Ex. No: 1 **DATE: 09-08-23**

BASICS OF JAVA PROGRAMMING

Aim: To explore about basics of java programming

Algorithm:

Step 1: **To print Hello world in Java**

- public static void main(String args[]): The main method is the entry point of the program, where execution begins.
- System.out.print() → Used to print the data in output console

Step 2: **To display user profile**

- in.close(): Closes the Scanner object to release system resources.
- in.nextLine(): Reads the entire line of text as input, which is used for capturing the user's name and address.
- in.nextInt(): Reads an integer input, which is used for capturing the user's register number.
- in.next(): Reads a single word or token of input, which is used for capturing the user's phone number

Step 3: **To add two numbers**

- Scanner in = new Scanner(System.in): Creates a Scanner object for reading user input from the console.
- System.out.println(String s): Prints the specified string to the standard output.
- in.nextInt(): Reads an integer input from the user, capturing the first number.

Step 4: **To display welcome note**

- input.nextLine(): Reads the entire line of text as input, capturing the user's name.
- input.nextLong(): Reads a long integer input from the user, capturing the user's register number.
- String concatenation: The code uses concatenation (+) to create a formatted welcome message that includes the user's name and register number.

Step 5: **To build chat application**

- Scanner in = new Scanner(System.in): Creates a Scanner object for reading user input from the console.
- ArrayList<String> chats = new ArrayList<>(): Creates an ArrayList to store chat messages.
- in.nextLine(): Reads the entire line of text as input, capturing chat messages for both "Person 1" and "Person 2."
- in.close() : Closes the Scanner object to release system resources

Step 6: **To compile and run a java program**

- javac Filename.java → To compile the java program
- java Classname → To run the java program

Step 7: **To print current date and time**

- java.time.LocalDate.now() → To get current System Date
- java.time.LocalTime.now() → To get current System Time

Step 8: **Identify the most error-prone one-line output & provide a brief description.**

- System.out.print() → Used to print the data in output console
- System.out.print(args[0]) → Used to print the cmdline input in output console

Step 9: **Identify the most error-prone statement and rectify them.**

- new Scanner(System.in) → Initialize the Scanner object
- in.nextInt() → Get the integer input from the user in command line during execution

Step 10: **To know about Data types in Java**

- int: -2,147,483,648 to 2,147,483,647
- float: Approximately 1.4e-45 to 3.4e38
- byte: -128 to 127
- long: -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807
- Boolean: Accepts only true and false

Step 11: **To convert Fahrenheit to Celsius and vice versa**

- new DecimalFormat("0.00"): Creates a DecimalFormat object for formatting floating-point numbers with two decimal places.

- decimalFormat.format(double d): Formats a double value as a string with the specified formatting.
- decimalFormat.setRoundingMode(RoundingMode.DOWN): Sets the rounding mode for the DecimalFormat to round down (toward zero)
- Convert temperatures between Fahrenheit and Celsius (f - 32) * 5 / 9 and (9 * c / 5) + 32.

Step 12: **To find the velocity**

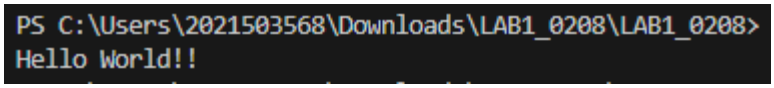
- in.nextDouble(): Reads a double value input from the user.
- v = u + a*t : Finding velocity of the program
- in.close() : Closes the Scanner object to release system resources

1.1) Write a java to print Hello world

CODE:

```
class HelloWorld3568{
    public static void main(String args[]){
        System.out.println("Hello World!!");
    }
}
```

OUTPUT:



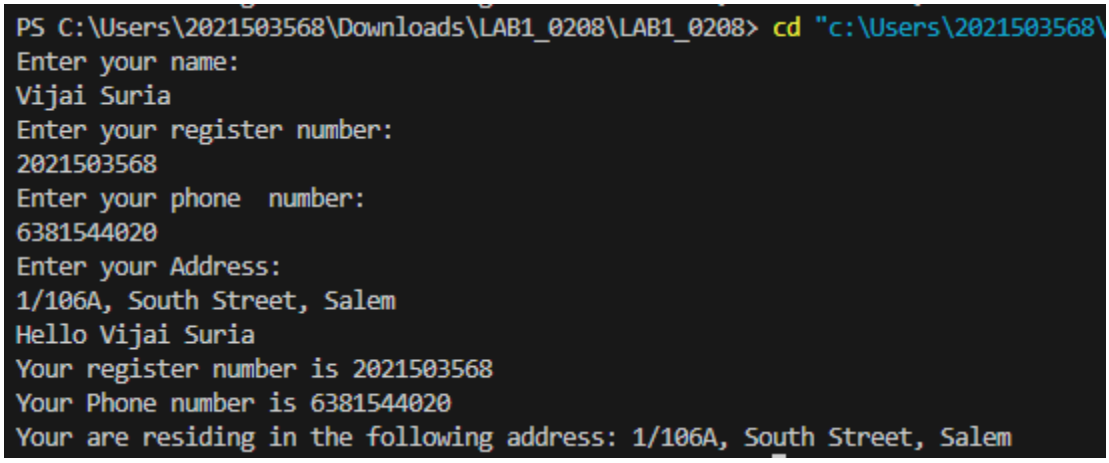
1.2) Write a Java program to get the user profile details and display them.

CODE:

```
import java.util.Scanner;
class Profile3568{
    public static void main(String args[]){
        Scanner in = new Scanner(System.in);

        System.out.println("Enter your name: ");
        String name = in.nextLine();
```

```
        System.out.println("Enter your register number: ");
        int reg = in.nextInt();
        System.out.println("Enter your phone  number: ");
        String phone = in.next();
        System.out.println("Enter your Address: ");
        String address = in.next();
        address += in.nextLine();

        System.out.println("Hello " + name );
        System.out.println("Your register number is " + reg );
        System.out.println("Your Phone number is " + phone);
        System.out.println("Your are residing in the following address: "+address);
        in.close();
    }
}
```

OUTPUT:



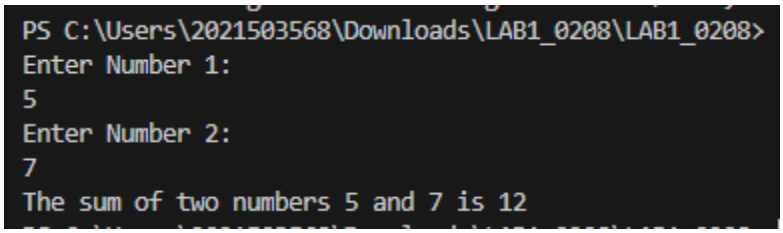
1.3) Write a Java program to add two given numbers from the user

CODE:

```
import java.util.Scanner;
class Add3568 {
    public static void main(String args[]){
        Scanner in = new Scanner(System.in);
        System.out.println("Enter Number 1: ");
```

```
        int a=in.nextInt();
        System.out.println("Enter Number 2: ");
        int b=in.nextInt();
        System.out.print("The sum of two numbers " + a + " and " + b + " is " +
(a+b));
        in.close();
    }
}
```

OUTPUT:

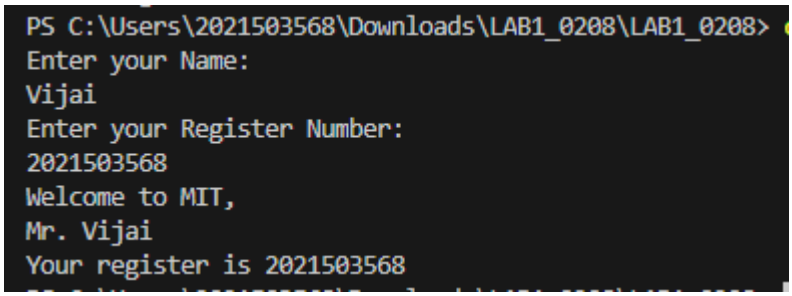


1.4) Write a Java program to display the welcome note to the user

CODE:

```
import java.util.Scanner;
class Welcome3568{
    public static void main(String args[]){
        String name;
        long reg;
        Scanner input = new Scanner(System.in);
        System.out.println("Enter your Name: ");
        name = input.nextLine();
        System.out.println("Enter your Register Number: ");
        reg = input.nextLong();
        System.out.println("Welcome to MIT, \nMr. " + name +"\nYour register is " +
reg);
        input.close();
    }
}
```

OUTPUT:



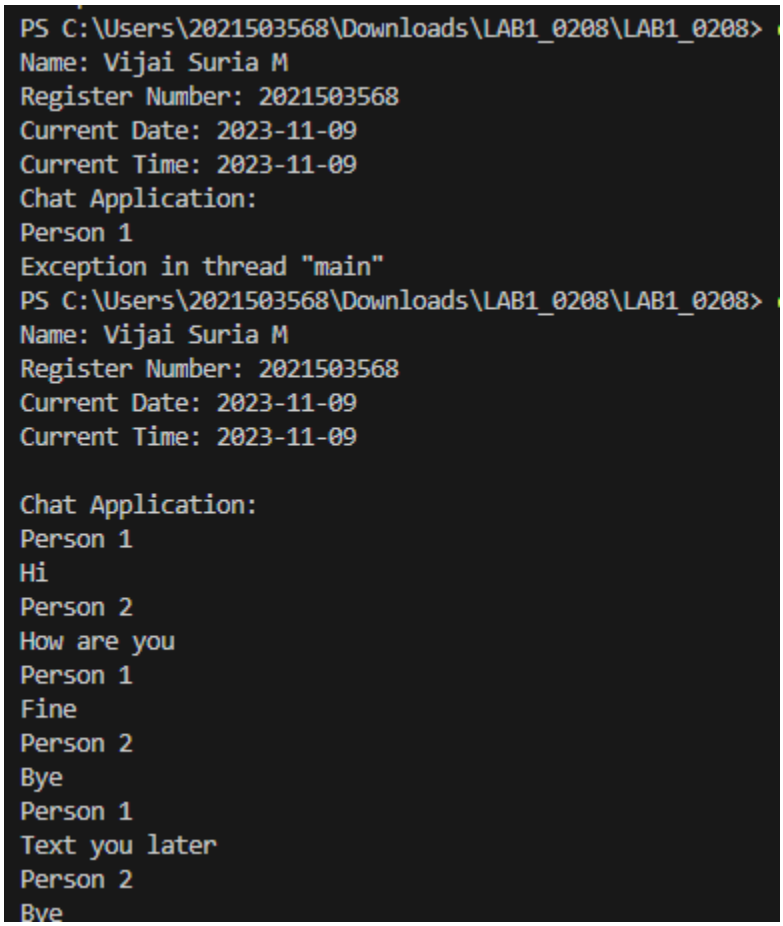
1.5) Write a Java Program to build the chat application where two user can chat.

CODE:

```
import java.util.*;
import java.time.LocalTime;
import java.time.LocalDate;
class Chat3568{
    public static void main(String args[]){
        System.out.println("Name: Vijai Suria M");
        System.out.println("Register Number: 2021503568");
        System.out.println("Current Date: " + LocalDate.now());
        System.out.println("Current Time: " + LocalDate.now());
        System.out.println("\nChat Application: ");
        String p1="", p2="";
        Scanner in = new Scanner(System.in);
        ArrayList<String> chats = new ArrayList<>();
        while(!p1.equals("bye") && !p2.equals("bye")){
            System.out.println("Person 1");
            p1=in.nextLine();
            chats.add(p1);
            System.out.println("Person 2");
            p2=in.next();
            p2+=in.nextLine();
            chats.add(p2);
        }
        System.out.println("The conversation is ");
        int i=1;
```

```
        for (String chat : chats) {
            System.out.println("Person" + i + ": " + chat);
            i= i==1?2:1;
        }
        in.close();
    }
}
```

OUTPUT:



1.6) Write a java program and compile the code @ command line to execute

CODE:

```
import java.time.LocalTime;
class HelloWorld3568 {
    public static void main(String args[]) {
        System.out.println(java.time.LocalDate.now());
        System.out.println(LocalTime.now());
        System.out.print("I am Vijai Suria M (2021503568), \n Hello, ");
        System.out.print(args[0]);
        System.out.println(" Good Morning!");
    }
}
```

Output:

```
D:\2021503568_JAVA\LAB2_0908>javac HelloWorld3568.java

D:\2021503568_JAVA\LAB2_0908>java HelloWorld3568 Yoga
2023-08-09
01:37:51.074277600
I am Vijai Suria M (2021503568),
 Hello, Yoga Good Morning!
```

1.7) Write the program to print current date and time

CODE:

```
import java.time.LocalTime;
public class Date3568 {
    public static void main(String args[]) {
        System.out.println("Current Date: " + java.time.LocalDate.now());
        System.out.println("Current Time: " + LocalTime.now());
        System.out.print("Name: Vijai Suria M \nRegister Number: (2021503568)");
    }
}
```

OUTPUT:

```
[Running] cd "d:\2021503568_JAVA\LAB2_0908\" && javac Date3568.java && java Date3568
Current Date: 2023-08-09
Current Time: 01:48:03.570836
Name: Vijai Suria M
Register Number: (2021503568)
[Done] exited with code=0 in 0.403 seconds
```

1.8) Find the maximum Compile time and Runtime error messages of simple one line output message

CODE:

```
import java.time.LocalTime;
class HelloWorld3568 {
    public static void main(String args[]) {
        System.out.println(java.time.LocalDate.now());
        System.out.println(LocalTime.now());
        System.out.print("I am Vijai Suria M (2021503568), \n Hello, ");
        System.out.print(args[0]);
        System.out.println(" Good Morning!");
    }
}
```

