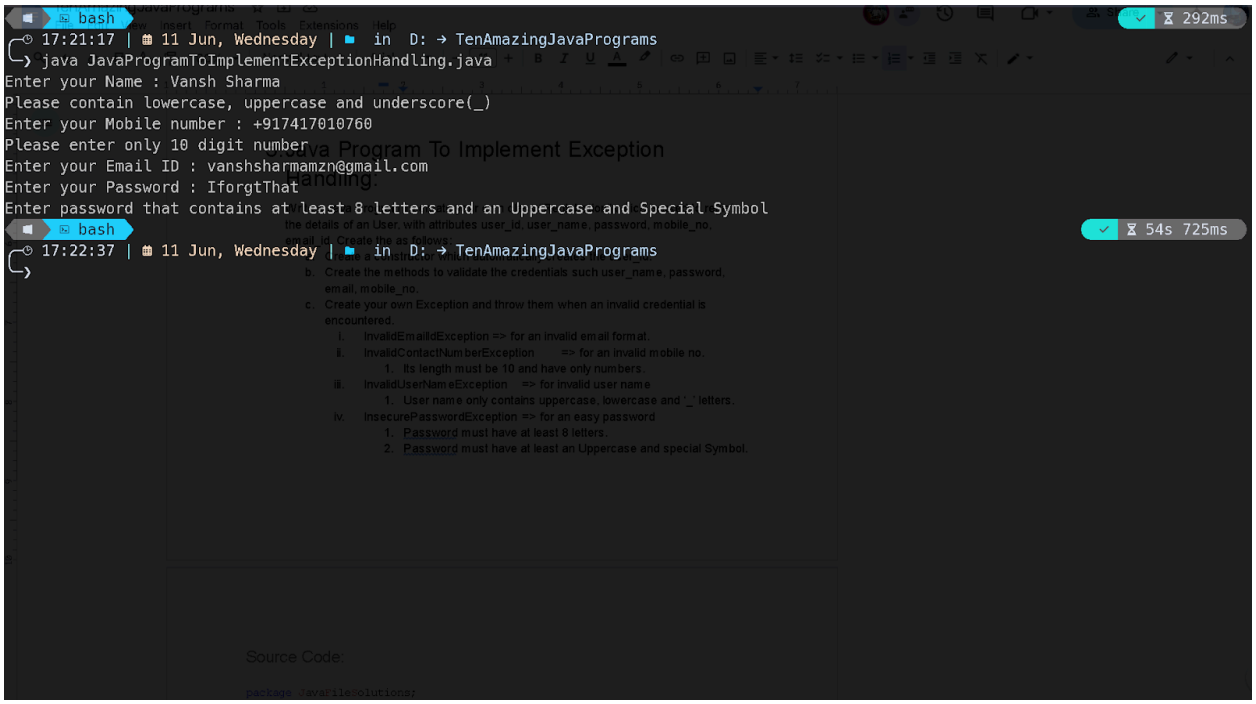
```
   boolean checkUserName(){
       return Pattern.matches("^[a-zA-z][a-zA-z0-9_]{2,10}$",user_name);
   }


   boolean checkPassword(){
       return
Pattern.matches("^(?=.*[a-z])(?=.*[A-Z])(?=.*[!@#$%^&*()])(?=.*[0-9]).{8,20}$"
,password);
   }


   boolean checkEmail(){
       return
Pattern.matches("^[a-zA-Z0-9_+&*-]+(?:\\.[a-zA-Z0-9_+&*-]+)*@(?:[a-zA-Z0-9-]+\
\.)+[a-zA-Z]{2,7}$",email_id);
   }


   boolean checkMobileNo(){
       String mobNo = Long.toString(mobile_no);
       return Pattern.matches("^[0-9][0-9]{9}$",mobNo);
   }
}


class InvalidUserNameException extends Exception{
   InvalidUserNameException(){

       super("You entered username in wrong format");
   }
}


class InvalidContactNumberException extends Exception{
   InvalidContactNumberException(){
       super("You entered more than 10 digits or enter alphabets");
   }
}
```

```
class InvalidEmailException extends Exception{
   InvalidEmailException(){
       super("Wrong Email Format");
   }
}


class InvalidPasswordException extends Exception{
   InvalidPasswordException(){
       super("Wrong Password Format");
   }
}


public class JavaProgramToImplementExceptionHandling {
   public static void main(String[] args) {
       Registration regis = new Registration();
       Scanner sc = new Scanner(System.in);




       try{
           System.out.print("Enter your Name : ");
           String user_name = sc.nextLine();
           regis.setUserName(user_name);


           boolean username = regis.checkUserName();
           if(!username) {
               throw new InvalidUserNameException();
           }
       }
       catch(InvalidUserNameException e){
           System.out.println("Please contain lowercase, uppercase and
underscore(_)");
       }
```

```
try{
    System.out.print("Enter your Mobile number : ");
    long mobile_no = sc.nextLong();
    regis.setMobileNo(mobile_no);


    boolean mobile = regis.checkMobileNo();
    if (!mobile) {
        throw new InvalidContactNumberException();
    }
}


catch(InvalidContactNumberException e){
    System.out.println("Please enter only 10 digit number");
}


sc.nextLine();
try{
    System.out.print("Enter your Email ID : ");
    String email_id = sc.nextLine();
    regis.setEmailID(email_id);


    boolean email = regis.checkEmail();
    if (!email) {
        throw new InvalidEmailException();
    }
}


catch(InvalidEmailException e){
    System.out.println("Enter an email with right format");
}


try {
    System.out.print("Enter your Password : ");
    String password = sc.nextLine();
```

```
            regis.setPassword(password);


            boolean pass = regis.checkPassword();
            if (!pass)
                throw new InvalidPasswordException();
        }


        catch(InvalidPasswordException e){
            System.out.println("Enter password that contains at least 8 letters
and an Uppercase and Special Symbol");
        }
    }
}
```