OUTPUT:

a) Delete any of the semicolons.

```
[Running] cd "d:\2021503568_JAVA\LAB2_0908\" && javac Error3568.java && java Error3568
Error3568.java:7: error: ';' expected
        System.out.print("Name: Vijai Suria M \nRegister Number: (2021503568)")
                                                                              ^
1 error

[Done] exited with code=1 in 0.269 seconds
```

b) Misspell the word public, static, void, main

```
[Running] cd "d:\2021503568_JAVA\LAB2_0908\" && javac Error3568.java && java Error3568
Error3568.java:3: error: class, interface, enum, or record expected
Public class Error3568 {
^
1 error

[Done] exited with code=1 in 0.265 seconds
```

c) omit the word public, static, void, main, arg

```
[Running] cd "d:\2021503568_JAVA\LAB2_0908\" && javac Error3568.java && java Error3568
Current Date: 2023-08-09
Current Time: 01:55:10.785300600
Name: Vijai Suria M
Register Number: (2021503568)
[Done] exited with code=0 in 0.429 seconds
```

d) Remove the quotation marks around string

```
[Running] cd "d:\2021503568_JAVA\LAB2_0908\" && javac Error3568.java && java Error3568
Error3568.java:7: error: unclosed string literal
        System.out.print("Name: Vijai Suria M \nRegister Number: (2021503568));
                         ^
1 error
```

e) change the main method argument data type

```
[Running] cd "d:\2021503568_JAVA\LAB2_0908\" && javac Error3568.java && java Error3568
Error: Main method not found in class Error3568, please define the main method as:
   public static void main(String[] args)
or a JavaFX application class must extend javafx.application.Application

[Done] exited with code=1 in 0.421 seconds
```

f) omit the argument

```
[Running] cd "d:\2021503568_JAVA\LAB2_0908\" && javac Error3568.java && java Error3568
Error: Main method not found in class Error3568, please define the main method as:
   public static void main(String[] args)
or a JavaFX application class must extend javafx.application.Application

[Done] exited with code=1 in 0.434 seconds
```

g) change the argument variable name

```
[Running] cd "d:\2021503568_JAVA\LAB2_0908\" && javac Error3568.java && java Error3568
Current Date: 2023-08-09
Current Time: 02:00:00.077779
Name: Vijai Suria M
Register Number: (2021503568)
[Done] exited with code=0 in 0.428 seconds
```

h) Remove the curly braces

```
[Running] cd "d:\2021503568_JAVA\LAB2_0908\" && javac Error3568.java && java Error3568
Error3568.java:3: error: '{' expected
public class Error3568
                     ^
Error3568.java:4: error: class, interface, enum, or record expected
    public static void main(String var[]) {
                ^
Error3568.java:6: error: class, interface, enum, or record expected
        System.out.println("Current Time: " + LocalTime.now());
        ^
Error3568.java:7: error: class, interface, enum, or record expected
        System.out.print("Name: Vijai Suria M \nRegister Number: (2021503568)");
        ^
Error3568.java:8: error: class, interface, enum, or record expected
    }
    ^
5 errors

[Done] exited with code=1 in 0.269 seconds
```

1.9) Copy the program and compile it. Find the error messages that the compiler finds out. Correct it out and repeat the process until the code runs.

CODE:

```
import java.time.*;
import java.util.Scanner;

public class Bug3568 {
    public static void main (String args[]) {
        System.out.println("Current Date: " + java.time.LocalDate.now());
        System.out.println("Current Time: " + LocalTime.now());
```

```
        System.out.println("Name: Vijai Suria M \nRegister Number:
(2021503568)");
        String name;
        Scanner in=new Scanner(System.in);
        System.out.println("Hello. Please type your name: ");
        name = in.next();
        System.out.println("Hello "+name);
        System.out.println ("Have a, nice day!");
        in.close();
    }
}
```

OUTPUT:

```
PS D:\2021503568_JAVA> cd "d:\2021503568_JAVA\LAB2_0908\" ; if ($?) { javac Bug3568.java } ;
Current Date: 2023-08-09
Current Time: 02:18:02.360450900
Name: Vijai Suria M
Register Number: (2021503568)
Hello. Please type your name:
Vijai
Hello Vijai
Have a, nice day!
PS D:\2021503568_JAVA\LAB2_0908>
```

1.10) Write a program to learn the different Java data types and their correct / incorrect values

CODE:

```
import java.time.*;
public class DataType3568 {
    public static void main(String args[]) {
        System.out.println("Current Date: " + LocalDate.now());
        System.out.println("Current Time: " + LocalTime.now());
        System.out.print("Name: Vijai Suria M \nRegister Number: (2021503568)");

        byte byteValue1 = 127; // Correct
        byte byteValue2 = 128; // Incorrect: Value exceeds the valid range (-128 to
```

```
127)
        long longValue1 = 9223372036854775807; // Incorrect, require L at the end
        long longValue2 = 9223372036854775807L; // Correct
        long longValue3 = 9223372036854775807; // Incorrect, require L at the end
        long longValue4 = 9223372036854775808L; // Incorrect: Value exceeds the
valid range

        // Floating-point data types
        float floatValue1 = 3.14f; // Correct
        float floatValue2 = 3.14; // Incorrect: Floating-point literals need 'f' or 'F' suffix
        // Boolean data type

        boolean booleanValue1 = true; // Correct
        boolean booleanValue2= 0; // Incorrect: Use 'true' or 'false' for boolean values
    }
}
```

OUTPUT:

```
PS D:\2021503568_JAVA\LAB2_0908> javac DataType3568.java
DataType3568.java:11: error: integer number too large
        long longValue1 = 9223372036854775807; // Incorrect, require L at the end
                          ^
DataType3568.java:13: error: integer number too large
        long longValue3 = 9223372036854775807; // Incorrect, require L at the end
                          ^
DataType3568.java:14: error: integer number too large
        long longValue4 = 9223372036854775808L; // Incorrect: Value exceeds the valid range
                          ^
3 errors
```

```
PS D:\2021503568_JAVA\LAB2_0908> javac DataType3568.java
DataType3568.java:10: error: incompatible types: possible lossy conversion from int to byte
       byte byteValue2 = 128; // Incorrect: Value exceeds the valid range (-128 to 127)
                         ^
DataType3568.java:18: error: incompatible types: possible lossy conversion from double to float
       float floatValue2 = 3.14; // Incorrect: Floating-point literals need 'f' or 'F' suffix
                           ^
DataType3568.java:22: error: incompatible types: int cannot be converted to boolean
       boolean booleanValue2= 0; // Incorrect: Use 'true' or 'false' for boolean values
                              ^
3 errors
```

1.11) Write a program that takes as input Fahrenheit temperature. It converts the input temperature to Celsius and prints out the converted temperature as shown in the example. The formula for conversion between the two is: C=5/9(F−32), Where C is the temperature in Celsius and F is the temperature in Fahrenheit

CODE:

```
import java.math.RoundingMode;
import java.text.DecimalFormat;
import java.time.*;
import java.util.Scanner;

public class F2C3568 {
   public static void main(String args[]) {
      Scanner in = new Scanner(System.in);
      float f, c;
      System.out.println("Current Date: " + LocalDate.now());
      System.out.println("Current Time: " + LocalTime.now());
      System.out.print("Name: Vijai Suria M \nRegister Number: (2021503568)");

      System.out.println("\n Enter your choice \n 1) Fahrenheit to Celsius \n 2)
Celsius to Fahrenheit ");
      int choice = in.nextInt();
      DecimalFormat decimalFormat = new DecimalFormat("0.00");
      switch (choice) {
         case 1:
            System.out.print("Temperature in Fahrenheit: ");
            f = in.nextFloat();
            c = (f - 32) * 5 / 9;
            System.out.println("Equivalent Temperature in Celsius: " +
decimalFormat.format(c));
            decimalFormat.setRoundingMode(RoundingMode.DOWN);
            break;
         case 2:
            System.out.print("Temperature in Celsius: ");
            c = in.nextFloat();
            f = (9 * c / 5) + 32;
            System.out.println("Equivalent Temperature in Fahrenheit: "+
                  decimalFormat.format(f));
```

```
                    decimalFormat.setRoundingMode(RoundingMode.DOWN);
                break;
            default:
                System.out.println("\n Please, Enter the valid choice.....");
        }
        in.close();
    }
}
```

OUTPUT:

```
PS D:\2021503568_JAVA\LAB2_0908> cd "d:\2021503568_JAVA\LAB2_0908\" ;
Current Date: 2023-08-09
Current Time: 15:18:14.530804900
Name: Vijai Suria M
Register Number: (2021503568)
 Enter your choice
 1) Fahrenheit to Celsius
 2) Celsius to Fahrenheit
1
Temperature in Fahrenheit: 0.555
Equivalent Temperature in Celsius: -17.47
```

```
PS D:\2021503568_JAVA\LAB2_0908> cd "d:\2021503568_JAVA\LAB2_0908\" ;
Current Date: 2023-08-09
Name: Vijai Suria M
Register Number: (2021503568)
 Enter your choice
 1) Fahrenheit to Celsius
 2) Celsius to Fahrenheit
1
Temperature in Fahrenheit: 212
Equivalent Temperature in Celsius: 100.00
```

```
Temperature in Fahrenheit: 212
Equivalent Temperature in Celsius: Exception in thread "main" java.util.IllegalFormatConversionException: f != java.lang.String
        at java.base/java.util.Formatter$FormatSpecifier.failConversion(Formatter.java:4522)
        at java.base/java.util.Formatter$FormatSpecifier.printFloat(Formatter.java:3056)
        at java.base/java.util.Formatter$FormatSpecifier.print(Formatter.java:3004)
        at java.base/java.util.Formatter.format(Formatter.java:2769)
        at java.base/java.io.PrintStream.format(PrintStream.java:1222)
        at java.base/java.io.PrintStream.printf(PrintStream.java:1118)
        at F2C3568.main(F2C3568.java:21)
PS D:\2021503568_JAVA\LAB2_0908>
```

1.12) Write a program that accepts the three numbers u, a, and t as input. Here, u denotes the starting speed, the acceleration, and t the amount of time. The program outputs the displacement covered (d) in time (t). The program prints the final velocity (v). v=u+at. Since velocity and acceleration are continuous vectors (in physics), u and a can have any real value. Only non-negative real values, or 0 t, can be assigned to time t, i.e., 0 ≤ t. Note: round your answer to up to two decimal places.

CODE:

```
import java.time.LocalTime;
import java.util.Scanner;

public class FinalVelocity3568 {
    public static void main(String args[]) {
        Scanner in = new Scanner(System.in);
        System.out.println("Current Date: " + java.time.LocalDate.now());
        System.out.println("Current Time: " + LocalTime.now());
        System.out.println("Name: Vijai Suria M \nRegister Number: (2021503568)");
        System.out.print("Enter the Starting Speed u=");
        double u = in.nextDouble();
        System.out.print("Enter the Acceleration a=");
        double a = in.nextDouble();
        System.out.print("Enter the Time t=");
        double t = in.nextDouble();
        while(t<0){
            System.out.print("(Invalid Time Value, Enter the Time t=");
            t = in.nextDouble();
        }
        double v = u + a*t;
        System.out.println("Value of Velocity v=" + v);
        in.close();
    }
}
```

OUTPUT:

```
PS D:\2021503568_JAVA\LAB2_0908> cd "d:\2021503568_JAVA\LAB2_0908\" ;
Current Date: 2023-08-09
Current Time: 15:34:39.078593900
Name: Vijai Suria M
Register Number: (2021503568)
Enter the Starting Speed u=20.0
Enter the Acceleration a=15
Enter the Time t=2
Value of Velocity v=50.0
```

```
PS D:\2021503568_JAVA\LAB2_0908> cd "d:\2021503568_JAVA\LAB2_0908\" ;
Current Date: 2023-08-09
Current Time: 15:35:12.795577300
Name: Vijai Suria M
Register Number: (2021503568)
Enter the Starting Speed u=20
Enter the Acceleration a=12
Enter the Time t=-2
(Invalid Time Value, Enter the Time t=-9
(Invalid Time Value, Enter the Time t=0
Value of Velocity v=20.0
```

Result: Thus, basic concepts in Java programming were explored successfully. And thus, outputs were verified.

Ex.No: 2 **Date: 09-08-23**

CONTROL STRUCTURES

Aim: To implement basic concepts of Control Structures in Java

Algorithm:

Step 1: **Custom Grade**

Methods used:

a) **System.out.println(String)** - Print a string to the standard output.
b) **LocalDate.now()** - To get the current date.
c) **Scanner in = new Scanner(System.in)** - To create a Scanner object to read user input from the standard input.
d) **in.nextInt()** - To read and return the next integer from the input.
e) **int[] arr = new int[n]** - To initialize an integer array to store marks for each subject.

Step 2: **Coin toss**

Methods used:

a) **Random rand = new Random()** - To create a random number generator.
b) **rand.nextInt(int)** - To generate a random integer within the specified range.
c) **in.next().charAt(int)** - To read the next token as a string and get the character at the specified index.

Step 3: **No of days in a month**

Methods used:

a) **isLeapYear(int) –** To check if the given year is leap year or not
b) **noOfDays(int) –** To get the no of days in a month

Step 4: **Verbalize numbers**

Methods used:

a) **convertToWords(int)** – To convert a number to its verbal representation, and it uses recursion to handle hundreds and thousands.

Step 5: **24-hour digital watch**

Methods used:

a) **isValidDay(String day)** - To check if the given day is valid.
b) **getNextDay(String day)** - To get the next day based on the input day.

2.1) Write a program to determine the custom grade of the marks given. If the marks, go beyond 100 or lower than 0 then state the input is invalid.

90 to 100: Grade O

80 TO 89: Grade A+

70 TO 79: Grade A

60 TO 69: Grade B+

50 to 59: Grade B

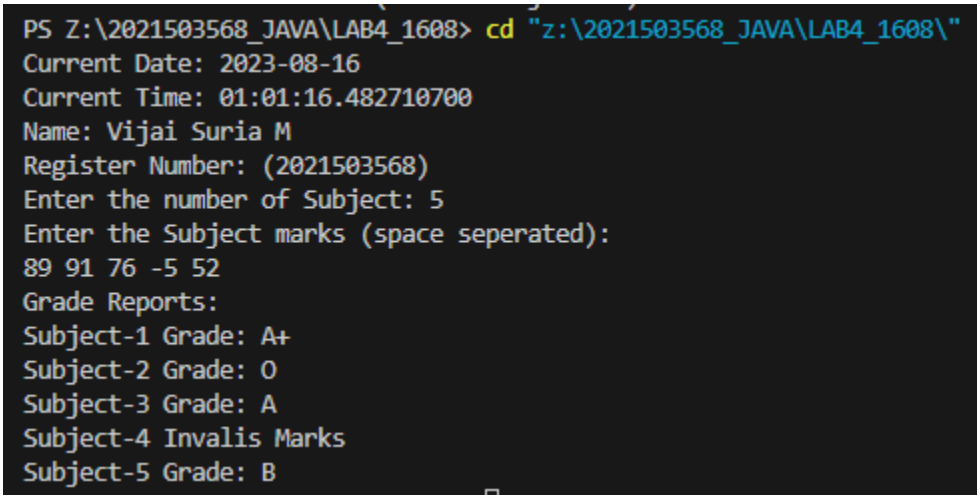
0 TO 49: Grade U Hint: use If statement.

CODE:

```
import java.time.LocalDate;
import java.time.LocalTime;
import java.util.Scanner;
public class Grade3568 {
    public static void main(String[] args) {
        System.out.println("Current Date: " + LocalDate.now());
        System.out.println("Current Time: " + LocalTime.now());
        System.out.println("Name: Vijai Suria M \nRegister Number: (2021503568)");
        Scanner in = new Scanner(System.in);
        System.out.print("Enter the number of Subject: ");
        int n = in.nextInt();
        int[] marks = new int[n];
        System.out.println("Enter the Subject marks (space seperated): ");
        for(int i=0;i<n;i++){
            marks[i] = in.nextInt();
        }
        System.out.println("Grade Reports:");
        for(int i=0;i<n;i++){
            if(marks[i]>=90 && marks[i]<=100)
                System.out.println("Subject-"+ (i+1)+" Grade: O");
            else if(marks[i]>=80 && marks[i]<=89)
                System.out.println("Subject-"+ (i+1)+" Grade: A+");
            else if(marks[i]>=70 && marks[i]<=79)
                System.out.println("Subject-"+ (i+1)+" Grade: A");
```

```
            else if(marks[i]>=60 && marks[i]<=69)
                System.out.println("Subject-"+ (i+1)+" Grade: B+");
            else if(marks[i]>=50 && marks[i]<=59)
                System.out.println("Subject-"+ (i+1)+" Grade: B");
            else if(marks[i]>=0 && marks[i]<=49)
                System.out.println("Subject-"+ (i+1)+" Grade: U");
            else
                System.out.println("Subject-"+ (i+1)+" Invalid Marks");
        }
        in.close();
    }
}
```

OUTPUT:



2.2) Write a program that simulates the coin toss as head or tail. Use Random number 0 0r 1 to determine the system input. The program should print the result as head if it one and tail if it is zero. Read the input from the user as a character 'h' or 't' or 'H' or 'T' and tell the user whether he or she has predicted the coin toss correctly. Declare the result as Won the toss! Or lose! (Hint: Use Random class to get the random input 0 to 1).

CODE:

```
import java.time.LocalDate;
import java.time.LocalTime;
import java.util.Random;
import java.util.Scanner;
public class Toss3568 {
   public static void main(String[] args) {
      System.out.println("Current Date: " + LocalDate.now());
      System.out.println("Current Time: " + LocalTime.now());
      System.out.println("Name: Vijai Suria M \nRegister Number: (2021503568)");
      Scanner in = new Scanner(System.in);
      Random random = new Random();
      int res = random.nextInt(2);
      System.out.print("Enter your choice (h/t) or (H/T): ");
      char ch=in.next().charAt(0);
      if(ch=='h' || ch=='H'){
         if(res==1)
            System.out.println("Won the toss!");
         else
            System.out.println("Loss the toss!");
      }
      else{
         if(res==0)
            System.out.println("Won the toss!");
         else
            System.out.println("Loss the toss!");
      }
      System.out.println(res);
      in.close();
   } }
```

OUTPUT:

```
PS Z:\2021503568_JAVA\LAB4_1608> cd "z:\2021503568_JAVA\LAB4_1608\"
Current Date: 2023-08-16
Current Time: 01:18:41.569796800
Name: Vijai Suria M
Register Number: (2021503568)
Enter your choice (h/t) or (H/T): t
Won the toss!
```

```
PS Z:\2021503568_JAVA\LAB4_1608> cd "z:\2021503568_JAVA\LAB4_1608\" ;
Current Date: 2023-08-16
Current Time: 01:18:36.785022600
Name: Vijai Suria M
Register Number: (2021503568)
Enter your choice (h/t) or (H/T): t
Loss the toss!
```

2.3) Write a program to output the number of days of the month(1 to 12) in the given year. The value of the month February is 28 or 20 based on the leap year. Give warning for invalid input that is of digit less than 4 or greater than 5. Hint: Use Switch statement.

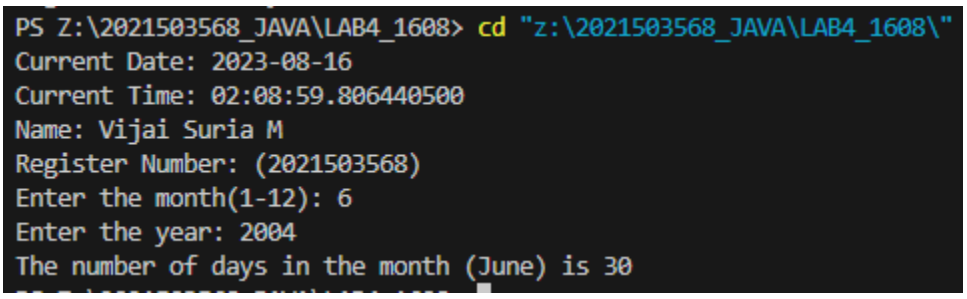
CODE:

```
import java.time.LocalDate;
import java.time.LocalTime;
import java.util.Scanner;
import java.time.Year;

public class Days3568 {
    public static void main(String[] args) {
        System.out.println("Current Date: " + LocalDate.now());
        System.out.println("Current Time: " + LocalTime.now());
        System.out.println("Name: Vijai Suria M \nRegister Number: (2021503568)");
        Scanner in = new Scanner(System.in);
        System.out.print("Enter the month(1-12): ");
        int month = in.nextInt();
        if(month<1 || month>12){
            System.out.println("Invalid month");
            System.exit(0);
```

```
        }
        System.out.print("Enter the year: ");
        int year = in.nextInt();
        if(year<=999 || year>9999){
           System.out.println("Invalid year");
           System.exit(0);
        }
        switch(month){
           case 1:
              System.out.println("The number of days in the month (January) is "+31);
              break;
           case 2:
              Year y = Year.of(year);
              if(y.isLeap())
                 System.out.println("The number of days in the month (Febrauary) is "+29);
              else
              System.out.println("The number of days in the month (Febrauary) is "+28);
              break;
           case 3:
              System.out.println("The number of days in the month (March) is "+31);
              break;
           case 4:
              System.out.println("The number of days in the month (April) is "+30);
              break;
           case 5:
              System.out.println("The number of days in the month (May) is "+31);
              break;
           case 6:
              System.out.println("The number of days in the month (June) is "+30);
              break;
           case 7:
              System.out.println("The number of days in the month (July) is "+31);
              break;
           case 8:
              System.out.println("The number of days in the month (August) is "+31);
              break;
           case 9:
              System.out.println("The number of days in the month (September) is "+30);
```

```
            break;
        case 10:
            System.out.println("The number of days in the month (October) is "+31);
            break;
        case 11:
            System.out.println("The number of days in the month (November) is "+30);
            break;
        case 12:
            System.out.println("The number of days in the month (December) is "+31);
            break;
        default:
            System.out.println("Invalid Month");
            break;
      }
      in.close();
   }
}
```

OUTPUT:



2.4) Write a program that verbalize the user inputs between 1 and 9999 number. For example : Input number : 852 => Output: Eight hundred and fifty two.

CODE:

```
import java.util.Scanner;

public class Verbalize3568 {
    private static final String[] units = {"", "One", "Two", "Three", "Four", "Five", "Six",
"Seven", "Eight", "Nine"};
    private static final String[] teens = {"Ten", "Eleven", "Twelve", "Thirteen", "Fourteen",
"Fifteen", "Sixteen", "Seventeen", "Eighteen", "Nineteen"};
    private static final String[] tens = {"", "", "Twenty", "Thirty", "Forty", "Fifty", "Sixty",
"Seventy", "Eighty", "Ninety"};
    private static final String[] thousands = {"", "Thousand", "Million", "Billion"};
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Input number: ");
        int number = scanner.nextInt();

        if (number < 1 || number > 9999) {
            System.out.println("Number out of range.");
        } else {
            String verbalized = verbalizeNumber(number);
            System.out.println(verbalized);
        }
        scanner.close();
    }

    private static String verbalizeNumber(int number) {
        if (number == 0) {
            return "Zero";
        }
        String verbalized = "";
        int thousandsIndex = 0;

        while (number > 0) {
            if (number % 1000 != 0) {
                verbalized = verbalizeChunk(number % 1000) + thousands[thousandsIndex] +
" " + verbalized;
```

```
            }
            number /= 1000;
            thousandsIndex++;
        }
        return verbalized.trim();
    }

    private static String verbalizeChunk(int number) {
        if (number == 0) {
            return "";
        }
        if (number < 10) {
            return units[number];
        } else if (number < 20) {
            return teens[number - 10];
        } else if (number < 100) {
            return tens[number / 10] + " " + units[number % 10];
        } else {
            return units[number / 100] + " Hundred " + verbalizeChunk(number % 100);
        }
    }
}
```

OUTPUT:

```
PS Z:\2021503568_JAVA\LAB4_1608> cd "z:\2021503568_JAVA\LAB4_1608\"
Current Date: 2023-08-16
Current Time: 02:00:19.260870200
Name: Vijai Suria M
Register Number: (2021503568)
Input number: 852
Eight Hundred Fifty Two
```

```
PS Z:\2021503568_JAVA\LAB4_1608> cd "z:\2021503568_JAVA\LAB4_1608\"
Current Date: 2023-08-16
Current Time: 02:00:04.972078200
Name: Vijai Suria M
Register Number: (2021503568)
Input number: 1234
OneThousand Two Hundred Thirty Four
```

2.5) Write a program to create 24-hour digital watch. Get the input from the user for the day(Mon, Tue, Wed, Thu, Fri, Sat, Sun) , hour (24-hour) , minute and seconds . The input of the user is valid then the digital clock should advance the input by one second and display the new day, hour, minute and second.

Sample input:

Input day : Mon

Input hour : 23

Input minute : 59

Input second : 59

Sample output:

Tue 0 0 0

Hint: Use for statement

CODE:

```
import java.time.LocalDate;
import java.time.LocalTime;
import java.util.Scanner;

public class Time3568 {
   public static String nextDay(String day)
   {
      if(day.equals("Sun"))
        return "Mon";
      if(day.equals("Mon"))
        return "Tue";
      if(day.equals("Tue"))
        return "Wed";
      if(day.equals("Wed"))
        return "Thu";
      if(day.equals("Thu"))
        return "Fri";
      if(day.equals("Fri"))
```

```
        return "Sat";

        return "Sun";

    }

    public static void main(String[] args) {
        System.out.println("Current Date: " + LocalDate.now());
        System.out.println("Current Time: " + LocalTime.now());
        System.out.println("Name: Vijai Suria M \nRegister Number: (2021503568)");

        String day;
        int hour,minute,sec;
        Scanner in = new Scanner(System.in);
        System.out.print("Enter day:");
        day=in.nextLine();
        System.out.print("Enter hour:");
        hour = in.nextInt();
        System.out.print("Enter minute:");
        minute = in.nextInt();
        System.out.print("Enter second:");
        sec = in.nextInt();

        if(sec+1 <=59)
            sec+=1;
        else{
            sec =0;
            if(minute+1 <=  59)
                minute++;
            else
            {
                minute =0;
                if(hour+1<=23)
                    hour++;
                else
                {
                    hour = 0;
                    day=nextDay(day);
```

```
            }
          }
        }
        System.out.println(day+" "+hour+" "+minute+" "+sec);
        in.close();
    }
}
```

OUTPUT:

```
PS Z:\2021503568_JAVA\LAB4_1608> cd "z:\2021503568_JAVA\LAB4_1608\" ;
Current Date: 2023-08-16
Current Time: 02:32:38.265161100
Name: Vijai Suria M
Register Number: (2021503568)
Enter day:Mon
Enter hour:23
Enter minute:59
Enter second:59
Tue 0 0 0
```

```
PS Z:\2021503568_JAVA\LAB4_1608> cd "z:\2021503568_JAVA\LAB4_1608\"
Current Date: 2023-08-16
Current Time: 02:39:39.046738
Name: Vijai Suria M
Register Number: (2021503568)
Enter day:Thu
Enter hour:12
Enter minute:58
Enter second:59
Thu 12 59 0
```

Result: Thus, basic concepts of Control Structures in java were implemented successfully. And thus, outputs were verified.

Ex. No: 3 **DATE: 30-08-23**

CONSTRUCTOR AND METHODS

Aim: To implement basic concepts of constructors and methods in Java

Algorithm:

Step 1: Circle class

Methods used:

a) **System.out.println(String)** - To print a string to the standard output.
b) **LocalDate.now()** - To get the current date.
c) **LocalTime.now()** - To get the current time.
d) **Math.PI** - To get the mathematical constant pi (π).
e) **Circle(double x, double y, double r)** - Constructor to initialize the properties of a circle with a specified center (x, y) and radius (r).
f) **display()** - To display the center and radius of a circle.
g) **calculateArea()** - To calculate the area of a circle.
h) **static String compareArea(Circle c1, Circle c2)** - To compare the areas of two circles (c1 and c2) and return a comparison result as a string.

Step 2: Usage of default constructor, parameterized constructor, this keyword and returning object

Methods used:

a) **Student()** - Default constructor that initializes a student with default values (id = 0, name = "Unknown").
b) **Student(int id, String name)** - Parameterized constructor that initializes a student with specified values for id and name.
c) **display()** - To display the student's ID and name.
d) **static Student createStudent(int id, String name)** - To return a new Student object with the specified id and name.
e) **void updateStudent(Student newStudent)** – To update the student's information with the values from another student object.

Step 3: Count the number of instances

Methods used:

a) **Counter()** - Constructor that increments the static instance count for each new instance and initializes the non-static instance count for each instance.

b) **displayCounts()** - To display the total number of instances created and the non-static instance count for a specific instance.