Output:



6. Java Program To Implement File Handling:

Write a menu oriented Java program to perform all 4 CRUD{ create, read, update, delete} operations on a basic text file. Make the program modular and user friendly.

Source Code{JavaProgramToImplementFileHandling.java}:

```
import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.FileReader;
import java.io.FileWriter;
import java.io.FileNotFoundException;
import java.io.IOException;
import java.nio.file.FileAlreadyExistsException;
import java.io.File;
import java.util.Scanner;

class FileHandeler {
    public File openFile(String filename) {
        File file = new File(filename);
        return file;

    }

    public File createFile(String filename) {
        try {
            File file = new File(filename);
            file.createNewFile();
            return file;
        }
        catch (FileAlreadyExistsException e) {
            System.out.println("File Already exists");
            return null;
        }
        catch (IOException e) {
            System.out.println("An error occurred.");
            return null;
        }
    }
```

```
    public void renameFile(String oldname, String newname) {
        File oldfile = new File(oldname);
        File newfile = new File(newname);
        if(oldfile.renameTo(newfile)) {
            System.out.println("File Renamed Successfully");
        } else {
            System.out.println("Error : Cannot Rename File");
        }
    }

    public void copyFile(File copyfileobj, File fileobj) throws IOException {
        copyFile(copyfileobj, fileobj, false);
    }
    public void copyFile(File copyfileobj, File fileobj, boolean append)
throws IOException {
        copyFile(copyfileobj.getName(), fileobj.getName(), append);
    }
    public void copyFile(String copyfilename, String filename, boolean append)
{
        try {
            FileReader file = new FileReader(filename);
            FileWriter copyfile = new FileWriter(copyfilename, append);
            BufferedReader bufferreader = new BufferedReader(file);

            String buffer;
            while ((buffer = bufferreader.readLine()) != null) {
                System.out.println(buffer+"\n");
                copyfile.write(buffer+"\n");
            }
            file.close();
            copyfile.close();
        }
        catch(FileNotFoundException e) {
            System.out.println("Error : File not Found");
        }
        catch (IOException e) {
            System.out.println("Error : cannot display File");
        }
    }
```

```
    public void readFile(File file, boolean displayflag) throws IOException {
        readFile(file.getName(), displayflag);
    }
    public FileReader readFile(File file) throws IOException {
        return readFile(file.getName());
    }
    public FileReader readFile(String filename) {
        try {
            FileReader readfile = new FileReader(filename);
            return readfile;
        }
        catch(FileNotFoundException e) {
            System.out.print("Error : File is not Found");
            return null;
        }
    }

    public void readFile(String filename, boolean displayflag) throws
IOException {
        try {
            if(displayflag) {
                FileReader file = readFile(filename);
                BufferedReader bufferreader = new BufferedReader(file);
                String buffer;
                while ((buffer = bufferreader.readLine()) != null) {
                    System.out.println(buffer);
                }
                file.close();
            }
        }
        catch (IOException e) {
            System.out.println("Error : cannot display File");
        }
    }

    public void writeLinesInFile(String filename, String[] lines) {
        for(String line: lines){
            writeLineInFile(filename,line);
        }
```

```
    }
    public void writeLinesInFile(File file, String[] lines) {
        for(String line: lines){
            writeLineInFile(file.getName(),line);
        }
    }
    public void writeLineInFile(File file, String line) {
        writeLineInFile(file.getName(),line);
    }
    public void writeLineInFile(String filename,String line) {
        try (BufferedWriter writer = new BufferedWriter(new
FileWriter(filename, true))) {
            writer.write(line);
            writer.newLine();
        } catch (IOException e) {
            System.err.println("Error writing to file: " + e.getMessage());
        }
    }

    public boolean deleteFile(File file) {
        return deleteFile(file.getName());
    }
    public boolean deleteFile(String filename) {