Step 4: Method overloading
Methods used:

a) **int sum(int x, int y)** - Method to sum two integers.
b) **int sum(int x, int y, int z)** - Method to sum three integers.
c) **double sum(double x, double y)** - Method to sum two doubles.
d) **public Data(double value)** - Constructor to initialize a "Data" object with a specified value.

Step 5: Difference of Instance variables, Instance methods, static variable, static methods
Methods used:

a) **Dummy(int instanceVar)** - Constructor to initialize the instance variable with a specified value.
b) **void instanceMethod()** - To display information about the instance variable and call the static method.
c) **static void staticMethod()** – To display information about the static variable.

Step 6: Immutable class Person
Methods used:

a) **Person(String name, int age)** - Constructor to initialize the name and age of the person.
b) **String getName()** – To get the name of the person.
c) **int getAge()** – To get the age of the person.

Step 7: Java class Clock
Methods used:

a) **setClock(int hours, int minutes, int seconds)** - To set the time values while checking for validity.
b) **getHours()** - To get the hours of the time.
c) **getMinutes()** - To get the minutes of the time.
d) **getSeconds()** - To get the seconds of the time.
e) **setHours(int hours)** - To set the hours while checking for validity.
f) **setMinutes(int minutes)** - To set the minutes while checking for validity.
g) **setSeconds(int seconds)** - To set the seconds while checking for validity.
h) **tick()** - To increment the time by one second.

i) **tickDown()** - To decrement the time by one second.
j) **addClock(Clock other)** - To add time from another Clock instance.
k) **subtractClock(Clock other)** - To subtract time from another Clock instance and return the difference as a new Clock.

3.1) Write a program to create a class circle with centre and the radius as instance variables.

Initialize and display its variables.

Modify the exercise to have a constructor in class circle to initialize its variables.

Modify the exercise to define the instance method calculateArea() to calculate the area and a static method compareArea() to compare the area of the circle and declare the result as smaller than or larger than or equal

CODE:

```
class Circle {
    double centerX;
    double centerY;
    double radius;

    public Circle(double centerX, double centerY, double radius) {
this.centerX = centerX;
this.centerY = centerY;
this.radius = radius;
    }
    public double calculateArea() {
        return Math.PI * radius * radius;
    }

    public static String compareArea(Circle circle1, Circle circle2) {
        double area1 = circle1.calculateArea();
        double area2 = circle2.calculateArea();
        if (area1 < area2) {
            return "Circle 1 has a smaller area than Circle 2.";
        } else if (area1 > area2) {
            return "Circle 1 has a larger area than Circle 2.";
```

```
        } else {
            return "Circle 1 and Circle 2 have equal areas.";
        }
    }
}

public class CircleArea3568 {
    public static void main(String[] args) {
        Circle circle1 = new Circle(0, 0, 5);
        Circle circle2 = new Circle(3, 4, 7);

        System.out.println("Circle 1 - Center: (" + circle1.centerX + ", " + circle1.centerY +
"), Radius: " + circle1.radius);
        System.out.println("Circle 2 - Center: (" + circle2.centerX + ", " + circle2.centerY +
"), Radius: " + circle2.radius);

        System.out.println("Area of Circle 1: " + circle1.calculateArea());
        System.out.println("Area of Circle 2: " + circle2.calculateArea());

        System.out.println(Circle.compareArea(circle1, circle2));
    }
}
```

OUTPUT:

```
PS Z:\2021503568_JAVA\LAB6_3008> cd "z:\2021503568_JAVA\LAB6_3008\" ;
Circle 1 - Center: (0.0, 0.0), Radius: 5.0
Circle 2 - Center: (3.0, 4.0), Radius: 7.0
Area of Circle 1: 78.53981633974483
Area of Circle 2: 153.93804002589985
Circle 1 has a smaller area than Circle 2.
```

3.2) Write a program to display the use of

- this keyword.
- Default constructor
- Parameterized constructor
- Pass Object as an argument
- Return object

CODE:

```
import java.time.LocalDate;
import java.time.LocalTime;
class Car {
   public String myCar;
Car(){
this.myCar="";
      System.out.println("car object initialized with default constructor " + myCar);
   }
Car(String myCar){
this.myCar=myCar;
      System.out.println("car object initialized with parameterised constructor " + myCar);
   }
}
public class Question2 {
   public static Car mergeObject(Car obj1, Car obj2){
      Car result = new Car();
result.myCar = obj1.myCar + " " + obj2.myCar;
      return result;
   }
   public static void main(String[] args) {
      System.out.println("Current Date: " + LocalDate.now());
      System.out.println("Current Time: " + LocalTime.now());
      System.out.println("Name: Vijai Suria M \nRegister Number: (2021503568)");
      Car obj1 = new Car("Audi");
      Car obj2 = new Car("BMW");
      Car result = mergeObject(obj1,obj2);
      System.out.println(result.myCar);
   }
}
```

OUTPUT:

```
PS Z:\2021503568_JAVA\LAB6_3008> cd "z:\2021503568_JAVA\LAB6_3008\" ;
Current Date: 2023-08-30
Current Time: 14:55:29.994621500
Name: Vijai Suria M
Register Number: (2021503568)
car object initialized with parameterised constructor Audi
car object initialized with parameterised constructor BMW
car object initialized with default constructor
Audi BMW
```

3.3) Write a program to count the number of instances created for the class using static variable and the non static variable not visible to all the instances.

CODE:

```
import java.time.LocalDate;
import java.time.LocalTime;

class InstanceCounter {
    private static int instanceCount = 0;
    private int nonStaticCount = 0;

    public InstanceCounter() {
        instanceCount++;
        nonStaticCount++;
    }

    public static int getInstanceCount() {
        return instanceCount;
    }

    public int getNonStaticCount() {
        return nonStaticCount;
    }
}
```

```
public class InstanceCounter3568 {
    public static void main(String[] args) {
        System.out.println("Current Date: " + LocalDate.now());
        System.out.println("Current Time: " + LocalTime.now());
        System.out.println("Name: Vijai Suria M \nRegister Number: (2021503568)");
        InstanceCounter obj1 = new InstanceCounter();
        InstanceCounter obj2 = new InstanceCounter();
        InstanceCounter obj3 = new InstanceCounter();

        System.out.println("Total instances created: " +
InstanceCounter.getInstanceCount());
        System.out.println("Non-static count for obj1: " + obj1.getNonStaticCount());
        System.out.println("Non-static count for obj2: " + obj2.getNonStaticCount());
        System.out.println("Non-static count for obj3: " + obj3.getNonStaticCount());
    }
}
```

OUTPUT:

```
PS Z:\2021503568_JAVA\LAB6_3008> cd "z:\2021503568_JAVA\LAB6_3008\"
Current Date: 2023-08-30
Current Time: 15:03:23.163135500
Name: Vijai Suria M
Register Number: (2021503568)
Total instances created: 3
Non-static count for obj1: 1
Non-static count for obj2: 1
Non-static count for obj3: 1
```

3.4) Write a program that implements method overloading(multiple methods in the same class can have the same name but different parameter lists) based on the following conditions

- By changing number of arguments
- By changing the data type of the arguments
- Passing object as parameter.

CODE:

```
import java.time.LocalDate;
import java.time.LocalTime;
class Car {
    public String myCar;
    Car(){
        this.myCar="null";
        System.out.println("car object initialized with default constructor ");
    }
    Car(String myCar){
        this.myCar=myCar;
        System.out.println("car object initialized with parameterised constructor ");
    }
    public String getMyCar(){
        return myCar;
    }
    public String getMyCar(String name){
        return "Your name: "+ name + "\n Your car name:" + myCar;
    }
    public String getMyCar(String name,int id){
        return "Your name and ID: "+ name + "\t" + id + "\n Your car name:" +
myCar;
    }
    public String getMyCar(int id){
        return "Your Id: "+ id + "\n Your car name:" + myCar;
    }
}
public class Overloading3568 {
    public static Car mergeObject(Car obj1, Car obj2){
        Car result = new Car();
        result.myCar = obj1.myCar + " " + obj2.myCar;
        return result;
    }
    public static void main(String[] args) {
        System.out.println("Current Date: " + LocalDate.now());
        System.out.println("Current Time: " + LocalTime.now());
        System.out.println("Name: Vijai Suria M \nRegister Number: (2021503568)");
        Car obj1 = new Car("Audi");
```

```
        Car obj2 = new Car("BMW");
        Car result = mergeObject(obj1,obj2);
        System.out.println(result.getMyCar("Vijai"));
        System.out.println(result.getMyCar("Vijai",2021503568));
        System.out.println(result.getMyCar(2021503568));
        System.out.println(result.getMyCar());
    }
}
```

OUTPUT:

- By changing number of arguments

```
public String getMyCar(){
    return myCar;
}
public String getMyCar(String name){
    return "Your name: "+ name + "\n Your car name:" + myCar;
}
public String getMyCar(String name,int id){
    return "Your name: "+ name + "\n Your car name:" + myCar;
}
public String getMyCar(int id){
    return "Your Id: "+ id + "\n Your car name:" + myCar;
}
```

- By changing the data type of the arguments

```
class Car {
    public String myCar;
    Car(){
        this.myCar="null";
        System.out.println(x:"car object initialized with default constructor ");
    }
    Car(String myCar){
        this.myCar=myCar;
        System.out.println(x:"car object initialized with parameterised constructor ");
    }
    public String getMyCar(){
        return myCar;
    }
    public String getMyCar(String name){
        return "Your name: "+ name + "\n Your car name:" + myCar;
    }
    public String getMyCar(int id){
        return "Your Id: "+ id + "\n Your car name:" + myCar;
    }
}
```

- Passing object as parameter.

```
public static Car mergeObject(Car obj1, Car obj2){
    Car result = new Car();
    result.myCar = obj1.myCar + " " + obj2.myCar;
    return result;
}
```

- Show that the method overloading is not possible by just changing the return type

```
public String getMyCar(String name){
    return "Your name: "+ name + "\n Your car name:" + myCar;
}
public String getMyCar(int id){
    return "Your Id: "+ id + "\n Your car name:" + myCar;
}
public int getMyCar(String name){
    return 1;
}
```

- Can we overload java main method?
  No, it is not possible.

OUTPUT:

```
Current Date: 2023-09-17
Current Time: 22:43:56.285508300
Name: Vijai Suria M
Register Number: (2021503568)
car object initialized with parameterised constructor
car object initialized with parameterised constructor
car object initialized with default constructor
Your name: Vijai
 Your car name:Audi BMW
Your name and ID: Vijai 2021503568
 Your car name:Audi BMW
Your Id: 2021503568
 Your car name:Audi BMW
Audi BMW
```

3.5) Write a program that show the differences of Instance variables, Instance methods, static variable, static methods

CODE:

```
import java.time.LocalDate;
import java.time.LocalTime;
class Example {
    int instanceVar;
    static int staticVar;
    void instanceMethod() {
        System.out.println("This is an instance method.");
        System.out.println("Instance variable value: " + instanceVar);
        System.out.println("Static variable value: " + staticVar);
    }
    static void staticMethod() {
        System.out.println("This is a static method.");
        // Instance variables cannot be accessed directly in a static method.
        // System.out.println("Instance variable value: " + instanceVar); // This will cause an
error.
        System.out.println("Static variable value: " + staticVar);
    }
}
public class Question5 {
    public static void main(String[] args) {
        System.out.println("Current Date: " + LocalDate.now());
        System.out.println("Current Time: " + LocalTime.now());
        System.out.println("Name: Vijai Suria M \nRegister Number: (2021503568)");
        Example.staticVar = 100;
        Example.staticMethod();
        Example obj1 = new Example();
        obj1.instanceVar = 42;
        obj1.instanceMethod();
        Example obj2 = new Example();
        obj2.instanceVar = 24;
        obj2.instanceMethod();
    }
}
```

OUTPUT:

```
PS Z:\2021503568_JAVA\LAB6_3008> cd "z:\2021503568_JAVA\LAB6_3008\"
Current Date: 2023-08-30
Current Time: 15:10:10.847719500
Name: Vijai Suria M
Register Number: (2021503568)
This is a static method.
Static variable value: 100
This is an instance method.
Instance variable value: 42
Static variable value: 100
This is an instance method.
Instance variable value: 24
Static variable value: 100
```

6) Write a program to create an immutable class Person (state cannot be changed)

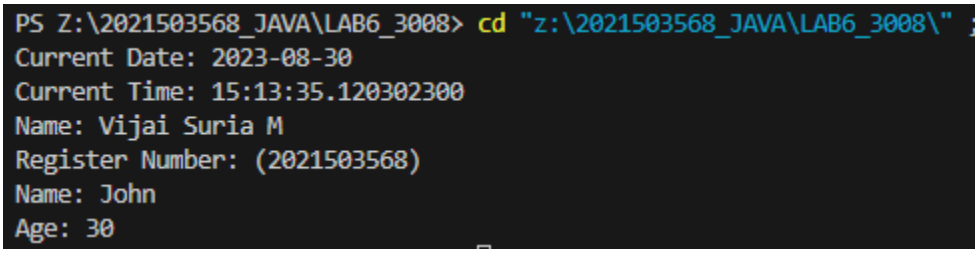
- Define private final fields of name and age
- Define a constructor to set the fields and a getter method to display the values.
- Show that the state cannot be changed and enhances robustness

CODE:

```
import java.time.LocalDate;
import java.time.LocalTime;

final class Person {
    private final String name;
    private final int age;

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

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

```
    public int getAge() {
        return age;
    }
}

public class Question6 {
    public static void main(String[] args) {
        System.out.println("Current Date: " + LocalDate.now());
        System.out.println("Current Time: " + LocalTime.now());
        System.out.println("Name: Vijai Suria M \nRegister Number: (2021503568)");
        Person person = new Person("John", 30);