        File file = new File(filename);
        if(file.delete()) {
            return true;
        }
        return false;
    }

}

class Crud extends FileHandeler{
    Scanner scan = new Scanner(System.in);
    boolean flag = true;
    private int menu() {
        System.out.print("""
        +-------------------------------------------------------+
        |            Welcome To CRUD Service Software           |
        +------+------------------------------------------------+
```

```
        |  1.  |        Create File                                |
        +------+---------------------------------------------------+
        |  2.  |        Read File                                  |
        +------+---------------------------------------------------+
        |  3.  |        Write File                                 |
        +------+---------------------------------------------------+
        |  4.  |        Copy File                                  |
        +------+---------------------------------------------------+
        |  5.  |        Delete File                                |
        +------+---------------------------------------------------+
        |  6.  |        Rename File                                |
        +------+---------------------------------------------------+
        |  Enter Your Choice = >""");
            return scan.nextInt();
    }
    private String input(String prompt) {
        System.out.print(prompt);
        String inp = scan.nextLine();
        inp = scan.nextLine();
        return inp;
    }

    private void mainloop() throws IOException {
        switch (menu()) {
            case 1 -> {
                createFile(input("Enter File name : "));

            }
            case 2 -> {
                readFile(input("Enter File name : "), true);
                input("");
            }
            case 3 -> {
                String filename = input("Enter File name :");
                System.out.println("NOTE Type \":EOF<save>\" to save and exit
the file");
                System.out.println("=========="+filename+"============");
                String buffer = input("");
                while(!buffer.equals(":EOF<save>")){
                    writeLineInFile(filename, buffer);
```

```
                    buffer = scan.nextLine();
                }
            }
            case 4 -> {
                Scanner scan = new Scanner(System.in);
                System.out.print("Enter Source File name : ");
                String source = scan.nextLine();
                System.out.print("Enter Destination File name : ");
                String destination =  scan.nextLine();
                copyFile(destination, source, false);
            }
            case 5 -> {
                deleteFile(input("Enter File name : "));
            }
            case 6 -> {
                Scanner scan = new Scanner(System.in);
                System.out.print("Enter old name : ");
                String oldname = scan.nextLine();
                System.out.print("Enter new name : ");
                String newname =  scan.nextLine();
                renameFile(oldname, newname);
            }
            default -> {
                flag = false;
            }
        }
    }
    public void clear() {
        System.out.print("\033[H\033[2J");
        System.out.flush();
    }
    public void start() throws Exception {
        while(flag){
            clear();
            mainloop();
        };
    }
}
public class JavaProgramToImplementFileHandling {
    public static void main(String[] args) throws Exception{
```

```
        Crud software = new Crud();
        software.start();
    }
}
```

Output:

```
+------------------------------------------------------+
|           Welcome To CRUD Service Software           |
+------+-----------------------------------------------+
|  1.  |        Create File                            |
+------+-----------------------------------------------+
|  2.  |        Read File                              |
+------+-----------------------------------------------+
|  3.  |        Write File                             |
+------+-----------------------------------------------+
|  4.  |        Copy File                              |
+------+-----------------------------------------------+
|  5.  |        Delete File                            |
+------+-----------------------------------------------+
|  6.  |        Rename File                            |
+------+-----------------------------------------------+
|  Enter Your Choice = >2
Enter File name : CommandLineArgument.java
```

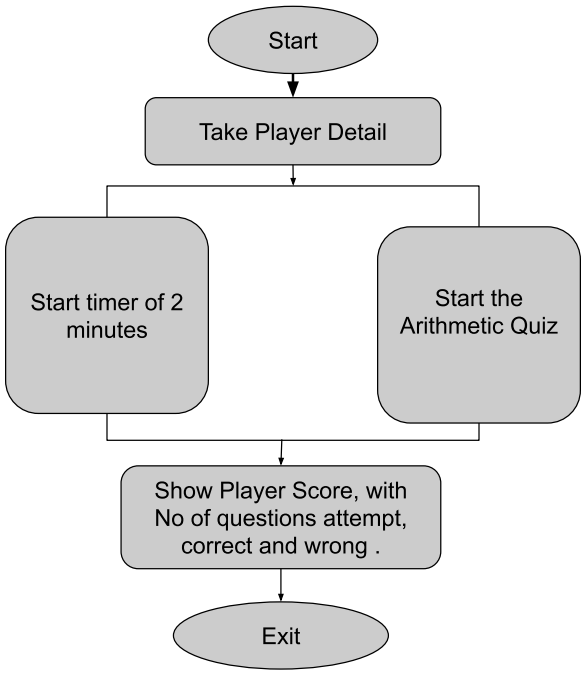
1. Java Program(Command Line Argument):

```
/*
 * Write a java program to take the number from the user from the command line and
 * check whether the number is palindrome or not.
 * E.g. if the number is 12321 then its reverse i.e. 12321 equals to the actual number.
 */
public class CommandLineArgument {
    public static void main(String[] args){
        int num = Integer.parseInt(args[0]);

        int numcpy = num, newnum = 0;

        while(numcpy != 0){
            newnum *= 10;
            newnum += numcpy%10;
            numcpy /= 10;
        }
        System.out.println("The Number "+num+" is "+(num == newnum?"":"not ")+"Palindrome");
    }
}
```

7. Java Program To Implement Multi Threading:

Create a Mathematics test in java to test the speed and accuracy of the player in solving the arithmetic expressions with a time limit of 2 mins.

a. The score is displayed at the end of the test.
b. The arithmetic expressions must be generated randomly.



Source Code{JavaProgramToImplementMultiThreading.java}:

```
import java.util.ArrayList;
import java.util.Scanner;
import java.util.Date;

class Timer extends Thread {
    final public int DURATION;
    Date startTime, nowTime;
    long timediff;
    public boolean isTimeOver = false;
    public boolean isOver = false;
    Timer(){
        this(2 * 60 * 1000);
    }
    Timer(int time){
        DURATION = time;
        startTime = new Date();
```