        System.out.println("Name: " + person.getName());
        System.out.println("Age: " + person.getAge());
    }
}
```

OUTPUT:



3.7) Write a Java class Clock for dealing with the day time represented by hours, minutes, and seconds. Your class must have the following features:

Three instance variables for the hours (range 0-23), minutes(range 0-59),and seconds(range 0-59).

Three constructors:

- default(with no parameters passed; initialize the represented time to12:0:0)
- a constructor with three parameters: hours, minutes, and seconds.
- a constructor with one parameter: the value of time in seconds since midnight (it should be converted into the time value in hours, minutes,and seconds) Instance methods:
- A set-method method setClock() with one parameter seconds since midnight (to be converted into the time value in hours, minutes, and seconds as above).
- get-methods : getHours(), getMinutes(), getSeconds() with no parameters that return the corresponding values.
- set-methods : setHours(), setMinutes(), setSeconds() with one parameter each that setup the corresponding instance variables.
- method tick() with no parameters that increments the time stored in a Clock object by one second.
- method addClock() accepting an object of type Clock as a parameter.The method should add the time represented by the parameter class to the time represented in the current class.
- Add an instance method tickDown() which decrements the time stored in a Clock object by one second.
- Add an instance method subtractClock() that takes one Clock parameter and returns the difference between the time represented in the current

Clock object and the one represented by the Clock parameter. Difference of time should be returned as an clock object.

Write a separate class Clock Demo with a main() method. The program should:

- Instantiate a Clock object first Clock using one integer seconds since midnight obtained from the keyboard.
- Print both clock object

Create a reference thirdClock that should reference to object of difference of first Clock and second Clock by calling the method subtractClock()

CODE:

```
import java.time.LocalDate;
import java.time.LocalTime;
import java.util.Scanner;

class Clock{
    private int hour;
    private int minute;
    private int second;
    Clock(){
        hour = 12;
        minute = 0;
        second = 0;
    }
    Clock(int h, int m, int s){
        hour = h;
        minute = m;
        second = s;
    }
    Clock(int secondsFromMindnight){
        hour = secondsFromMindnight / 3600;
        minute = (secondsFromMindnight % 3600) / 60;
        second = (secondsFromMindnight % 3600) % 60;
    }
    public void setClock(int secondsFromMindnight){
        hour = secondsFromMindnight / 3600;
        minute = (secondsFromMindnight % 3600) / 60;
        second = (secondsFromMindnight % 3600) % 60;
    }
    public int getHours(){
        return hour;
    }
    public int getMinutes(){
        return minute;
    }
    public int getSeconds(){
        return second;
    }
```

```
public void setHours(int h){
    hour = h;
}
public void setMinutes(int m){
    minute = m;
}
public void setSeconds(int s){
    second = s;
}
public void tick(){
    second++;
    if(second == 60){
        second = 0;
        minute++;
        if(minute == 60){
            minute = 0;
            hour++;
            if(hour == 24){
                hour = 0;
            }
        }
    }
}
public void tickDown(){
    second--;
    if(second == -1){
        second = 59;
        minute--;
        if(minute == -1){
            minute = 59;
            hour--;
            if(hour == -1){
                hour = 23;
            }
        }
    }
}
public Clock addClock(Clock A){
```

```
        int s = this.second + A.getSeconds();
        int m = this.minute + A.getMinutes();
        int h = this.hour + A.getHours();
        while(s >= 60){
            s-=60;
            m++;
        }
        while(m >= 60){
            m-=60;
            h++;
        }
        if(h>=24)
            h=0;
        Clock res = new Clock(h,m,s);
        return res;
    }
    public Clock subtractClock(Clock A){
        int h = hour - A.getHours();
        int m = minute - A.getMinutes();
        int s = second - A.getSeconds();
        if(s < 0){
            s += 60;
            m--;
        }
        if(m < 0){
            m += 60;
            h--;
        }
        if(h < 0){
            h += 24;
        }
        Clock res = new Clock(h, m, s);
        return res;
    }
    public void show(){
        System.out.println(hour + ":" + minute + ":" + second);
    }
}
```

```
public class ClockDemo3568 {
    public static void main(String args[]){
        System.out.println("Current Date: " + LocalDate.now());
        System.out.println("Current Time: " + LocalTime.now());
        System.out.println("Name: Vijai Suria M \nRegister Number: (2021503568)");
        Scanner in = new Scanner(System.in);
        System.out.println("Enter the number of seconds from midnight: ");
        int secondsFromMindnight = in.nextInt();
        Clock b = new Clock(secondsFromMindnight);
        Clock a = new Clock();
        System.out.print("Clock 1:");
        a.show();
        System.out.print("Clock 2:");
        b.show();
        Clock c = a.subtractClock(b);
        System.out.print("Clock 3 => Clock_1 - Clock_2 = ");
        c.show();
        in.close();
    }
}
```

OUTPUT:

```
PS C:\Users\vijai\Documents\ASSIGNMENTS\JAVA-PROGRAMMING\LAB6_3008>
?) { javac ClockDemo3568.java } ; if ($?) { java ClockDemo3568 }
Current Date: 2023-09-12
Current Time: 12:26:27.235196600
Name: Vijai Suria M
Register Number: (2021503568)
Enter the number of seconds from midnight:
3600
Clock 1:12:0:0
Clock 2:1:0:0
Clock 3 => Clock_1 - Clock_2 = 11:0:0
```

Result: Thus, basic concepts of Constructors and Methods in java were implemented successfully. And thus, outputs were verified.

Ex.No: 4 **DATE: 09-08-23**

ARRAYS

Aim: To implement basic concepts of arrays in Java

Algorithm:

Step 1: To displaying the sorted array format. Include O(n^2) and O(n) complexity sorting algorithms and print the comparisons for each.

Methods used:

a) **System.out.println(String)** - Print a string to the standard output.

b) **LocalDate.now()** - To get the current date.

c) **LocalTime.now()** - To get the current time.

d) **Scanner in = new Scanner(System.in)** - To create a Scanner object to read user input from the standard input.

e) **in.nextInt()** - To read and return the next integer from the input.

f) **in.nextLine()** - To consume the newline character left in the input buffer.

g) **int[] arr = new int[n]** - To initialize an integer array to store marks for each subject.

h) bubbleSort(int[] arr) - Sorts an array in ascending order using the Bubble Sort algorithm and returns the number of comparisons.

i) selectionSort(int[] arr) - Sorts an array in ascending order using the Selection Sort algorithm and returns the number of comparisons.

j) printArray(int[] arr) - Prints the elements of an array in a formatted manner.

Step 2: To create a Java program with static methods to read and sort n random integers, returning comparisons. Also, generate and sort a character array from the random integers.

Methods used:

a) **RandomArray(int[] arr)**- Generates an array of random integers between 0 and 25..

b) **Random rand = new Random()** - To create a random number generator.

c) **rand.nextInt(int)** - To generate a random integer within the specified range.

d) **Scanner in = a Scanner(System.in)** - To create a Scanner object to read user input from the standard input.

e) **CharArray(int[] intArray)** - Converts an integer array to a character array, mapping integers to corresponding lowercase English letters.

f) OrderNSort(char[] arr) - Sorts a character array in ascending order using the Selection Sort algorithm and returns the number of comparisons.

g) OrderNSort(int[] arr) - Sorts an integer array in ascending order using the Selection Sort algorithm and returns the number of comparisons.

h) printArray(int[] arr) - Prints the elements of an integer array in a formatted manner with vertical bars.

i) printArray(char[] arr) - Prints the elements of a character array in a formatted manner with horizontal bars.

Step 3: To write Java program to read n random integers into arrays A and B, both of size n. Use a method to search and print the occurrences of each element from B in A.

Methods used:

a) **Random rand = new Random()** - To create a random number generator.

b) **rand.nextInt(int)** - To generate a random integer within the specified range.

c) **countOccurrences(int[] arr, int target)** - Counts the occurrences of a specified target integer in an array and returns the count.

Step 4: To calculates column-wise sums using a method, and sorts the arrays based on the column sums.

Methods used:

a) **Scanner scanner = new Scanner(System.in)** - To create a Scanner object to read user input from the standard input.

b) **scanner.nextInt()** - To read and return the next integer from the input.

c) **readMatrix(Scanner scanner, int[][] matrix)** - Reads input elements for a matrix from the user using a scanner.

d) addMatrices(int[][] matrix1, int[][] matrix2) - Adds two matrices and returns the result.

e) **sortMatrixByColumnSum(int[][] matrix)** - Sorts the matrix by the sum of each column in ascending order.

f) printMatrix(int[][] matrix) - Prints the elements of a matrix in a formatted manner.

4.1) Sort Write a program to read n integer in a 1D array and print the sorted array in the following format. Use static methods and find the number of comparisons for the sorting algorithm whose worst-case complexity is O(n^2) and O(n)

CODE:

```
import java.time.LocalDate;
import java.time.LocalTime;
import java.util.Scanner;

public class Sort3568 {
    public static void main(String[] args) {
        System.out.println("Current Date: " + LocalDate.now());
        System.out.println("Current Time: " + LocalTime.now());
        System.out.println("Name: Vijai Suria M \nRegister Number: (2021503568)");
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter no of elements: ");
        int n = scanner.nextInt();
        int[] arr = new int[n];

        System.out.println("Enter the array elements: ");
        for (int i = 0; i < n; i++) {
            arr[i] = scanner.nextInt();
        }
```

```
        int comparisonsBubbleSort = bubbleSort(arr.clone());
        int comparisonsSelectionSort = selectionSort(arr.clone());

        System.out.println("Number of comparisons for Bubble Sort: " +
comparisonsBubbleSort);
        System.out.println("Number of comparisons for Selection Sort: " +
comparisonsSelectionSort);
    }

    public static int bubbleSort(int[] arr) {
        int n = arr.length;
        int comparisons = 0;
        for (int i = 0; i < n - 1; i++) {
            for (int j = 0; j < n - i - 1; j++) {
                comparisons++;
                if (arr[j] > arr[j + 1]) {
                    int temp = arr[j];
                    arr[j] = arr[j + 1];
                    arr[j + 1] = temp;
                }
            }
        }

        System.out.println("Sorted array in ascending order(Bubble Sort):");
        printArray(arr);
        return comparisons;
    }

    public static int selectionSort(int[] arr) {
        int n = arr.length;
        int comparisons = 0;
        for (int i = 0; i < n - 1; i++) {
            int minIndex = i;
            for (int j = i + 1; j < n; j++) {
                comparisons++;
                if (arr[j] < arr[minIndex]) {
                    minIndex = j;
                }
```

```
            }
            int temp = arr[minIndex];
            arr[minIndex] = arr[i];
            arr[i] = temp;
        }
        System.out.println("Sorted array in ascending order(Selection Sort):");
        printArray(arr);
        return comparisons;
    }

    public static void printArray(int[] arr) {
        System.out.print("|");
        for (int i = 0; i < arr.length; i++) {
            System.out.printf(" %d |", arr[i]);
        }
        System.out.println();
    }
}
```

OUTPUT:

```
PS C:\Users\vijai\Documents\ASSIGNMENTS\JAVA-PROGRAMMING\LAB5_2308>
568.java } ; if ($?) { java Sort3568 }
Current Date: 2023-08-28
Current Time: 22:18:49.261587900
Name: Vijai Suria M
Register Number: (2021503568)
Enter no of elements: 6
Enter the array elements:
3 12 67 1 0 78
Sorted array in ascending order(Bubble Sort):
| 0 | 1 | 3 | 12 | 67 | 78 |
Sorted array in ascending order(Selection Sort):
| 0 | 1 | 3 | 12 | 67 | 78 |
Number of comparisons for Bubble Sort: 15
Number of comparisons for Selection Sort: 15
```

4.2) Sort random integer/character Write a program to read n random integer in a 1D array.

a) Apply method to sort the generated array content and return the number of comparisons done.

b) Apply another method to generate character array using the random integer and sort the array.

CODE:

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

public class SortRandom3568 {

    public static void main(String[] args) {
        System.out.println("Current Date: " + LocalDate.now());
        System.out.println("Current Time: " + LocalTime.now());
        System.out.println("Name: Vijai Suria M \nRegister Number: (2021503568)");
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter no of elements: ");
        int n = scanner.nextInt();
        int[] arr = new int[n];

        arr = RandomArray(arr);
        int comparisonsInt = OrderNSort(arr.clone());
        char[] charArray = CharArray(arr);
        int comparisonsChar = OrderNSort(charArray.clone());
        System.out.println("Number of comparisons for integer array sorting: " +
comparisonsInt);
        System.out.println("Number of comparisons for character array sorting: " +
comparisonsChar);
    }

    public static int[] RandomArray(int[] arr) {
        Random random = new Random();
```

```
        for (int i = 0; i < arr.length; i++) {
            arr[i] = random.nextInt(26);
        }
        return arr;
    }
    public static char[] CharArray(int[] intArray) {
        char[] charArray = new char[intArray.length];
        for (int i = 0; i < intArray.length; i++) {
            charArray[i] = (char) (intArray[i] + 'a');
        }
        return charArray;
    }
    public static int OrderNSort(int[] arr) {
        int comparisons = 0;
        for (int i = 0; i < arr.length - 1; i++) {
            for (int j = 0; j < arr.length - i - 1; j++) {
                comparisons++;
                if (arr[j] > arr[j + 1]) {
                    int temp = arr[j];
                    arr[j] = arr[j + 1];
                    arr[j + 1] = temp;
                }
            }
        }
        System.out.println("\nSorted array in ascending order(Selection Sort):");
        printArray(arr);
        return comparisons;
    }
    public static int OrderNSort(char[] arr) {
        int comparisons = 0;
        for (int i = 0; i < arr.length - 1; i++) {
            for (int j = 0; j < arr.length - i - 1; j++) {
                comparisons++;
                if (arr[j] > arr[j + 1]) {
                    char temp = arr[j];
                    arr[j] = arr[j + 1];
                    arr[j + 1] = temp;
                }
```

```
            }
        }
        System.out.println("\nSorted array in ascending order(Selection Sort):");
        printArray(arr);
        return comparisons;
    }
    public static void printArray(int[] arr) {
        System.out.print("|");
        for (int i = 0; i < arr.length; i++) {
            System.out.printf(" %d |", arr[i]);
        }
    }
    public static void printArray(char[] arr) {
        System.out.print("--------------------------\n");
        System.out.print("|");
        for (int i = 0; i < arr.length; i++) {
            System.out.printf(" %c |", arr[i]);
        }
        System.out.print("\n--------------------------\n");
    }
}
```

OUTPUT:

```
PS C:\Users\vijai\Documents\ASSIGNMENTS\JAVA-PROGRAMMING\LAB5_2308>
andom3568.java } ; if ($?) { java SortRandom3568 }
Current Date: 2023-08-28
Current Time: 22:29:17.281840300
Name: Vijai Suria M
Register Number: (2021503568)
Enter no of elements: 5

Sorted array in ascending order(Selection Sort):
| 1 | 2 | 6 | 17 | 25 |
Sorted array in ascending order(Selection Sort):
--------------------------
| b | c | g | r | z |
--------------------------
Number of comparisons for integer array sorting: 10
Number of comparisons for character array sorting: 10
```

4.3) Search element Occurrence Write a program to read n random integer in a 1D array of A and B of size n. Apply method to search the occurrence of element in B and print the number of B element occurrence in A.

CODE:

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

public class SortRandom3568 {

    public static void main(String[] args) {
        System.out.println("Current Date: " + LocalDate.now());
        System.out.println("Current Time: " + LocalTime.now());
        System.out.println("Name: Vijai Suria M \nRegister Number: (2021503568)");
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter no of elements: ");
        int n = scanner.nextInt();
        int[] arr = new int[n];

        arr = RandomArray(arr);
        int comparisonsInt = OrderNSort(arr.clone());

        char[] charArray = CharArray(arr);
        int comparisonsChar = OrderNSort(charArray.clone());

        System.out.println("Number of comparisons for integer array sorting: " +
comparisonsInt);
        System.out.println("Number of comparisons for character array sorting: " +
comparisonsChar);
    }

    public static int[] RandomArray(int[] arr) {
        Random random = new Random();
        for (int i = 0; i < arr.length; i++) {
            arr[i] = random.nextInt(26);
        }
```

```
    return arr;
}

public static char[] CharArray(int[] intArray) {
    char[] charArray = new char[intArray.length];
    for (int i = 0; i < intArray.length; i++) {
        charArray[i] = (char) (intArray[i] + 'a');
    }
    return charArray;
}

public static int OrderNSort(int[] arr) {
    int comparisons = 0;
    for (int i = 0; i < arr.length - 1; i++) {
        for (int j = 0; j < arr.length - i - 1; j++) {
            comparisons++;
            if (arr[j] > arr[j + 1]) {
                int temp = arr[j];
                arr[j] = arr[j + 1];
                arr[j + 1] = temp;
            }
        }
    }
    System.out.println("\nSorted array in ascending order(Selection Sort):");
    printArray(arr);
    return comparisons;
}

public static int OrderNSort(char[] arr) {
    int comparisons = 0;
    for (int i = 0; i < arr.length - 1; i++) {
        for (int j = 0; j < arr.length - i - 1; j++) {
            comparisons++;
            if (arr[j] > arr[j + 1]) {
                char temp = arr[j];
                arr[j] = arr[j + 1];
                arr[j + 1] = temp;
            }
```

```
            }
        }
        System.out.println("\nSorted array in ascending order(Selection Sort):");
        printArray(arr);
        return comparisons;
    }

    public static void printArray(int[] arr) {
        System.out.print("|");
        for (int i = 0; i < arr.length; i++) {
            System.out.printf(" %d |", arr[i]);
        }
    }

    public static void printArray(char[] arr) {
        System.out.print("--------------------------\n");
        System.out.print("|");
        for (int i = 0; i < arr.length; i++) {
            System.out.printf(" %c |", arr[i]);
        }
        System.out.print("\n--------------------------\n");
    }
}
```

OUTPUT:

```
PS C:\Users\vijai\Documents\ASSIGNMENTS\JAVA-PROGRAMMING\LAB5_2308>
ence3568.java } ; if ($?) { java Occurence3568 }
Current Date: 2023-08-28
Current Time: 22:34:04.628814100
Name: Vijai Suria M
Register Number: (2021503568)
Randomly generated value of n: 1
Random integers for array A:
49
Random integers for array B:
16
Number of occurrences of 16 in array A: 0
```

4.4) Sum of arrays Write a program to read two 2D array. Apply method to perform column major sum and sort the array based on the sum of columns.

CODE:

```
import java.time.LocalDate;
import java.time.LocalTime;
import java.util.Scanner;

public class sumArrays3524 {
    static void arraySum(int[][] a, int[][] s, int rows, int cols) {
        // Calculate the sum of corresponding elements in two arrays
        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < cols; j++) {
                a[i][j] += s[i][j];
            }
        }
        System.out.println("Sum of two arrays is:");
        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < cols; j++) {
                System.out.print(a[i][j] + " ");
            }
            System.out.println();
        }
    }
    static void colSumMax(int[][] x, int[][] y, int rows, int cols) {
        // Calculate the sum of arrays and sort based on column sum
        arraySum(x, y, rows, cols);
        int[] colsum = new int[cols];
        for (int i = 0; i < cols; i++) {
            for (int j = 0; j < rows; j++) {
                colsum[i] += x[j][i];
            }
        }
        int[] csindex = new int[cols];
        for (int j = 0; j < cols; j++) {
            csindex[j] = j;
        }
```

```
        for (int i = 0; i < cols - 1; i++) {
            for (int j = 0; j < cols - i - 1; j++) {
                if (colsum[j] > colsum[j + 1]) {
                    int temp = colsum[j];
                    colsum[j] = colsum[j + 1];
                    colsum[j + 1] = temp;


                    int temp1 = csindex[j];
                    csindex[j] = csindex[j + 1];
                    csindex[j + 1] = temp1;
                }
            }
        }
        System.out.println("The array after sorting based on column sum is:");
        // Display the sorted array
        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < cols; j++) {
                System.out.print(x[i][csindex[j]] + " ");
            }
            System.out.println();
        }
    }
    public static void main(String[] args) {
        System.out.println("Current Date: " + LocalDate.now());
        System.out.println("Current Time: " + LocalTime.now());
        System.out.println("Name: Vijai Suria M \nRegister Number:
(2021503568)");
        Scanner t = new Scanner(System.in);
        System.out.print("Enter the no. of rows: ");
        int r = t.nextInt();
        System.out.print("Enter the no. of columns: ");
        int c = t.nextInt();
        int[][] a1 = new int[r][c];
        int[][] a2 = new int[r][c];
        System.out.println("Enter the values of matrix 1:");
        // Input values for matrix 1
        for (int i = 0; i < r; i++) {
```

```
            for (int j = 0; j < c; j++) {
                a1[i][j] = t.nextInt();
            }
        }
        System.out.println("Enter the values of matrix 2:");
        // Input values for matrix 2
        for (int k = 0; k < r; k++) {
            for (int l = 0; l < c; l++) {
                a2[k][l] = t.nextInt();
            }
        }
        colSumMax(a1, a2, r, c);
    }
}
```

OUTPUT:

```
Name: Vijai Suria M
Register Number: (2021503568)
Enter the no. of rows: 2
Enter the no. of columns: 2
Enter the values of matrix 1:
1 2
3 4
Enter the values of matrix 2:
5 6
7 8
Sum of two arrays is:
6 8
10 12
The array after sorting based on column sum is:
6 8
10 12
```

Result: Thus, basic concepts of arrays in java were implemented successfully. And thus, outputs were verified.

Ex.No: 5 **DATE: 20-09-23**

STRING HANDLING

Aim: To implement basic concepts of String handling in Java

Algorithm:

Step 1: Perform string methods

Methods used:

a) **System.out.println(String)** - Print a string to the standard output.
b) **LocalDate.now()** - To get the current date.
c) **LocalTime.now()** - To get the current time.
d) **s1 == s2** - Checks if s1 and s2 are the same object in memory.
e) **s1.equals(s2)** - Compares the content of s1 and s2 to check if they are equal.
f) **s1.compareTo(s2)** - Compares two strings lexicographically and returns an integer based on their order.
g) **s1.charAt(index)** - Returns the character at the specified index in the string.
h) **s1.indexOf(char)** - Returns the first occurrence index of the specified character in the string.
i) **s1.lastIndexOf(char)** - Returns the last occurrence index of the specified character in the string.
j) **s1.length()** - Returns the number of characters in the string.
k) **s1.substring(start,end)** - Extracts a substring from s1 starting at start index and ending before end index.
l) **s1.startsWith(string)** - Checks if s1 starts with the specified string.
m) **s1.endsWith(string)** - Checks if s1 ends with the specified string.
n) **s1.toLower()** - Converts s1 to lowercase.
o) **s1.toUpper()** - Converts s1 to uppercase.
p) **s1.replace(char,char)** - Replaces all occurrences of a character with another character in the string.
q) **s1.replaceAll(char,char)** - Replaces all occurrences of a character with another character in the string.
r) **s1.replaceFirst(char,char)** - Replaces the first occurrence of a character with another character in the string.
s) **s1.toCharArray(char)** - Converts the string to a character array.

Step 2: Show that String is immutable

Methods used:

a) **referenceCheck():** Method to check if two objects have the same reference or not.
b) **displayStudentInfo():** Method to display the student's name and registration number.

Step 3: Reverse the Strinng
Methods used:

a) **ReverseString** - class with a constructor to initialize the str variable.
b) **reverseString()** - method takes user input, reverses the input string, and prints the reversed string.

Step 4: Count the number of occurrence of the each letter in the given string.
Methods used:

a) **LetterOccurrences** - class with a constructor to initialize the inputString variable.
b) **countLetterOccurrences()** - method takes user input, counts the occurrences of each letter in the input string, and prints the results.

Step 5: Count the number of words in the given string
Methods used:

a) **WordCounter** - class with a constructor to initialize the inputString variable.
b) **countWord()** - To get the current time.

5.1.1) Write a java program to perform string methods by considering the given string inputs String s1="Welcome to Java"; String s2=s1; String s3=new String("Welcome to Java"); String s4=s1.intern();

CODE:

```
import java.util.*;
import java.time.LocalDate;
import java.time.LocalTime;
public class String3568{
   public static void main(String[] args){
     System.out.println("Current Date: " + LocalDate.now());
     System.out.println("Current Time: " + LocalTime.now());
     System.out.println("Name: Vijai Suria M \nRegister Number: (2021503568)");
     System.out.println("\n");
     String s1 = "Welcome to java";
```

```
      String s2 = s1;
      String s3 = new String("Welcome to java");
      String s4 = s1.intern();
     if(s1 == s2)
       System.out.println("1.String s1 and s2 are equal  checked using s1 == s2 ");
      else  System.out.println("1.String s1 and s2 are not equal checked using s1 == s2");
     if(s2 == s3)
       System.out.println("2.String s2 and s3 are equal checked using s2 == s3");
      else  System.out.println("2.String s2 and s3 are not equal checked using s2 == s3");
       if(s1.equals(s2))
       System.out.println("3.String s1 and s2 are equal checked using s1.equals(s2) ");
      else  System.out.println("3.String s1 and s2 are not equal checked using s1.equals(s2) ");
      if(s2.equals(s3))
       System.out.println("4.String s2 and s3 are equal checked using s2.equals(s3)");
      else  System.out.println("4.String s2 and s3 are not equal checked using s2.equals(s3) ");
       if(s1.compareTo(s2) == 0)
       System.out.println("5.String s1 and s2 are equal checked using s1.compareTo(s2) ");
      else  System.out.println("5.String s1 and s2 are not equal checked using
s1.compareTo(s2) ");
      if(s2.compareTo(s3) == 0)
       System.out.println("6.String s2 and s3 are equal checked using s2.compareTo(s3)");
      else  System.out.println("6.String s2 and s3 are not equal checked using
s2.compareTo(s3) ");
        if(s1 == s4)
       System.out.println("7.String s1 and s2 are equal  checked using s1 == s4 ");
      else  System.out.println("7.String s1 and s2 are not equal checked using s1 == s4");
      System.out.println("8. s1.charAt(0) : "+s1.charAt(0));
      System.out.println("9. s1.indexOf('j') : "+s1.indexOf('j'));
      System.out.println("10. s1.indexOf(\"to\") : "+s1.indexOf("to"));
      System.out.println("11. s1.lastIndexOf('a') : "+s1.lastIndexOf('a'));
      System.out.println("12. s1.lastIndexOf(\"o\",15) : "+s1.lastIndexOf('a'));
      System.out.println("13. s1.length() : " + s1.length());
      System.out.println("14. s1.substring(5) : " + s1.substring(5));
      System.out.println("15. s1.substring(5,11) : " + s1.substring(5,11));
      System.out.println("16. s1.startsWith(\"Wel\") : " + s1.startsWith("Wel"));
      System.out.println("17. s1.endsWith(\"java\") : " + s1.startsWith("java"));
       System.out.println("18.  s1.toLowerCase() : " +  s1.toLowerCase());
       System.out.println("19.  s1.toUpperCase() : " +  s1.toUpperCase());
```

```
        String s7 = " Welcome ";
        System.out.println("20. "+ s7+".trim()"+  " : " +s7.trim());
        System.out.println("21. s1.replace('o','T') : " + s1.replace('o','T') );
        System.out.println("22. s1.replaceAll(\"o\",\"T\") : " + s1.replace("o","T") );
        System.out.println("23. s1.replaceFirst(\"o\",\"T\") : " + s1.replaceFirst("o","T") );
        System.out.println("24. s1.toCharArray() : " );
        char ch[] = s1.toCharArray() ;
        for(char c : ch)
         System.out.print(c);
    }
}
```

OUTPUT:

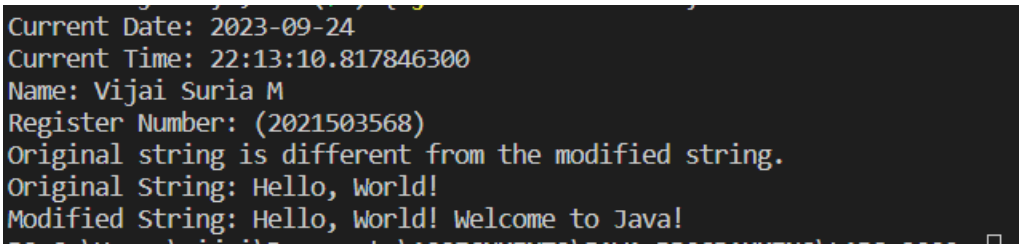
```
Current Date: 2023-09-24
Current Time: 22:09:40.228307600
Name: Vijai Suria M
Register Number: (2021503568)


1.String s1 and s2 are equal  checked using s1 == s2
2.String s2 and s3 are not equal checked using s2 == s3
3.String s1 and s2 are equal checked using s1.equals(s2)
4.String s2 and s3 are equal checked using s2.equals(s3)
5.String s1 and s2 are equal checked using s1.compareTo(s2)
6.String s2 and s3 are equal checked using s2.compareTo(s3)
7.String s1 and s2 are equal  checked using s1 == s4
8. s1.charAt(0) : W
9. s1.indexOf('j') : 11
10. s1.indexOf("to") : 8
11. s1.lastIndexOf('a') : 14
12. s1.lastIndexOf("o",15) : 14
13. s1.length() : 15
14. s1.substring(5) : me to java
15. s1.substring(5,11) : me to
16. s1.startsWith("Wel") : true
17. s1.endsWith("java") : false
18.  s1.toLowerCase() : welcome to java
19.  s1.toUpperCase() : WELCOME TO JAVA
20.  Welcome .trim() : Welcome
21. s1.replace('o','T') : WelcTme tT java
22. s1.replaceAll("o","T") : WelcTme tT java
23. s1.replaceFirst("o","T") : WelcTme to java
24. s1.toCharArray() :
```

5.1.2) Write a program to show that String is immutable in java.

CODE:

```
import java.time.LocalDate;
import java.time.LocalTime;
public class Immutable3568 {
    public static void main(String[] args) {
        System.out.println("Current Date: " + LocalDate.now());
        System.out.println("Current Time: " + LocalTime.now());
        System.out.println("Name: Vijai Suria M \nRegister Number:
(2021503568)");
        String originalString = "Hello, World!";
        String modifiedString = originalString.concat(" Welcome to Java!");
        if (originalString == modifiedString) {
            System.out.println("Original string is the same as the modified string.");
        } else {
            System.out.println("Original string is different from the modified
string.");
        }
        // Print the original and modified strings
        System.out.println("Original String: " + originalString);
        System.out.println("Modified String: " + modifiedString);
    }
}
```

OUTPUT:



REASON:

We attempt to modify the string by concatenating " Welcome to Java!" to it using the **concat** method. However, instead of modifying the **originalString**, a new string is created, and **modifiedString** references the new string.

5.2) Write a java program to read the string and displays the reverse of the string

CODE:

```
import java.time.LocalDate;
import java.time.LocalTime;
import java.util.Scanner;;
public class StringReverse3568 {
    public static void main(String args[]){
        System.out.println("Current Date: " + LocalDate.now());
        System.out.println("Current Time: " + LocalTime.now());
        System.out.println("Name: Vijai Suria M \nRegister Number:
(2021503568)");
        Scanner sc = new Scanner(System.in);
        String s;
        System.out.print("Enter a string: ");
        s = sc.nextLine();
        char[] charArray = s.toCharArray();
        int length = charArray.length;
        for (int i = 0; i < length / 2; i++) {
            char temp = charArray[i];
            charArray[i] = charArray[length - 1 - i];
            charArray[length - 1 - i] = temp;
        }
        String reversedString = new String(charArray);
        System.out.println("Reversed String: " + reversedString);
        sc.close();
    }
}
```

OUTPUT:

```
Current Date: 2023-09-24
Current Time: 22:14:50.724950100
Name: Vijai Suria M
Register Number: (2021503568)
Enter a string: Vijaisuria
Reversed String: airusiajiV
```

5.3) Write a java program to count the number of occurrence of the each letter in the given string

CODE:

```
import java.util.Scanner;
public class LettersCount3568 {
    public static void main(String args[]){
        //Write a java program to count the occurence of each letters in a string
        System.out.println("Current Date: " + java.time.LocalDate.now());
        System.out.println("Current Time: " + java.time.LocalTime.now());
        System.out.println("Name: Vijai Suria M \nRegister Number:
(2021503568)");
        String s;
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a string: ");
        s = sc.nextLine();
        int[] count = new int[26];
        for(int i = 0; i < s.length(); i++){
            if(s.charAt(i) >= 'a' && s.charAt(i) <= 'z'){
                count[s.charAt(i) - 'a']++;
            }
        }
        for(int i = 0; i < 26; i++){
            if(count[i] != 0){
                System.out.println((char)(i + 'a') + " occurs " + count[i] + " times");
            }
        }
        sc.close();
    } }
```

Output:

```
Current Date: 2023-09-24
Current Time: 22:17:44.477670500
Name: Vijai Suria M
Register Number: (2021503568)
Enter a string: Hello
e occurs 1 times
l occurs 2 times
o occurs 1 times
```

5.4) Write a Java program to count the number of words in the given string

CODE:

```
import java.util.Scanner;
public class WordsCount3568 {
    public static void main(String args[]){
        // Write a java program to count the number of words in a string
        System.out.println("Current Date: " + java.time.LocalDate.now());
        System.out.println("Current Time: " + java.time.LocalTime.now());
        System.out.println("Name: Vijai Suria M \nRegister Number:
(2021503568)");
        String s;
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a string: ");
        s = sc.nextLine();
        int count = 1;
        for(int i = 0; i < s.length(); i++){
            if(s.charAt(i) == ' '){
                count++;
            }
        }
        System.out.println("Number of words in the string: " + count);
    }
}
```

OUTPUT:

```
Current Date: 2023-09-24
Current Time: 22:18:47.844729600
Name: Vijai Suria M
Register Number: (2021503568)
Enter a string: Good Morning, Vijai
Number of words in the string: 3
```

5.5.1) Write a java program to check the given string is palindrome or not (Example:Race car)

CODE:

```
import java.time.LocalDate;
```

```
import java.time.LocalTime;
import java.util.Scanner;
public class Palindrome3568{
    public static boolean isPalindrome(String s){
        if(s.length() == 0 || s.length() == 1){
            return true;
        }
        if(s.charAt(0) == s.charAt(s.length()-1)){
            return isPalindrome(s.substring(1, s.length()-1));
        }
        return false;
    }
    public static void main(String[] args){
        System.out.println("Current Date: " + LocalDate.now());
        System.out.println("Current Time: " + LocalTime.now());
        System.out.println("Name: Vijai Suria M \nRegister Number: (2021503568)");
        Scanner sc = new Scanner(System.in);
        String s;
        System.out.print("Enter a string: ");
        s = sc.nextLine();
        System.out.println(isPalindrome(s));
        sc.close();
    }
}
```

OUTPUT:

```
Current Date: 2023-09-24
Current Time: 22:20:49.935699600
Name: Vijai Suria M
Register Number: (2021503568)
Enter a string: madam
true
```

```
Current Date: 2023-09-24
Current Time: 22:21:29.539266200
Name: Vijai Suria M
Register Number: (2021503568)
Enter a string: Vijai
false
```

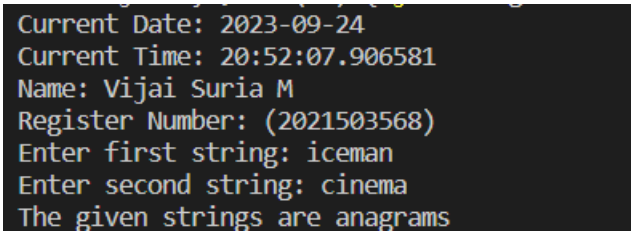
5.2) Write a java program to check the given string is anagram or not (Example Iceman vs Cinema)

CODE:

```
import java.time.LocalDate;
import java.time.LocalTime;
import java.util.Scanner;
import java.util.Arrays;
public class Anagram3568 {
    public static boolean areAnagrams(String str1, String str2) {
        // Remove all whitespace and convert to lowercase
        str1 = str1.replaceAll("\\s", "").toLowerCase();
        str2 = str2.replaceAll("\\s", "").toLowerCase();

        // Check if the lengths are different
        if (str1.length() != str2.length()) {
            return false;
        }

        // Convert the strings to char arrays and sort them
        char[] charArray1 = str1.toCharArray();
        char[] charArray2 = str2.toCharArray();
        Arrays.sort(charArray1);
        Arrays.sort(charArray2);

        // Compare the sorted char arrays
        return Arrays.equals(charArray1, charArray2);
    }
    public static void main(String args[]){
        //Write a java program to check the given string is anagram or not
        System.out.println("Current Date: " + LocalDate.now());
        System.out.println("Current Time: " + LocalTime.now());
        System.out.println("Name: Vijai Suria M \nRegister Number: (2021503568)");
        Scanner sc = new Scanner(System.in);
        String s1, s2;
        System.out.print("Enter first string: ");
        s1 = sc.nextLine();
        System.out.print("Enter second string: ");
        s2 = sc.nextLine();
        if(areAnagrams(s1, s2)){
```

```
            System.out.println("The given strings are anagrams");
        }else{
            System.out.println("The given strings are not anagrams");
        }
        sc.close();
    }
}
```

OUTPUT:



6) Write a java program that read a two string of the given format and compares the string

CODE:

```
import java.util.Scanner;

public class StringCompare3568 {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.println("Enter the first string: ");
        String str1 = scanner.nextLine();
        System.out.println("Enter the second string: ");
        String str2 = scanner.nextLine();

        String[] parts1 = str1.split("\\.");
        String[] parts2 = str2.split("\\.");

        if (parts1.length != 3 || parts2.length != 3) {
            System.out.println("Invalid input format. Please use dd.mm.ss format.");
            return;
```

```
        }

        int day1 = Integer.parseInt(parts1[0]);
        int month1 = Integer.parseInt(parts1[1]);
        int second1 = Integer.parseInt(parts1[2]);

        int day2 = Integer.parseInt(parts2[0]);
        int month2 = Integer.parseInt(parts2[1]);
        int second2 = Integer.parseInt(parts2[2]);

        if (day1 > day2) {
            System.out.println(str1 + " is greater than " + str2 + " as " + day1 + " > " +
day2);
        } else if (day1 < day2) {
            System.out.println(str2 + " is greater than " + str1 + " as " + day2 + " > " +
day1);
        } else {
            if (month1 > month2) {
                System.out.println(str1 + " is greater than " + str2 + " as " + month1 + " > "
+ month2);
            } else if (month1 < month2) {
                System.out.println(str2 + " is greater than " + str1 + " as " + month2 + " > "
+ month1);
            } else {
                if (second1 > second2) {
                    System.out.println(str1 + " is greater than " + str2 + " as " + second1 + "
> " + second2);
                } else if (second1 < second2) {
                    System.out.println(str2 + " is greater than " + str1 + " as " + second2 + "
> " + second1);
                } else {
                    System.out.println(str1 + " is equal to " + str2);
                }
            }
        }

        scanner.close();
    }
}
```

OUTPUT:

```
Current Date: 2023-09-24
Current Time: 22:26:41.397290800
Name: Vijai Suria M
Register Number: (2021503568)
Enter the first string:
15.10.10
Enter the second string:
14.20.10
15.10.10 is greater than 14.20.10 as 15 > 14
```

```
Current Date: 2023-09-24
Current Time: 22:27:11.319744100
Name: Vijai Suria M
Register Number: (2021503568)
Enter the first string:
14.12.10
Enter the second string:
14.10.50
14.12.10 is greater than 14.10.50 as 12 > 10
```

5.7) Write a java program using String methods to compare the email is valid is invalid and returns the username and domain name

i) Valid Username: numbers[0-7], alphabets[a-z][A-Z], underscore, dot, hypen and plus characters

ii) Presence of @ symbol

iii) Presence of domainname.com or .in or .edu

Hint use givenstring.split("@") to find specific user(case-insensitive:jc_vp) and specific domain

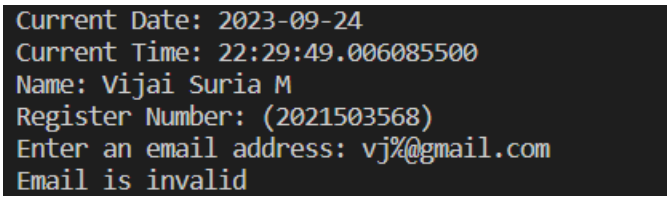
(case-insensitive: gmail.com) for example jc_vp@gmail.com.

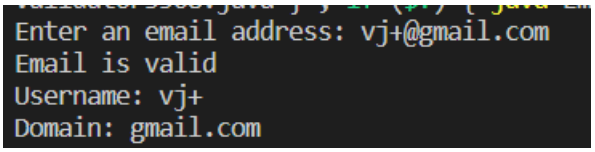
CODE:

```
import java.util.regex.Matcher;
import java.util.regex.Pattern;
import java.time.LocalDate;
import java.time.LocalTime;
import java.util.Scanner;
```

```
public class EmailValidator3568 {
    public static void main(String[] args) {
        System.out.println("Current Date: " + LocalDate.now());
        System.out.println("Current Time: " + LocalTime.now());
        System.out.println("Name: Vijai Suria M \nRegister Number:
(2021503568)");
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter an email address: ");
        String email =sc.nextLine();

        // Regular expression pattern for email validation
        String emailPattern = "^[a-zA-Z0-9_+.-]+@[a-zA-Z0-9.-
]+\\.(com|in|edu)$";

        // Create a Pattern object
        Pattern pattern = Pattern.compile(emailPattern);

        // Match the input email against the pattern
        Matcher matcher = pattern.matcher(email);
        if (matcher.matches()) {
            System.out.println("Email is valid");

            // Split the email address using "@" symbol
            String[] parts = email.split("@");

            // Extract and print the username (case-insensitive)
            String username = parts[0];
            System.out.println("Username: " + username.toLowerCase());

            // Extract and print the domain name (case-insensitive)
            String domain = parts[1];
            System.out.println("Domain: " + domain.toLowerCase());
        } else {
            System.out.println("Email is invalid");
        }
        sc.close();
    }
}
```

OUTPUT:





5.8. Write a java program to create a dictionary using 2D string array for any 10 programming languages.

Write a method that return the definition for the input of PL name.

Java- pure object oriented programming language by James Gosling

C++ - Object oriented programming language by Stroustrup

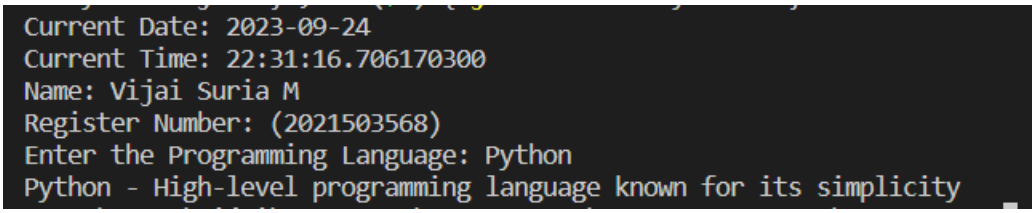
CODE:

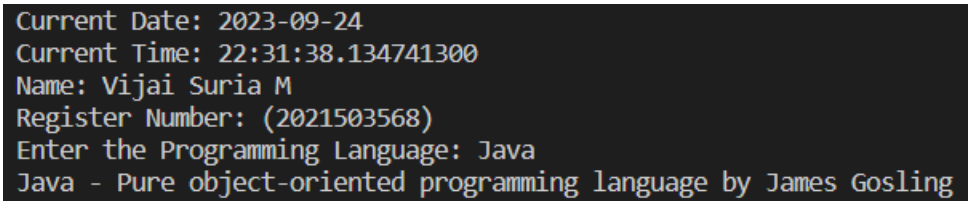
```
import java.time.LocalDate;
import java.time.LocalTime;
import java.util.Scanner;

public class DictionaryPL3568 {
    public static void main(String[] args) {
        System.out.println("Current Date: " + LocalDate.now());
        System.out.println("Current Time: " + LocalTime.now());
        System.out.println("Name: Vijai Suria M \nRegister Number:
(2021503568)");
        // Define a 2D string array to store programming languages and their
definitions
        String[][] programmingLanguages = {
            {"Java", "Pure object-oriented programming language by James
Gosling"},
            {"C++", "Object-oriented programming language by Stroustrup"},
            // Add definitions for more programming languages here
```

```
        {"Python", "High-level programming language known for its
simplicity"},
        {"JavaScript", "Scripting language commonly used for web
development"},
        {"C#", "Object-oriented language developed by Microsoft"},
        {"Ruby", "Dynamic, reflective, and object-oriented language"},
        {"Swift", "Apple's programming language for iOS and macOS
development"},
        {"Kotlin", "Modern statically-typed language for Android
development"},
        {"Go", "Concurrent and statically typed language developed by
Google"},
        {"Rust", "Systems programming language focused on safety and
performance"}
    };

    Scanner sc = new Scanner(System.in);
    // Example input programming language name
    System.out.print("Enter the Programming Language: ");
    String inputLanguage = sc.nextLine(); // Change this to the language
you want to look up

    // Call the method to get the definition and print it
    String definition = getDefinition(programmingLanguages,
inputLanguage);
    if (definition != null) {
        System.out.println(inputLanguage + " - " + definition);
    } else {
        System.out.println("Programming language not found in the
dictionary.");
    }
  }

  // Method to retrieve the definition for a given programming language
  public static String getDefinition(String[][] languages, String
languageName) {
    for (String[] language : languages) {
        if (language[0].equalsIgnoreCase(languageName)) {
```

```
            return language[1];
        }
    }
    return null; // Return null if the language is not found
  }
}
```

OUTPUT:





5.9. Write a program to implement Hangman game in Java, a word-guessing game where one player thinks of a Secret word, and another player (the computer i.e., Guesser or user) tries to guess it by stating one letter at a time. If the guessed letter is in the secret word, it's revealed in the display; otherwise the number of attempt is reduced by one. The player usually has a limited number of incorrect guesses allowed before they lose the game. The game ends when the player successfully guesses the word (win) or when they run out of allowed incorrect guesses (lose).The game typically has the following components:

Input:

The guesser or user tries to guess the secret word by stating each of the character in the secret word.

Output:

A representation of the secret word is displayed with underscores for each letter that hasn't been guessed yet.

For example, if the secret word is "java" and the player has guessed 'a', the display would be "_ a _ a".

Method: Guessing

Guessing: Fix the maximum attempts. Create a guessed Letter Boolean array of secret word length size to verify the guessed character of secret word.

Process the guess [10 points each]

i. Check input is a valid input (i.e., a character) Check if the guessed letter is in the secret word.
ii. Update guessed Letters array.
iii. Check if the entire word has been guessed.
iv. Increment attempts if the guess is incorrect.
v. Print game messages Win/Lose Conditions
vi. The player receives feedback on their guesses, including whether the guessed letter is in the word and the current state of the word display.
vii. Replay: After the game ends, the players may choose to play again with a new secret word.

Increase the complexity of the program by having more than one secret Word and choose it randomly. Give clues about the secret Word and reduce the points accordingly.

CODE:

```
import java.time.LocalDate;
import java.time.LocalTime;
import java.util.*;

class Hangman3568 {
    public static void main(String[] args) {
        System.out.println("Current Date: " + LocalDate.now());
        System.out.println("Current Time: " + LocalTime.now());
        System.out.println("Name: Vijai Suria M \nRegister Number:
```

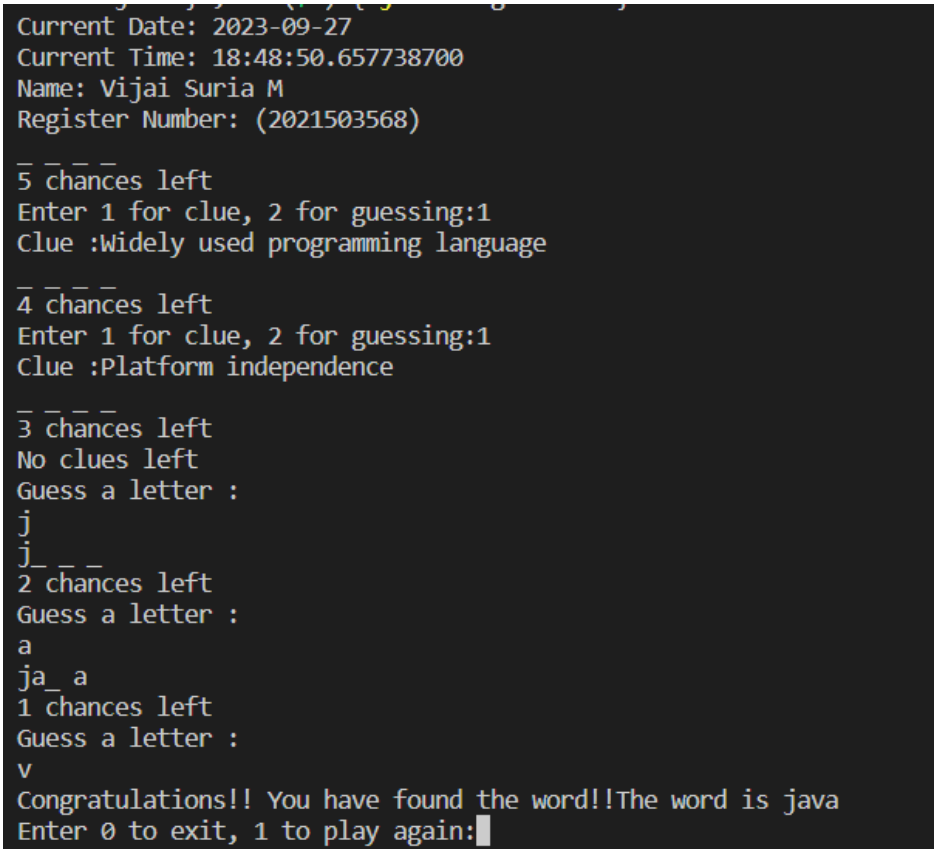
```
(2021503568)");

        String words[] = { "java", "internship", "college", "computer", "github" };
        String hints[][] = {
                { "Widely used programming language", "Platform independence" },
                { "Temporary work experience", "Academic years" },
                { "Higher education institution", "Beyond high school" },
                { "Data processing device", "Calculations electronically" },
                { "Code collaboration platform", "Version control" }
        };

        Scanner in = new Scanner(System.in);

        int chances, length, mode;
        boolean k = true;
        while (k) {
            int clu = 0;
            Random rand = new Random();
            int ran = rand.nextInt(5);

            String tobe = words[ran];
            length = tobe.length();
            char[] actual = tobe.toCharArray();
            char[] guessword = new char[length];
            for (int i = 0; i < length; i++)
                guessword[i] = ' ';
            chances = length + (length / 3);

            boolean t = false;
            while (chances > 0) {
                for (int i = 0; i < length; i++) {
                    if (guessword[i] == ' ')
                        System.out.print("_ ");
                    else
                        System.out.print(guessword[i]);
                }
                System.out.println("\n" + chances + " chances left");
                char guess;
```

```
            if (clu <= 1) {
                System.out.print("Enter 1 for clue, 2 for guessing:");
                mode = in.nextInt();
            } else {
                if (t == false)
                    System.out.println("No clues left");
                t = true;
                mode = 2;
            }
            if (mode == 1) {

                System.out.println("Clue :" + hints[ran][clu]);

                clu++;
                chances--;
            } else {
                in.nextLine();
                System.out.println("Guess a letter : ");
                guess = in.next().charAt(0);
                chances--;
                if (!(guess >= 'a' && guess <= 'z')) {
                    System.out.println("Invalid input. Only give small letter
alphabets!");
                } else {
                    for (int i = 0; i < actual.length; i++) {
                        if (guess == actual[i]) {
                            guessword[i] = actual[i];
                        }
                    }
                }
            }
            boolean found = true;
            for (int i = 0; i < length; i++) {
                if (guessword[i] == ' ') {
                    found = false;
                    break;
                }
```

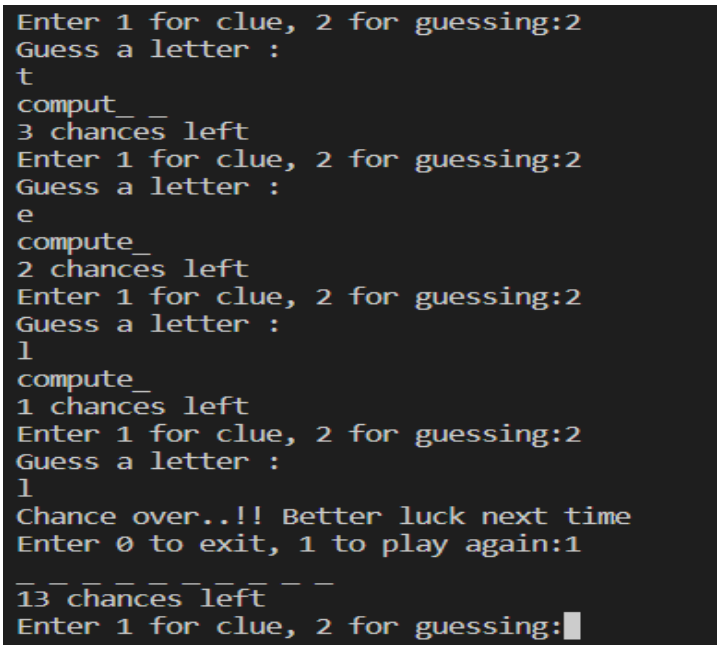
```
            }
            if (found == true) {
                break;
            }

        }
        boolean f = true;
        for (int i = 0; i < length; i++) {
            if (guessword[i] == ' ') {
                f = false;
                break;
            }
        }
        if (f == true) {
            System.out.println("Congratulations!! You have found the word!!The
word is " + tobe);
        } else {
            System.out.println("Chance over..!! Better luck next time");
        }
        int p = 0;
        System.out.print("Enter 0 to exit, 1 to play again:");
        p = in.nextInt();
        if (p == 0)
            k = false;

    }
    in.close();
  }
}
```

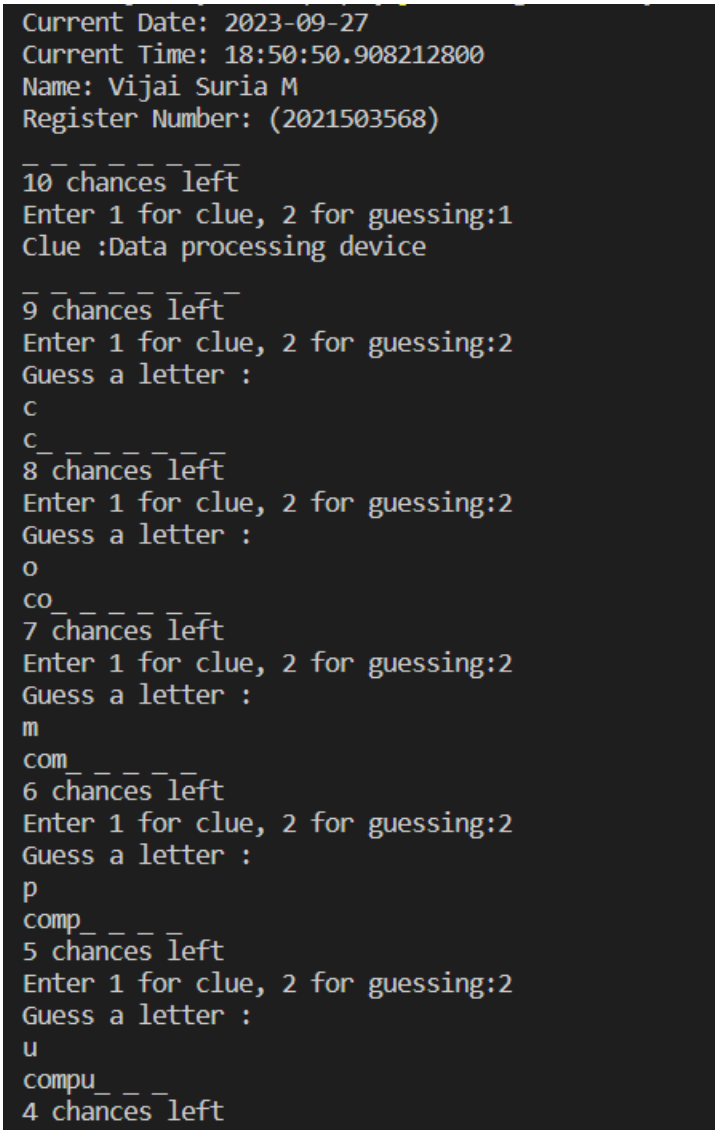
OUTPUT:

1) Player winning against computer:



2) Player losing against computer





Result: Thus, basic concepts of String methods in java were implemented successfully. And thus, outputs were verified.