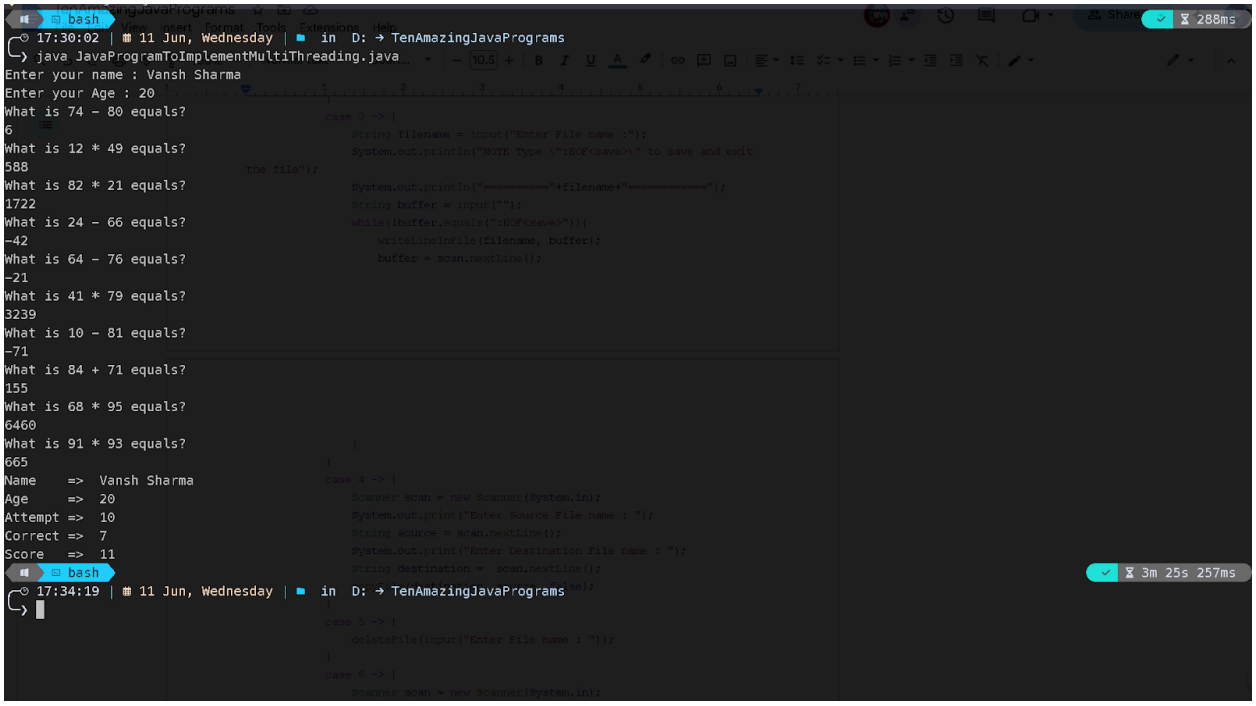
```
            nowTime = new Date();
            timediff = nowTime.getTime() - startTime.getTime();
        }
        public void run(){
            while(timediff < DURATION && !isOver){
                nowTime = new Date();
                timediff = nowTime.getTime() - startTime.getTime();
            }
            if(timediff >= DURATION){
                isTimeOver = true;
            }
        }
    }
    class Ques{
        int num1, num2;
        char operator;
        int answer;
        Ques(int num1, int num2, char operator){
            this.num1 = num1;
            this.num2 = num2;
            this.operator = operator;
            answer = getAnswer();
        }
        public int getAnswer(){
            return switch(operator){
                case '+'->  num1+num2;
                case '-'->  num1-num2;
                case '%'->  num1%num2;
                case '*'->  num1*num2;
                case '/'->  num1/num2;
                default ->  -1;
            };
        }
        public void display(){
            System.out.printf("What is %d %c %d equals?\n",num1,operator, num2);
        }
    }
    class QuesHandeler{
        ArrayList<Ques> questions;
        int length;
```

```
    QuesHandeler(int num){
        this(num, 1, 100);
    }
    QuesHandeler(int num, int min, int max){
        length = num;
        questions =  new ArrayList<>();
        for(int i = 0; i < num; i++){
            int num1 = random(min, max);
            int num2 = random(min, max);
            char operator = randamo();
            questions.add(new Ques(num1, num2, operator));
        }
    }
    private int getAnswer(int index){
        return questions.get(index).getAnswer();
    }
    public boolean isCorrect(int answer, int index){
        return answer == getAnswer(index);
    }
    public void display(int index){
        Ques ques = questions.get(index);
        ques.display();
    }
    int random(int start, int end){
        double temp = Math.random()*1000;
        int rand = (int)(start + temp%(end - start));
        return rand;
    }
    char randamo(){
        char[] operators = {'+', '-', '*', '/', '%'};

        return operators[random(0,5)];
    }
}
class Player extends QuesHandeler{
    String name;
    short age;
    int ques_attempt;
    int ques_correct;
    int score;
```

```
    int question_no = 0;
    Player(String name, int age){
        this(name, age, 100);
    }
    Player(String name, int age, int num){
        super(num);
        this.name = name;
        this.age = (short)age;
    }
    public void askQuestion(){
        Scanner scan = new Scanner(System.in);
        display(question_no);
        int ans = scan.nextInt();
        if(isCorrect(ans, question_no)){
            score+=2;
            ques_correct++;
        }
        else{
            score--;
        }
        question_no++;
    }
    public boolean start() {
        if(length > question_no){
            askQuestion();
            return true;
        }
        else{
            System.out.println("The Questions are Over");
            return false;
        }
    }
    public void display(){
        System.out.printf("""
Name    =>  %s
Age     =>  %d
Attempt =>  %d
Correct =>  %d
Score   =>  %d
                """,name, age, question_no, ques_correct, score);
```

```
    }
}
class Quiz{
    Timer timer_thread;
    Player player;
    Quiz(){
        this(1000, 2 * 60 * 1000);
    }
    Quiz(int num,int time){
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter your name : ");
        String name = scan.nextLine();
        System.out.print("Enter your Age : ");
        int age = scan.nextInt();
        timer_thread = new Timer(time);
        player = new Player(name, age, num);

    }
    Quiz(String name, int age, int time){
        this(name, age, 100, time);
    }
    Quiz(String name, int age, int num, int time ){
        timer_thread = new Timer(time);
        player = new Player(name, age, num);
    }
    public void start(){
        timer_thread.start();
        while(!timer_thread.isTimeOver && player.start());
        player.display();
    }
}
class JavaProgramToImplementMultiThreading{
    public static void main(String[] args) {
        Quiz obj = new Quiz();
        obj.start();
    }
}
```

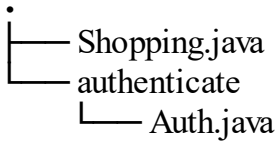
Output:


bash
17:30:02 | 11 Jun, Wednesday | in D: → TenAmazingJavaPrograms
java JavaProgramToImplementMultiThreading.java
Enter your name : Vansh Sharma
Enter your Age : 20
What is 74 - 80 equals?
6
What is 12 * 49 equals?
588
What is 82 * 21 equals?
1722
What is 24 - 66 equals?
-42
What is 64 - 76 equals?
-21
What is 41 * 79 equals?
3239
What is 10 - 81 equals?
-71
What is 84 + 71 equals?
155
What is 68 * 95 equals?
6460
What is 91 * 93 equals?
665
Name => Vansh Sharma
Age => 20
Attempt => 10
Correct => 7
Score => 11
288ms
bash
17:34:19 | 11 Jun, Wednesday | in D: → TenAmazingJavaPrograms
3m 25s 257ms

8. Java Program Showcases the use of Packages in java.

Create a package name “authenticate” in java with the “ Auth” class .

a. The class “Auth” having the attribute user_id, user_name, password and is_login
b. Having methods:
    i. login(user_name, password) => to login the user.
    ii. logout(user_name) => to logout the user.
    iii. isUser(user_name) => to check user_name is present
    iv. isLogin(user_name) => to check the user is login or not
    v. changePassword(password, new_password) => to change the user password
c. use this in a Source file with class “Shopping”.
d. Package tree Structure.



Source Code{Shopping.java}:

```
import JavaFileSolutions.authenticate.Auth;
import java.util.Arrays;
import java.util.Scanner;


public class Package {
   public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        Auth a1 = new Auth();


        int[] checkSlot = new int[100];
        Arrays.fill(checkSlot,0);
        do{
            System.out.print("Enter the slot { Between 1 to 100 } : ");
            int i = sc.nextInt();
            checkSlot[i] = 1;
            if (checkSlot[i] == 1){
```

```
System.out.println("""
                   _____________________________________________
                   |              Shopping Center              |
                   |___________________________________________|
                   | 1. Registration                           |
                   |-------------------------------------------|
                   | 2. Login                                  |
                   |-------------------------------------------|
                   | 3. Change Password                        |
                   |-------------------------------------------|
                   | 4. Check User                             |
                   |-------------------------------------------|
                   | 5. Check Login                            |
                   |-------------------------------------------|
                   | 6. Logout                                 |
                   |-------------------------------------------|
                   |7. Exit                                    |
                   |___________________________________________|
                   """);
System.out.print("Enter your choice : ");
int choice = sc.nextInt();
sc.nextLine();
switch (choice){
    case 1 ->{
        a1.register();
    }


    case 2 ->{
        System.out.print("Enter your name : ");
        String name = sc.nextLine();
        System.out.print("Enter your password : ");
        String password = sc.nextLine();
        a1.login(name , password);
    }


    case 3 ->{
        String old_pass , new_pass;
        System.out.print("Enter your old password : ");
```

```
                    old_pass = sc.nextLine();
                    System.out.print("Enter your new password : ");
                    new_pass = sc.nextLine();
                    a1.changePassword(old_pass , new_pass);
                }


                case 4 ->{
                    System.out.print("Enter your Username : ");
                    String username = sc.nextLine();
                    System.out.println(a1.isUser(username));
                }


                case 5 ->{
                    System.out.println("Enter your Username : ");
                    String username = sc.nextLine();
                    boolean is_login = a1.isLogin(username);
                    if (is_login)
                        System.out.println("You are already login");
                    else
                        System.out.println("You are not login");
                }


                case 6 ->{



                }


                case 7 ->{
                    System.exit(0);
                }
            }
        }
        else
            System.out.println("This slot is already filled.");
    }while(true);
}
```

```
}
```

Source Code{Shopping.java}{Auth.java}:

```
package authenticate;

import java.util.HashMap;
import java.util.Objects;
import java.util.Scanner;

class User{
   String user_name,password;
   boolean is_login;


   User(){}


   User(User user){
       this.user_name = user.user_name;
       this.password = user.password;
       this.is_login = user.is_login;
   }
}


public class Auth {


   HashMap<String, User> hash = new HashMap<>();
   User user = new User();
   Scanner sc = new Scanner(System.in);




   public boolean isLogin(String user_name){
       if (Objects.equals(user_name, user.user_name))
           user.is_login = true;
       return user.is_login;
```

```
    }


    public void logout(String user_name){
        if (hash.containsKey(user_name))
            hash.get(user_name).is_login = false;
    }


    public void login(String user_name, String password){


        if (hash.containsKey(user_name)) {
            if (hash.get(user_name).password.equals(password)) {
                hash.get(user_name).is_login = true;
                System.out.println("Login Successfully");
            }
            else
                System.out.println("Password is incorrect");
        }
        else
            System.out.println("Incorrect Username");
    }


    public String isUser(String user_name){
        return ((user_name.equals(user.user_name)?"Yes, You are
registered":"No, You are not registered"));
    }


    public void register(){
        System.out.println("
----------------------Registration-------------------");
        System.out.print("\tEnter your Name : ");
        user.user_name = sc.nextLine();
        System.out.print("\tEnter your Password : ");
        user.password = sc.nextLine();
        hash.put(user.user_name,new User(user));
        System.out.println("You are successfully registered\n");
```

```
        System.out.println("Now you can login 😊");
    }


    public void changePassword(String old_pass , String new_pass){
        if (old_pass.equals(user.password))
            user.password = new_pass;
        else
            System.out.println("Old Password is incorrect");
    }
}
```

Output:

9. Java Program to use lambda function with interface.

Write a java program to create an interface with the name “Shape” having an abstract method name area(). Use the interface “Shape” in the main class using lambda Expression by overriding the method area(). Create the objects with lambda expression for the following:

Object name	Formula use { double area = }
circle	Math.PI * radius * radius
square	side * side
triangle	1/2 * base * height

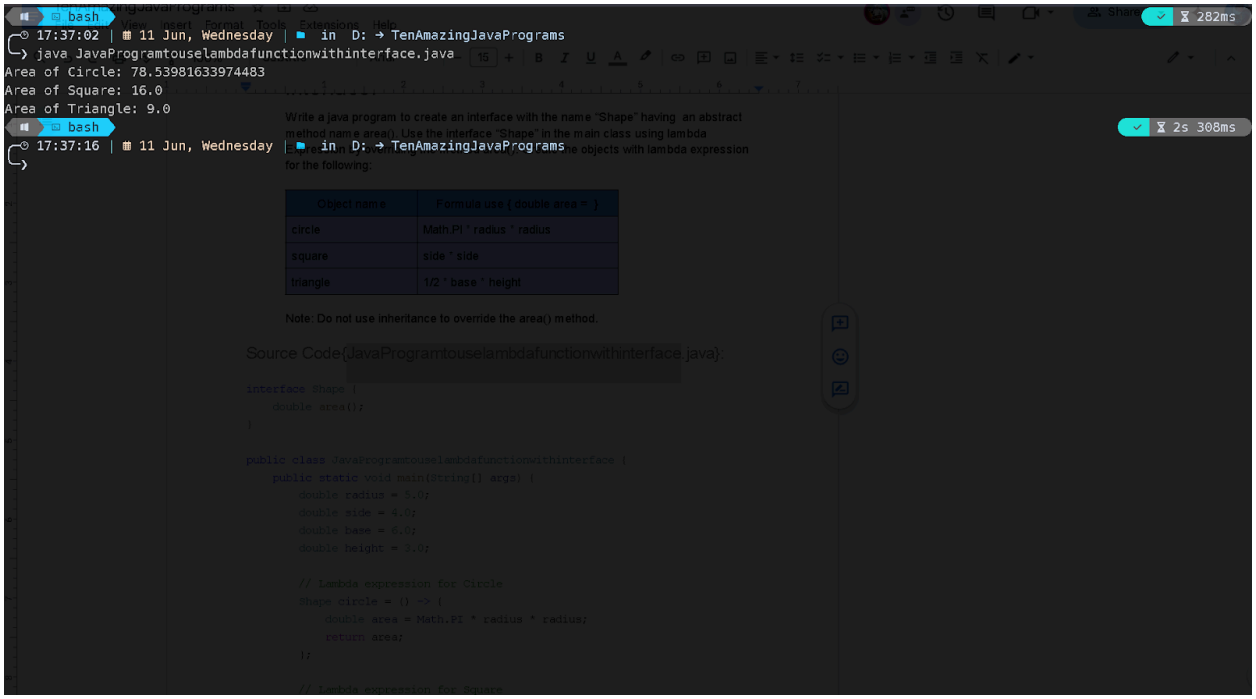
Note: Do not use inheritance to override the area() method.

Source Code{JavaProgramtouselambdafunctionwithinterface.java}:

```
interface Shape {
    double area();
}

public class JavaProgramtouselambdafunctionwithinterface {
    public static void main(String[] args) {
        double radius = 5.0;
        double side = 4.0;
        double base = 6.0;
        double height = 3.0;

        // Lambda expression for Circle
        Shape circle = () -> {
            double area = Math.PI * radius * radius;
            return area;
        };

        // Lambda expression for Square
        Shape square = () -> {
            double area = side * side;
            return area;
```

```
        };

        // Lambda expression for Triangle
        Shape triangle = () -> {
            double area = 0.5 * base * height;
            return area;
        };

        System.out.println("Area of Circle: " + circle.area());
        System.out.println("Area of Square: " + square.area());
        System.out.println("Area of Triangle: " + triangle.area());
    }
}
```

Output:



10. Java Program to use class Base64.

Write a menu oriented program to take choice from the user to encode or decode the messages using Base64 class in java.util package.

Source Code{JavaProgramtouseclassBase64.java}:

```
import java.util.Base64;
import java.util.Scanner;

class Secret {
    // Encoding
    public static String encodeMessage(String phraseString) {
        byte[] encodedBytes =
Base64.getEncoder().encode(phraseString.getBytes());
        String encodedString = new String(encodedBytes);

        return encodedString;

    }

    // Decoding
    public static String decodeMessage(String encodedString) {
        byte[] decodedBytes = Base64.getDecoder().decode(encodedString);
        String decodedString = new String(decodedBytes);

        return decodedString;
    }

}

public class JavaProgramtouseclassBase64 {
    public static int menu(){
        try{
            Scanner scan = new Scanner(System.in);
            System.out.print("""
                +-------------------------------------------------------+
                |            Welcome To Secret Software                 |
                +------+------------------------------------------------+
                |  1.  |        Encrypt Message                         |
```

```
            +------+-------------------------------------------------+
            |  2.  |        Decrypt Message                          |
            +------+-------------------------------------------------+
            |  Enter Your Choice = >""");
        return scan.nextInt();
    } catch(Exception e) {
        System.out.println(e.getMessage());
        return -1;
    }
}

public static boolean mainloop() {
    switch(menu()) {
        case 1 -> {
            System.out.print("Enter Message to Encode : ");
            try {
                Scanner scan = new Scanner(System.in);
                String message = scan.nextLine();
                System.out.println(Secret.encodeMessage(message));
            } catch (Exception e) {
                System.out.println(e.getMessage());
            }
        }
        case 2 -> {
            System.out.print("Enter Message to Decode : ");
            try {
                Scanner scan = new Scanner(System.in);
                String message = scan.nextLine();
                System.out.println(Secret.decodeMessage(message));
            } catch (Exception e) {
                System.out.println(e.getMessage());
            }
        }
        default -> {
            return false;
        }
    }
    return true;
}
public static void main(String[] args) {
```

```
        while(mainloop());
    }
}
```

Output: