

| | | |
|-------------|---|------------------|
| REG NO | : | 19BCS0012 |
| NAME | : | NITHISH G |
| COURSE CODE | : | 3001 |
| COURSE | : | JAVA PROGRAMMING |
| DATE | : | 24.03.2021 |

1. Write a java program to declare the variables of following data types, to initialise the values, to initialise the values for all variables and to display the same :

- Char
- Byte
- Short
- Int
- Long
- Float
- Double

Source code

```
public class datatypes {  
  
    public static void main(String[] args) {  
  
        char c = 'N';  
        byte b = 127;  
        short s = 32767;  
        int i = 2147483647;  
        long l = 9223372036854775807L;  
        float f = 7.96F;  
        double d = 20.65555455;
```

```

System.out.print("Name      : Nithish G \nReg No.      : 19BCS0012\n\n");
    System.out.println("The value for char is :" +c);
    System.out.println("The value for byte is :" +b);
    System.out.println("The value for short is:" +s);
    System.out.println("The value for int is      :" +i);
    System.out.println("The value for long is :" +l);
    System.out.println("The value for float is :" +f);
    System.out.println("The value for double is      :" +d);
}
}

```

Output :

The screenshot shows an IDE with two windows. The left window, titled 'Console', displays the output of a Java application. The right window, titled 'store_1st_appearance' and 'datatypes.java', shows the source code of the program.

Console Output:

```

Name      : Nithish G
Reg No.   : 19BCS0012

The value for char is      :N
The value for byte is      :127
The value for short is     :32767
The value for int is       :2147483647
The value for long is      :9223372036854775807
The value for float is     :7.96
The value for double is    :20.65555455

```

Source Code (datatypes.java):

```

1
2 class datatypes {
3
4     static void main(String[] ar
5
6     char c = 'N';
7     byte b = 127;
8     short s = 32767;
9     int i = 2147483647;
10    long l = 9223372036854775807L;
11    float f = 7.96F;
12    double d = 20.65555455;
13    System.out.print("\tName      : N
14    System.out.println("The value f
15    System.out.println("The value f
16    System.out.println("The value f
17    System.out.println("The value f
18    System.out.println("The value f
19    System.out.println("The value f
20    System.out.println("The value f

```

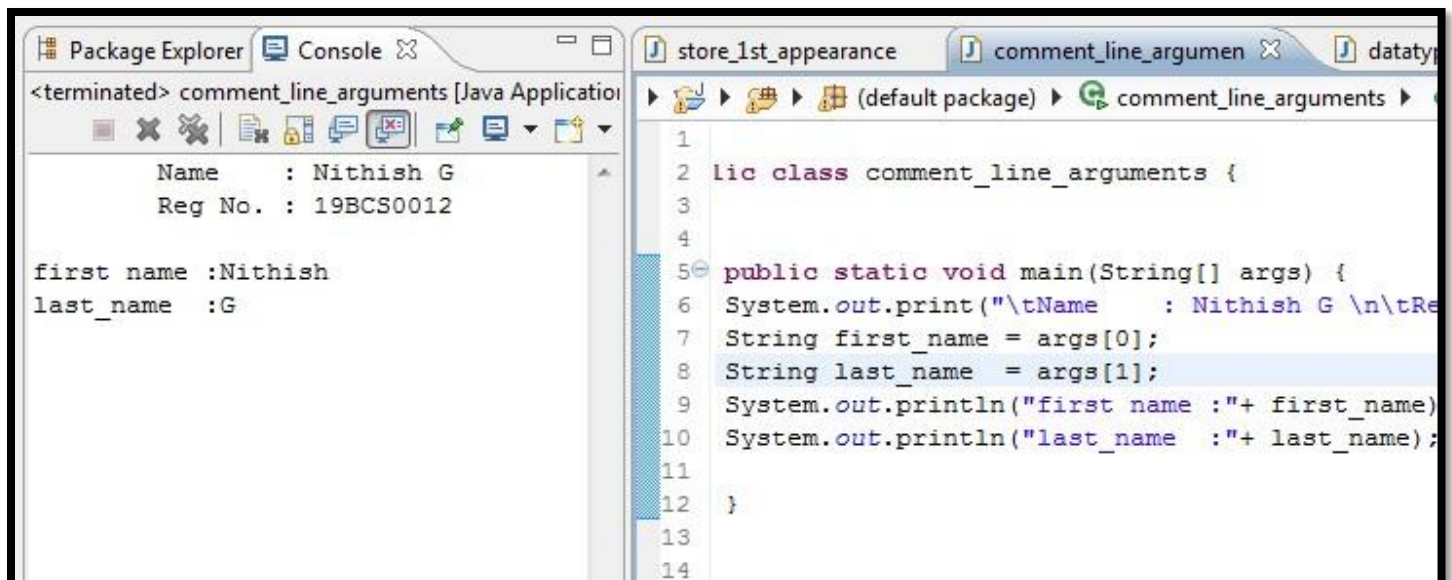
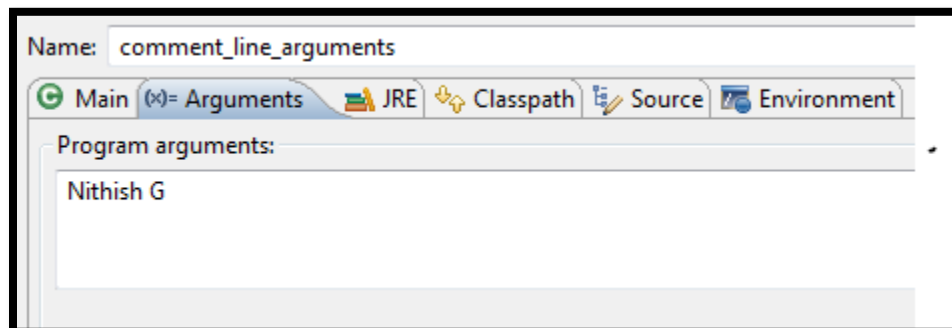
2. Write a java program to get your first name and last name from command line arguments and print the same.

Source code:

```
public class comment_line_arguments {  
  
    public static void main(String[] args) {  
  
        String first_name = args[0];  
        String last_name = args[1];  
        System.out.println("first name :"+ first_name);  
        System.out.println("last_name :"+ last_name);  
  
    }  
}
```

Comment line argument : Nithish_G , 19BCS0012

Output



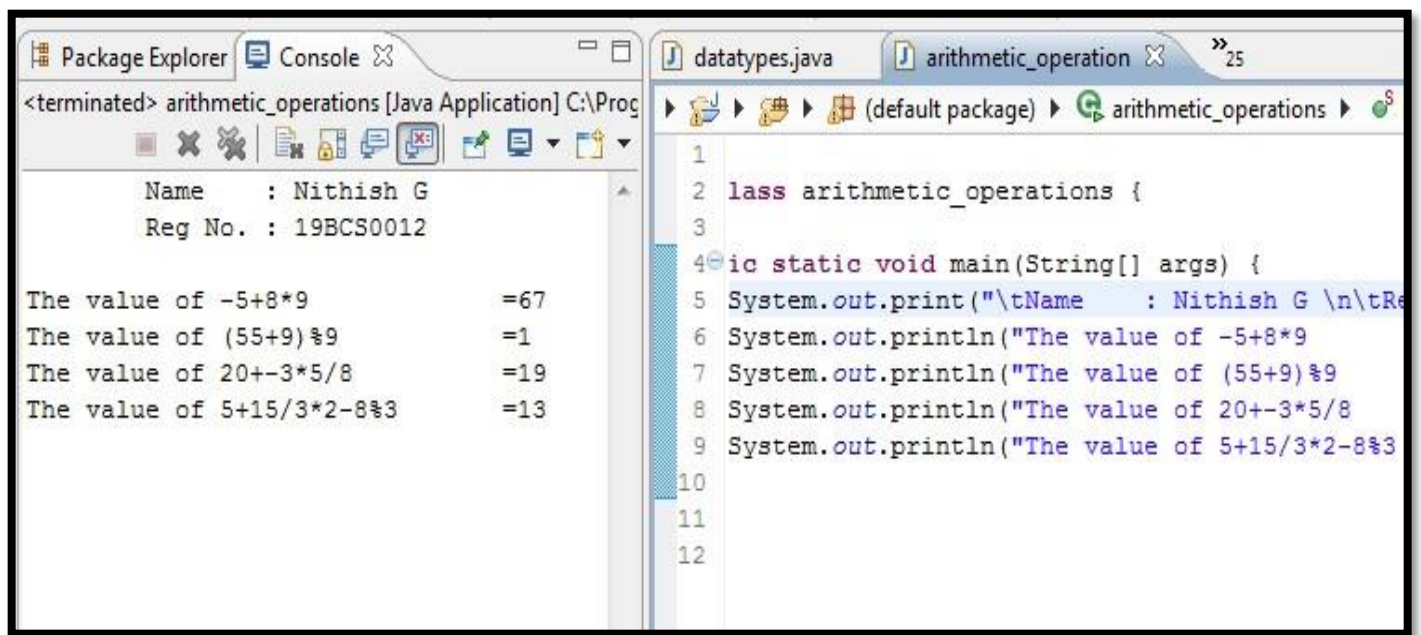
3. Write a java program to print the result of the following operations.

- a. $-5 + 8 * 6$
- b. $(55+9)\%9$
- c. $20 + -3*5/8$
- d. $5+15/3*2-8\%3$

Source code

```
public class arithmetic_operations {  
  
    public static void main(String[] args) {  
  
        System.out.println("The value of -5+8*9          =" +(-5+8*9));  
        System.out.println("The value of (55+9)%9        =" +(55+9)%9);  
        System.out.println("The value of 20+-3*5/8        =" +(20+-3*5/8));  
        System.out.println("The value of 5+15/3*2-8%3      =" +(5+15/3*2-8%3));  
  
    }  
  
}
```

Output



4. Find the area of the circle using comment line argument.

Source code:

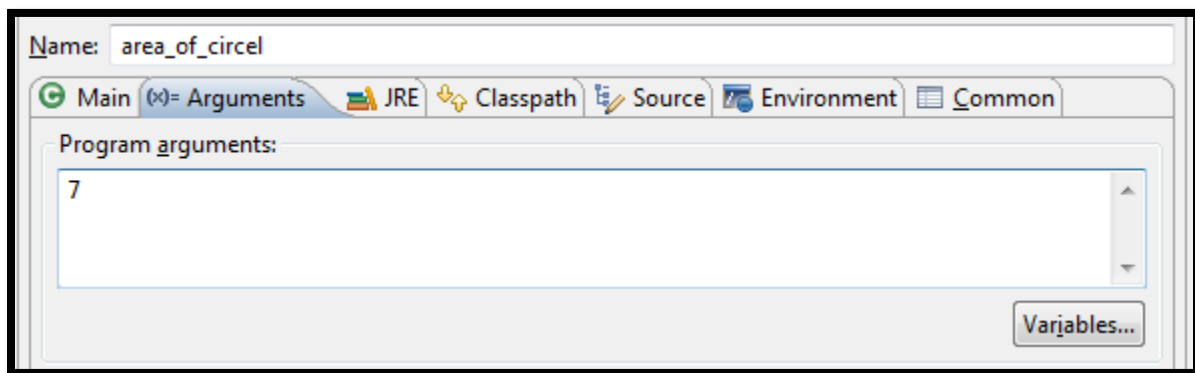
```
import java.lang.Math;

public class area_of_circel {

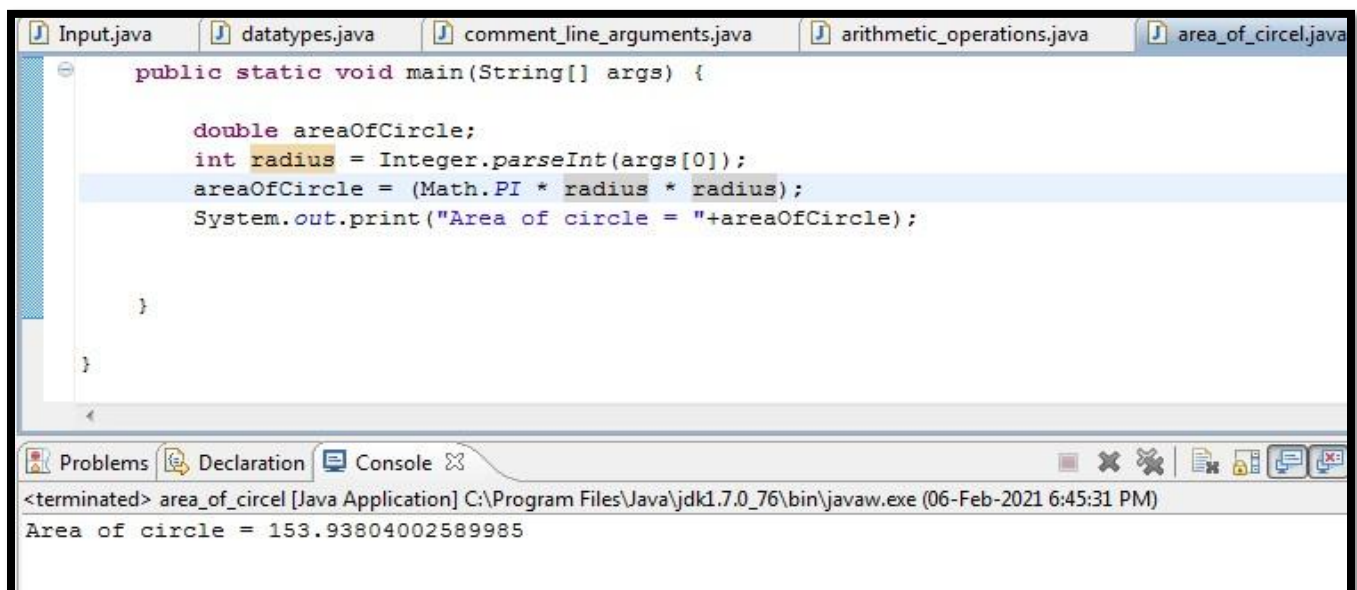
    public static void main(String[] args) {

        double areaOfCircle;
        int radius = Integer.parseInt(args[0]);
        areaOfCircle = (Math.PI * radius * radius);
        System.out.print("Area of circle = "+areaOfCircle);
    }
}
```

Comment line argument



Output:



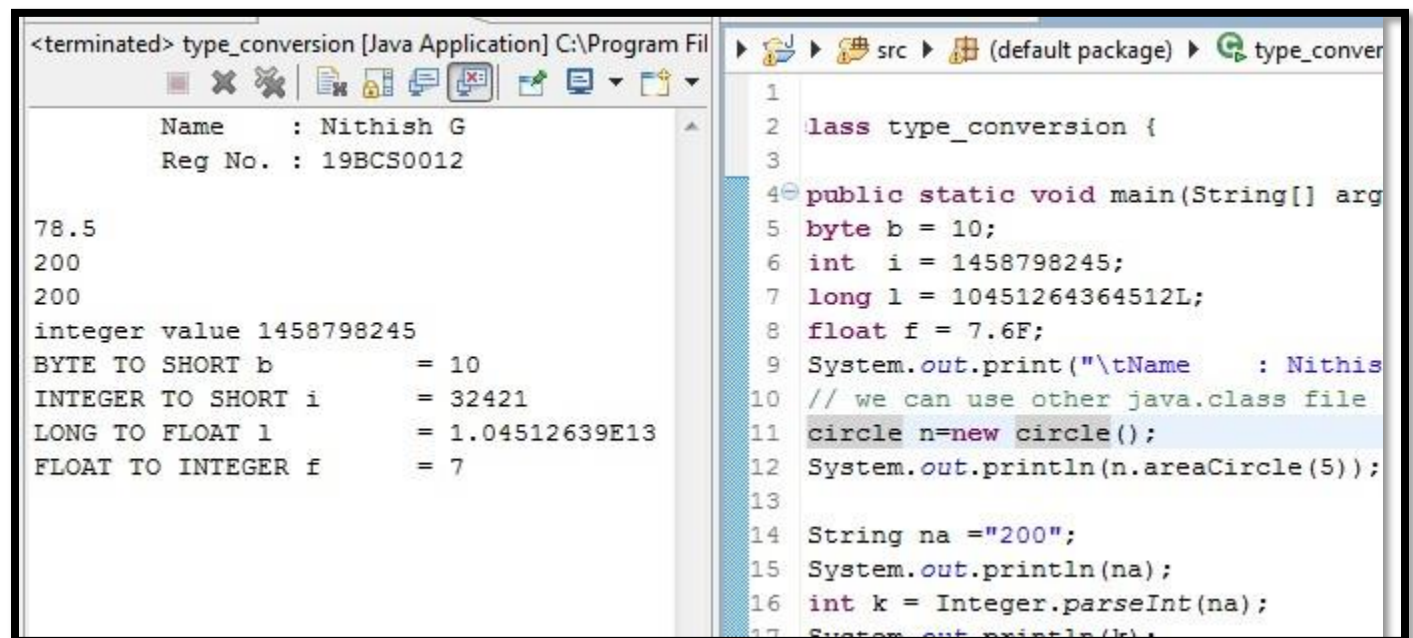
5. Type conversion

Source code:

```
public class type_conversion {  
  
    public static void main(String[] args) {  
        byte b = 10;  
        int i = 1458798245;  
        long l = 10451264364512L;  
        float f = 7.6F;
```

```
        short b_To_s = (short)b; //BYTE TO SHORT  
        short i_To_s = (short)i; //INTEGER TO SHORT  
        float l_To_f = (float)l; //LONG TO FLOAT  
        int f_To_i = (int)f; //FLOAT TO INTEGER
```

```
        System.out.println("BYTE TO SHORT b    = "+b_To_s);  
        System.out.println("INTEGER TO SHORT i    = "+i_To_s);  
        System.out.println("LONG TO FLOAT l    = "+l_To_f);  
        System.out.println("FLOAT TO INTEGER f = "+f_To_i);  
    } Output:
```



The screenshot displays a Java IDE with two windows. The left window, titled '<terminated> type_conversion [Java Application] C:\Program Fil', shows the output of the program. The right window, titled 'src \> (default package) \> type_conver', shows the source code.

Output:

```
78.5  
200  
200  
integer value 1458798245  
BYTE TO SHORT b          = 10  
INTEGER TO SHORT i       = 32421  
LONG TO FLOAT l          = 1.04512639E13  
FLOAT TO INTEGER f       = 7
```

Source Code:

```
1  
2 class type_conversion {  
3  
4 public static void main(String[] arg  
5 byte b = 10;  
6 int i = 1458798245;  
7 long l = 10451264364512L;  
8 float f = 7.6F;  
9 System.out.print("\tName      : Nithis  
10 // we can use other java.class file  
11 circle n=new circle();  
12 System.out.println(n.areaCircle(5));  
13  
14 String na ="200";  
15 System.out.println(na);  
16 int k = Integer.parseInt(na);  
17 System.out.println(k);
```


LAB - 2

6. Write a java code for finding the area of the rectangle by getting the height and width from user.

Source code :

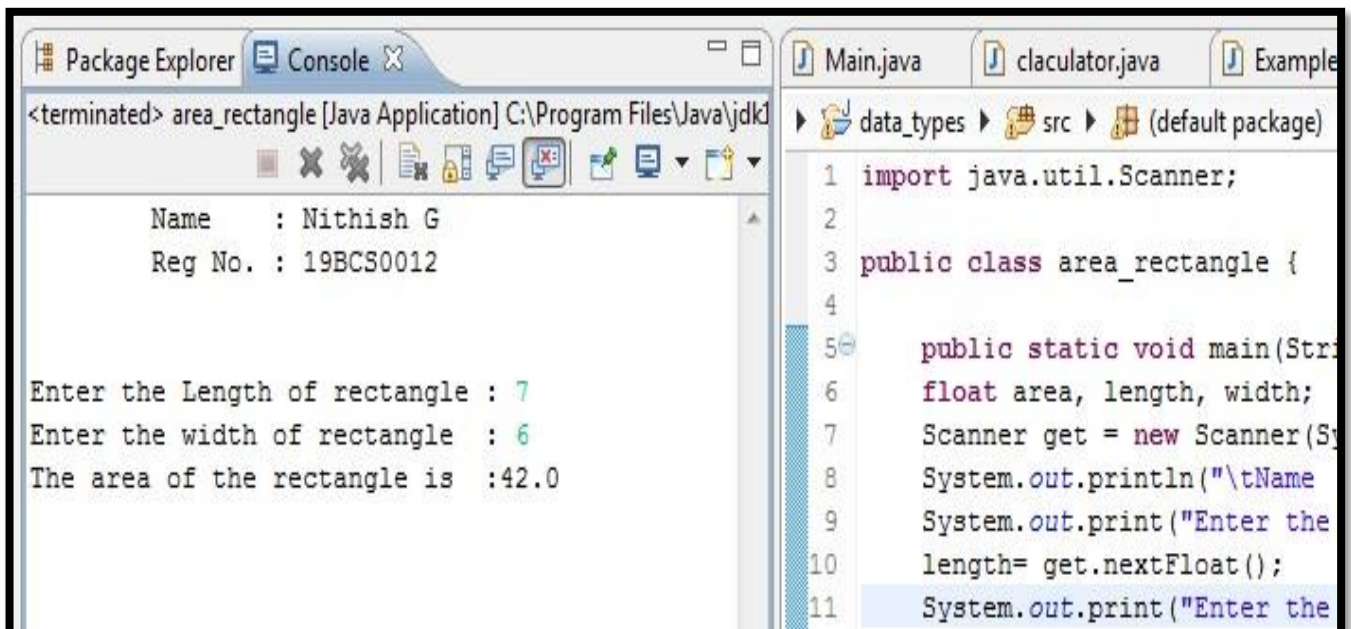
```
import java.util.Scanner;

public class area_rectangle {

    public static void main(String[] args) {
        float area, height, width;
        Scanner get = new Scanner(System.in);
        System.out.println("\tName      : Nithish G\n\tReg No.      : 19BCS0012\n\n");
        System.out.println("Enter the height of rectangle : ");
        height= get.nextFloat();
        System.out.println("Enter the width of rectangle : ");
        width= get.nextFloat();
        area = width * height;
        System.out.println("The area of the rectangle is : " + area);

    } }
```

Output :



```
<terminated> area_rectangle [Java Application] C:\Program Files\Java\jdk1
Name      : Nithish G
Reg No.   : 19BCS0012

Enter the Length of rectangle : 7
Enter the width of rectangle  : 6
The area of the rectangle is  : 42.0

1 import java.util.Scanner;
2
3 public class area_rectangle {
4
5     public static void main(Stri
6     float area, length, width;
7     Scanner get = new Scanner(Sy
8     System.out.println("\tName
9     System.out.print("Enter the
10    length= get.nextFloat());
11    System.out.print("Enter the
```

7. Create a class in java called Calculator, declare two float variables and four member functions named Add() for addition, Sub() for subtraction, Mulo for multiplication and Div() for division. Perform appropriate operations by getting two float values from user and calling the methods from the another class called Example by creating object of class Calculator.

Source code:

```
import java.util.Scanner;
public class Example {

    public static void main(String[] args) {

        claculator c =new claculator();
        float a,b;
        Scanner get = new Scanner(System.in);
        System.out.println("\tName      : Nithish G\n\tReg No.      : 19BCS0012\n\n");
        System.out.println(" Enter the 1st float value : ");
        a = get.nextFloat();
        System.out.println(" Enter the 2nd float value : ");
        b = get.nextFloat();
        System.out.println("1. Addition of      "+a+" + "+b+" : "+c.add(a, b));
        System.out.println("2. subtraction of   "+a+" - "+b+" : "+c.sub(a, b));
        System.out.println("3. Mutiplication of "+a+" * "+b+" : "+c.mul(a, b));
        System.out.println("4. Division of "+a+" / "+b+" : "+c.div(a, b));
    }
}

public class claculator
{
    float a=0,b=0;
    float add(float a,float b)
    {
```



```

    return a+b;
}

float sub(float a,float b)
{
    return a-b;
}

float mul(float a,float b)
{
    return a*b;
}

float div(float a,float b)
{
    return a/b;
}
}

```

Output :

The screenshot shows a Java IDE with two main windows: Package Explorer and Console. The Package Explorer shows the project structure with a package named 'data_types' containing a class 'Example'. The Console window displays the output of the program, which prompts the user to enter two float values (6.4 and 3.2) and then performs four operations: addition, subtraction, multiplication, and division. The results are displayed as follows:

| Operation | Calculation | Result |
|---------------------|-------------|-----------|
| 1. Addition of | 6.4 + 3.2 | 9.6 |
| 2. subtraction of | 6.4 - 3.2 | 3.2 |
| 3. Mutiplication of | 6.4 * 3.2 | 20.480001 |
| 4. Division of | 6.4 / 3.2 | 2.0 |

The Editor window shows the source code of the 'Example.java' file, which includes the following code:

```

1 import java.util.Scanner;
2 public class Example {
3
4
5     public static void main(String[] args) {
6
7         claculator c =new claculator();
8         float a,b;
9         Scanner get = new Scanner(System.in);
10        System.out.println("\tName : Nithish G");
11        System.out.print(" Enter the 1st float value : ");
12        a = get.nextFloat();
13        System.out.print(" Enter the 2nd float value : ");

```

8. Write a Java code to get employee ID(6-digits), Employee Name and Employee Designation for 5 employees using classes and Array of Objects concept

Source code:

```
import java.util.Scanner;

public class employeesample {

    long empid;

    String em_name;

    String em_des;

    public static void main(String[] args) {

        employeesample[] em = new employeesample[5];

        Scanner read = new Scanner (System.in);

        System.out.println("\tName      : Nithish G\n\tReg No.      : 19BCS0012\n\n");

        int i=0;

        for(i=0;i<5;i++)

        {

            em[i]=new employeesample();

            System.out.println( i+1 +". Enter the employee details ");

            System.out.println("Enter the employee id : ");

            em[i].empid=read.nextLong();

            System.out.println("Enter the employee name :");

            em[i].em_name = read.next();

            System.out.println("Enter the employee designation :");

            em[i].em_des = read.next();

        }

        for(i=0;i<5;i++)

        {

            System.out.println( i+1 +". employee details ");
```

```
System.out.println(" the employee id : "+em[i].empid);
```

```
System.out.println("the employee name : " + em[i].em_name);
```

```
System.out.println(" the employee designation : " +em[i].em_des );
```

```
}
```

```
}}
```

Output:

The screenshot shows an IDE with two main windows. The left window, titled 'Console', displays the output of a Java application. The output shows the details of five employees, each with their ID, name, and designation. The right window, titled 'Example.java', shows the source code of the application. The code includes a loop that reads employee details from a file and prints them to the console.

```
<terminated> employeesample [Java Application] C:\Program Files\Java\j...  
Name      : Nithish G      Reg No. : 19BCS0012  
Employee details  
  
Employee : 1  
Employee Id      : 432101  
Employee Name    : Nithish  
employee Designation : Manager  
  
Employee : 2  
Employee Id      : 432102  
Employee Name    : Yash  
employee Designation : Asst.Manager  
  
Employee : 3  
Employee Id      : 432103  
Employee Name    : Aanav  
employee Designation : Team_Leader  
  
Employee : 4  
Employee Id      : 432104  
Employee Name    : Aakil  
employee Designation : Team_member  
  
Employee : 5  
Employee Id      : 432105  
Employee Name    : Vedant
```

```
Example.java  
area_rectangle.  
data_types  
src  
(defa  
12  
13 : i=0;  
14 : (i=0;i<5;i++)  
15  
16 em[i]=new employeesam  
17 System.out.println("  
18 System.out.print("Ent  
19 em[i].empid=read.next  
20 System.out.print("Ent  
21 em[i].em_name = read.  
22 System.out.print("Ent  
23 em[i].em_des = read.r  
24 System.out.println("\n  
25  
26 stem.out.println("Name  
27 stem.out.println(" E  
28 : (i=0;i<5;i++)  
29  
30 System.out.println("\n  
31  
32 System.out.println("  
33  
34 System.out.println("  
35  
36 System.out.println("  
37  
38  
39  
40  
Problems @ Javadoc De  
0 errors, 8 warnings, 0 others
```

9. Write a java code to get two values of Name, Age and ID from the user using *BufferedReader* and *InputStreamReader* packages

Source code:

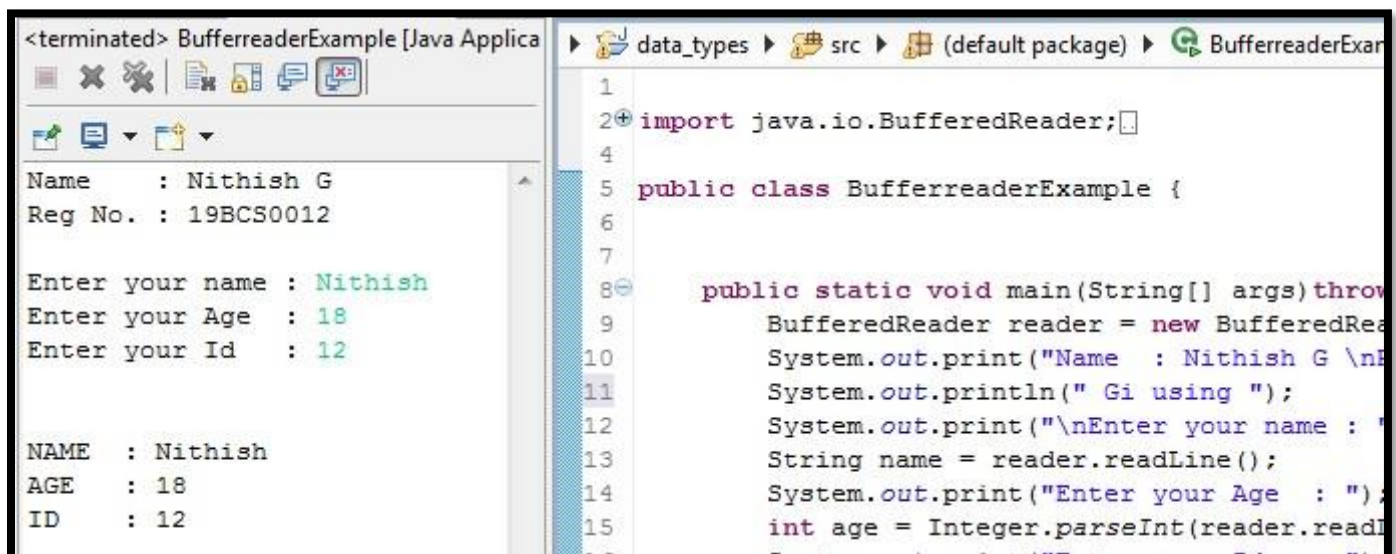
```
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;

public class BufferreaderExample {

    public static void main(String[] args)throws Exception {
        BufferedReader reader = new BufferedReader(new InputStreamReader(System.in));

        System.out.println("Enter your name : ");
        String name = reader.readLine();
        System.out.println("Enter your Age : ");
        int age = Integer.parseInt(reader.readLine());
        System.out.println("Enter your Id : ");
        int ID = Integer.parseInt(reader.readLine());
        System.out.println("NAME : "+ name);
        System.out.println("AGE : "+age);
        System.out.println("ID  : "+ID);
    }
}
```

Output :



The screenshot shows a Java IDE with two panes. The left pane, titled "BufferreaderExample [Java Application]", displays the program's output. It shows the user input for Name, Age, and ID, followed by the formatted output. The right pane shows the source code of the program, which is the same as the code provided in the previous block.

Output:

```
Name      : Nithish G
Reg No.   : 19BCS0012

Enter your name : Nithish
Enter your Age  : 18
Enter your Id   : 12

NAME : Nithish
AGE  : 18
ID   : 12
```

Source Code:

```
1 import java.io.BufferedReader;
2
3
4
5 public class BufferreaderExample {
6
7
8     public static void main(String[] args)throws
9         BufferedReader reader = new BufferedRea
10        System.out.print("Name  : Nithish G \nB
11        System.out.println(" Gi using ");
12        System.out.print("\nEnter your name : "
13        String name = reader.readLine();
14        System.out.print("Enter your Age  : ");
15        int age = Integer.parseInt(reader.readI
```

Lab-3

10. Write a java code to get First Name, Last name, and age of two persons from the user and display whether they are in same generation or not (if age difference is less than 15, same generation) along with their names.

Source code

```
import java.util.Scanner;
public class generation {

    String fn;
    String ln;
    int age;

    public static void main(String[] args) {

        generation[] g =new generation[2];
        Scanner r = new Scanner(System.in);
        int i,dif=0;
        System.out.print("Name : Nithish G \nReg No. : 
19BCS0012\n");
        for( i =0;i<2;i++)
        {
            g[i]=new generation();
            System.out.println("    person - "+(i+1));
            System.out.print(" first name : " );
            g[i].fn=r.next();
            System.out.print(" last name : " );
            g[i].ln=r.next();
            System.out.print(" age          : " );
            g[i].age=r.nextInt();
            dif=Math.abs(g[i].age-dif);

        }

        for(i=0;i<2;i++)
        {
            System.out.println("    person - "+(i+1));
            System.out.println(" first name : "+g[i].fn );

            System.out.println(" last name : "+g[i].ln );

            System.out.println(" age          : "+g[i].age );
```

```

    }
    if (dif<15)
    {

        System.out.println(" the Age differenc between them " + dif);
        System.out.println("so both are same generation ");
    }
    else
    {

        System.out.println(" the Age differenc between them " + dif);
        System.out.println("so both are different generation ");
    }

}

}

}

```

Output:

The screenshot shows an IDE with two windows. The left window, titled 'Console', displays the output of a Java application. The right window, titled 'employeesample.java', shows the source code.

Console Output:

```

<terminated> generation [Java Application] C:\Program Files\
Name      : Nithish G
Reg No.   : 19BCS0012
    person - 1
    first name : Nithish
    last name  : G
    age       : 18
    person - 2
    first name : Zack
    last name  : king
    age       : 38
    person - 1
    first name : Nithish
    last name  : G
    age       : 18
    person - 2
    first name : Zack
    last name  : king
    age       : 38
    the Age differenc between them 20
    so both are different generation

```

Source Code (employeesample.java):

```

10  generation[] g =new generation[2];
11  Scanner r = new Scanner(System.in);
12  int i,dif=0;
13  System.out.print("Name : Nithish G
14  for( i =0;i<2;i++)
15  {
16      g[i]=new generation();
17      System.out.println("    person -
18      System.out.print(" first name :
19      g[i].fn=r.next();
20      System.out.print(" last name :
21      g[i].ln=r.next();
22      System.out.print(" age      :
23      g[i].age=r.nextInt();
24      dif=Math.abs(g[i].age-dif);
25
26  }
27
28  for(i=0;i<2;i++)
29  {
30      System.out.println("    person -
31      System.out.println(" first name

```


11. Write a java code to display the truth table of logic gates using logical operators as shown below.

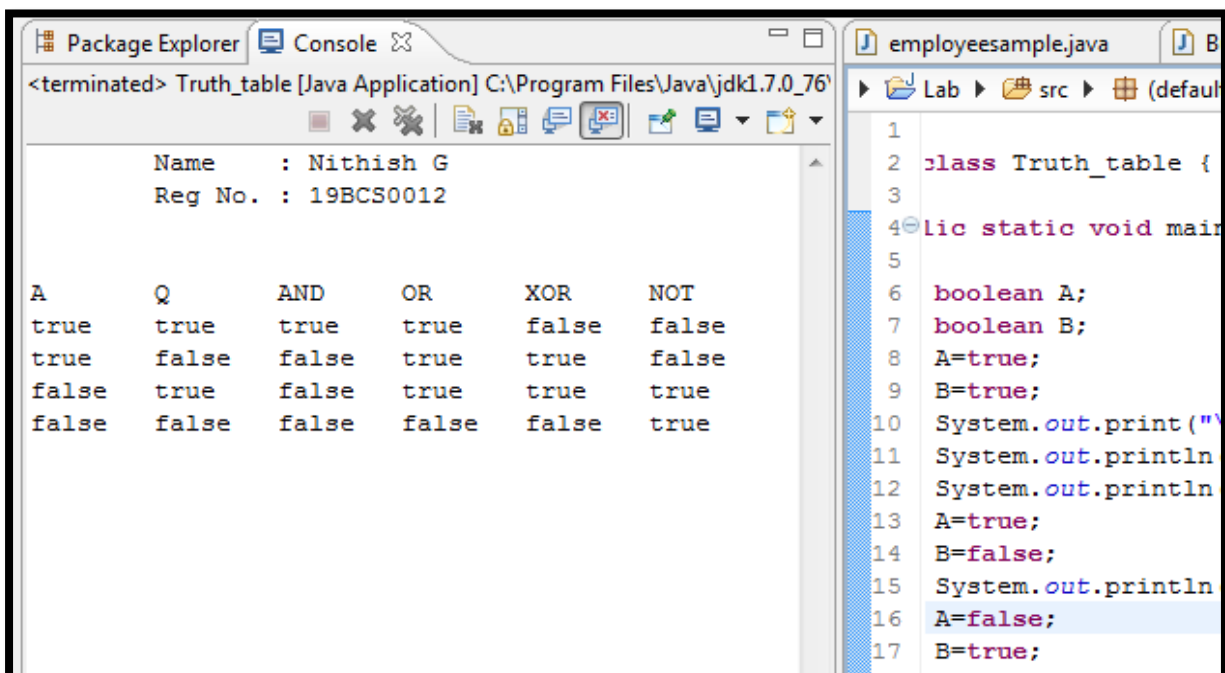
SOURCE CODE :

```
public class Truth_table {

    public static void main(String[] args) {

        boolean A;
        boolean B;
        A=true;
        B=true;
        System.out.print("\tName    : Nithish G \n\tReg No.        : 19BCS0012\n");
        System.out.println("\n\nA\tQ\tAND\tOR\tXOR\tNOT");
        System.out.println((A)+"\t"+(B)+"\t"+(A&&B)+"\t"+(A|B)+"\t"+(A^B)+"\t"+(!A));
        A=true;
        B=false;
        System.out.println((A)+"\t"+(B)+"\t"+(A&&B)+"\t"+(A|B)+"\t"+(A^B)+"\t"+(!A));
        A=false;
        B=true;
        System.out.println((A)+"\t"+(B)+"\t"+(A&&B)+"\t"+(A|B)+"\t"+(A^B)+"\t"+(!A));
        A=false;
        B=false;
        System.out.println((A)+"\t"+(B)+"\t"+(A&&B)+"\t"+(A|B)+"\t"+(A^B)+"\t"+(!A));
    }
}
```

OUTPUT



The screenshot shows a Java IDE with two windows. The left window, titled 'Console', displays the output of the program. It starts with a header for the truth table and then shows a table of values for A, B, AND, OR, XOR, and NOT. The right window, titled 'employeesample.java', shows the source code of the program, which is the same as the one provided in the 'SOURCE CODE' section.

```
<terminated> Truth_table [Java Application] C:\Program Files\Java\jdk1.7.0_76
Name    : Nithish G
Reg No. : 19BCS0012

A      Q      AND      OR      XOR      NOT
true   true   true    true    false   false
true   false  false   true    true    false
false  true    false   true    true    true
false  false  false   false   false   true

class Truth_table {
    public static void main
        boolean A;
        boolean B;
        A=true;
        B=true;
        System.out.print("\
        System.out.println
        System.out.println
        A=true;
        B=false;
        System.out.println
        A=false;
        B=true;
```


12. Print the ASCII values of the letters present in your name.

SOURCE CODE:

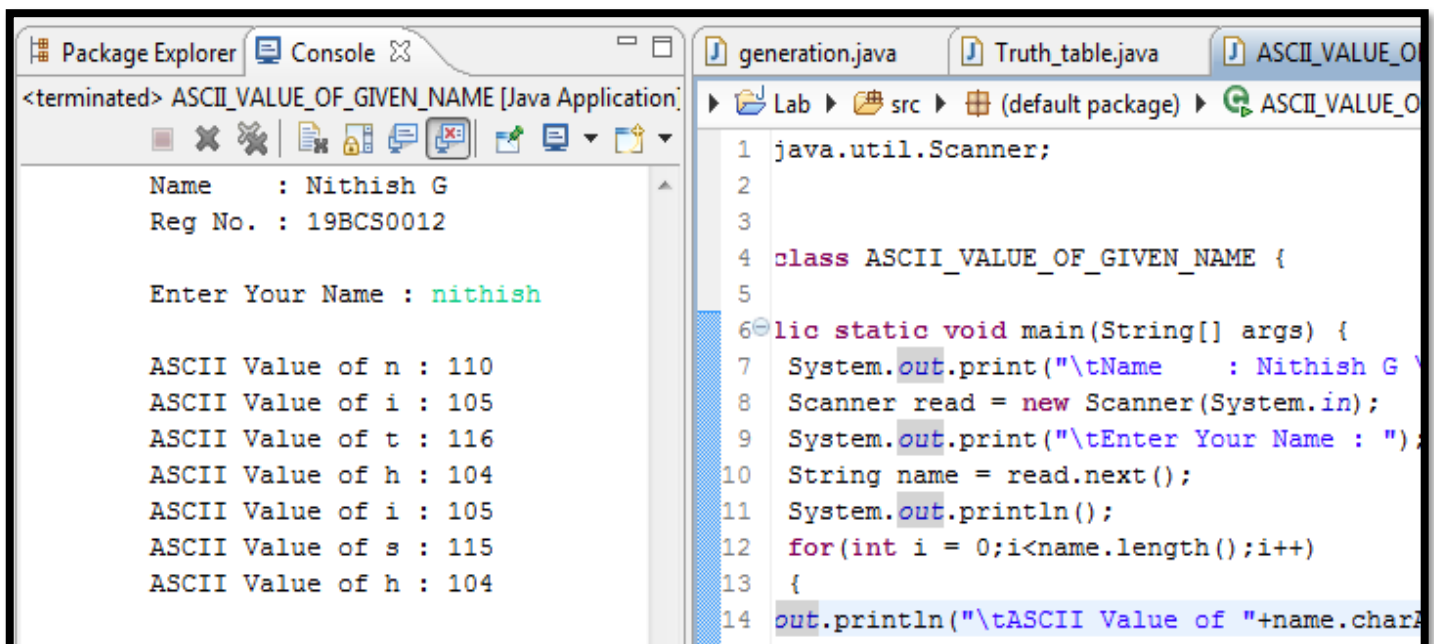
```
import java.util.Scanner;

public class ASCII_VALUE_OF_GIVEN_NAME {

    public static void main(String[] args) {
        System.out.print("\tName : Nithish G \n\tReg No. : 19BCS0012\n\n");
        Scanner read = new Scanner(System.in);
        System.out.print("\tEnter Your Name : ");
        String name = read.next();
        System.out.println();
        for(int i = 0;i<name.length();i++)
        {
            System.out.println("\tASCII Value of "+name.charAt(i)+" : " + (int)(name.charAt(i)));
        }

    }
}
```

Output:

The screenshot shows a Java IDE with two main panes. The left pane is the 'Console' window, which displays the output of the program. It shows the name 'Nithish G' and registration number '19BCS0012' printed, followed by a prompt 'Enter Your Name : nithish'. Below this, the ASCII values for each character of 'nithish' are listed: n (110), i (105), t (116), h (104), i (105), s (115), and h (104). The right pane shows the source code file 'ASCII_VALUE_OF_GIVEN_NAME.java'. The code is a Java class with a main method that uses a Scanner to read a name and then iterates through each character to print its ASCII value. The code is color-coded, with keywords in purple, strings in blue, and comments in green. The package explorer on the left shows the project structure with 'Lab' as the root, containing 'src' and 'default package' folders, with the current file 'ASCII_VALUE_OF_GIVEN_NAME.java' selected in the 'default package'.

Lab-4

1. Write a Grade calculation program. Input to be taken as marks scored out of 100 in 5 subjects of a student. In case a student scores less than 50%, award him 'D' grade. If marks are between 50% & 60% then 'C' grade. If marks are between 60% and 80% then 'B' grade , otherwise if marks are between 80% and 100% then 'A' grade. (Do it using if & else if statements.)

Source code:

```
import java.util.Scanner;

public class Grade_if_else_if {
    public int percentage()
    {
        int maths,tamil,history,english,science;
        int percentage = 0 ;
        Scanner obj = new Scanner(System.in);

        System.out.print("Name   : Nithish G \nReg No.       : 19BCS0012\n\n");
        System.out.print("\tEnter the Subject Marks\n");
        System.out.println("");
        System.out.print("Maths   : ");
        maths = obj.nextInt();
        System.out.print("Tamil   : ");
        tamil = obj.nextInt();
        System.out.print("English : ");
        english = obj.nextInt();
        System.out.print("History : ");
        history = obj.nextInt();
        System.out.print("Science : ");
        science = obj.nextInt();
        percentage = (maths+tamil+history+english+science)/5;
        return percentage;
    }

    public static void main(String[] args) {

        Grade_if_else_if obj = new Grade_if_else_if();
        int percentage = 0;
```

```

percentage = obj.percentage();

System.out.println("\n\t Percentage : "+percentage+"%");
if (percentage < 50 )
{
    System.out.println("\n\t Grade : D");
}
else if (percentage > 50 && percentage < 60)
{
    System.out.println("\n\t Grade : C");
}
else if (percentage > 60 && percentage < 80)
{
    System.out.println("\n\t Grade : B");
}
else if (percentage > 80 && percentage < 100)
{
    System.out.println("\n\t Grade : A");
}
}

```

}
Output :

The screenshot shows a Java IDE with two main windows: a Console window on the left and a Code Editor window on the right.

Console Window:

```

<terminated> Grade_if_else_if [Java Application] C:\Program File
Name      : Nithish G
Reg No.   : 19BCS0012

Enter the Subject Marks

Maths      : 86
Tamil      : 91
English    : 85
History    : 94
Science    : 87

Percentage : 88.0%

Grade : A

```

Code Editor Window:

```

Truth_table.java  ASCII_VALUE_OF_GIVEN  Grade_if_else_if.java
Lab > src > (default package) > Grade_if_else_if.java
1 java.util.Scanner;
2
3 class Grade_if_else_if {
4
5     public static void main(String[] args) {
6         int maths,tamil,history,english,science;
7         float percentage = 0 ;
8         Scanner obj = new Scanner(System.in);
9
10        System.out.print("Name : Nithish G \nReg No. : ");
11        System.out.print("\tEnter the Subject Marks : ");
12        System.out.println("");
13        System.out.print("Maths : ");
14        maths = obj.nextInt();
15        System.out.print("Tamil : ");
16        tamil = obj.nextInt();

```

2. Repeat the above program using a 'Switch' statement.

Source code:

```
public class Grade_Switch {

    public static void main(String[] args) {

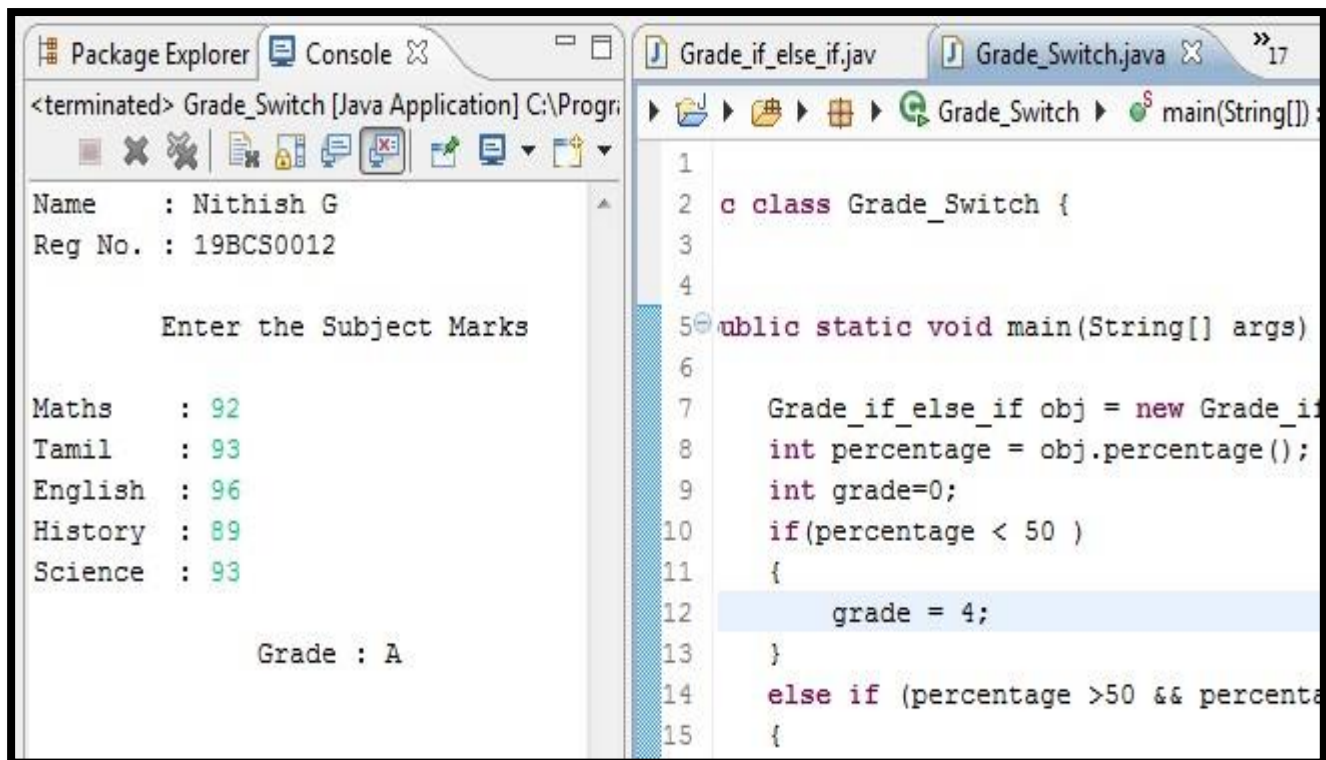
        Grade_if_else_if obj = new Grade_if_else_if();
        int percentage = obj.percentage();
        int grade=0;
        if(percentage < 50 )
        {
            grade = 4;
        }
        else if (percentage >50 && percentage < 60)
        {
            grade = 3;
        }
        else if (percentage>60 && percentage < 80)
        {
            grade = 2;
        }
        else if (percentage>80 && percentage < 100)
        {
            grade = 1;
        }
        switch(grade)
        {
            case 1 :
                System.out.println("\n\t      Grade : A");
                break;
            case 2 :
```

```

        System.out.println("\n\t\t\t\t\tGrade : B");
        break;
    case 3 :
        System.out.println("\n\t\t\t\t\tGrade : C");
        break;
    case 4:
        System.out.println("\n\t\t\t\t\tGrade : D");
        break;
    default:
        System.out.println("Enter the appropriate Marks");
    }
}

```

Output



3. Write a program that will get 'hour', 'minute' and 'second' of the current timing from your PC. And will print the message 'Good Morning' in case hour is less than 12 noon, it will print 'Good afternoon' if hour is between 12 & 6 p.m. and will print 'Good evening' otherwise. Also it should print the current time.

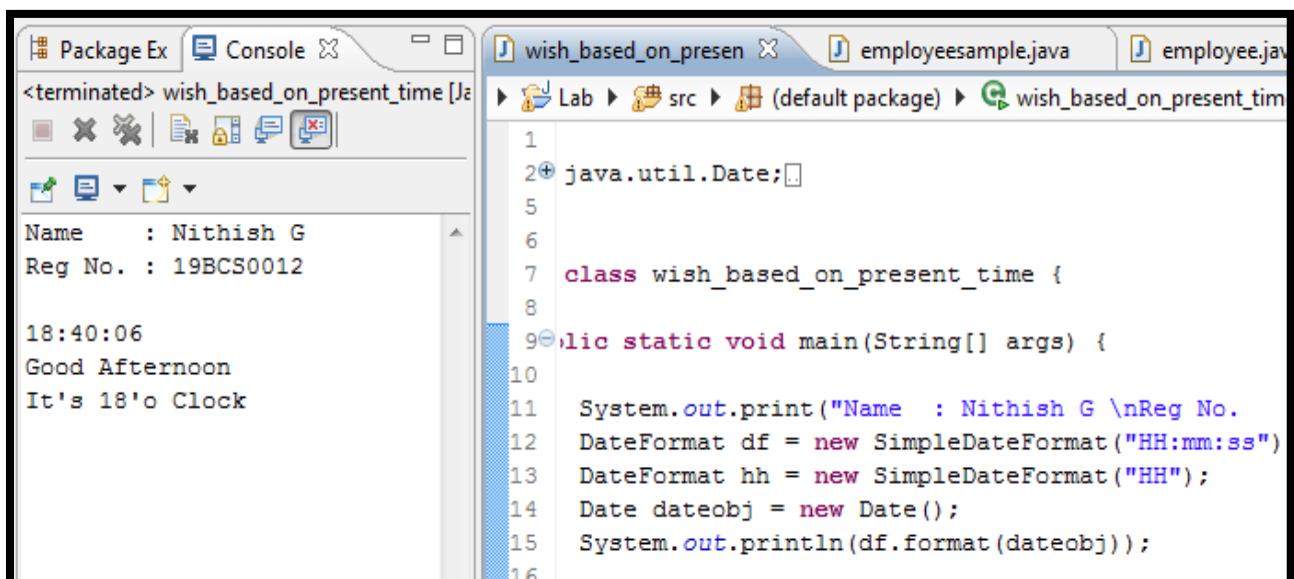
Source Code:

```
import java.util.Date;
import java.text.DateFormat;
import java.text.SimpleDateFormat;

public class wish_based_on_present_time {

    public static void main(String[] args) {
        System.out.print("Name : Nithish G \nReg No. : 19BCS0012\n\n");
        DateFormat df = new SimpleDateFormat("HH");
        Date dateobj = new Date();
        int time = Integer.parseInt(df.format(dateobj));
        if (time<=12)
            System.out.println("Good Moring \nIt's "+time+"o Clock");
        else if((time>12) &&( time<=18))
            System.out.println("Good Afternoon \nIt's "+time+"o Clock");
        else
            System.out.println("Good Evening \nIt's "+ (24-time)+"o Clock");
    }
}
```

Output



Lab5:

1. Write a program to check the divisibility of an integer by 3. Your program must make use of the fact that an integer is divisible by 3 if and only if the sum of its digits is divisible by 3. You must use this fact repeatedly, till the sum reduces to a single digit. For example, 123456789 is divisible by 3 if and only if $1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 = 45$ is divisible by 3. Now, 45 is divisible by 3 if and only if $4 + 5 = 9$ is divisible by 3. Observe that 9 is a single digit and is divisible by 3. Therefore, your program concludes that 123456789 is divisible by 3.

Source code:

```
import java.util.Scanner;
public class sum_each_num_from_given_single_num {

    public static void main(String[] args) {

        long n,m,sum=0,c=1,i;
        Scanner obj = new Scanner(System.in) ;
        System.out.print("Name      : Nithish G \nReg No. : 19BCS0012\n\n");
        System.out.print("enter the Number : ");
        n=obj.nextInt();

do{
    while(n>0)
    {
        m=n%10;
        sum+=m;
        n/=10;
    }

    if(sum>9)
    {
        n=sum;
        sum=0;
    }
}while(n>9);
```



```
System.out.println("\nSum of Each Digit until It Become Single Digit :"+sum);
```

```
    if(sum%3==0)
```

```
        System.out.println("\nIt's Divisible by 3");
```

```
    else
```

```
        System.out.println("It's not Divisible by 3");
```

```
    }
```

```
}
```

Output:

```
<terminated> sum_each_num_from_given_single_num [Java Application] C:\Progra
Package Explorer Console
Name : Nithish G
Reg No. : 19BCS0012
enter the Number : 123456789
Sum of Each Digit until It Become Single Digit :9
It's Divisible by 3
sum_each_num_from_gi type_conversion.java
main(String[]): void
1 java.util.Scanner;
2 class sum_each_num_from_given_single
3
4
5 public static void main(String[] args) {
6
7
8 long n,m,sum=0,c=1,i;
9 Scanner obj = new Scanner(System.in)
10 System.out.print("Name : Nithish G
11 System.out.print("enter the Number :
12 n=obj.nextInt();
13
14
15 while(n>0)
16 {
17     m=n%10;
18     sum+=m;
19     n/=10;
20 }
21
22 if(sum>9)
23 {
24     n=sum;
25     sum=0;
26 }
27
28 while(n>9);
29 out.println("\nSum of Each Digit until
```

2. Write a Java program to print sum of the squares of first n natural numbers.

Source code

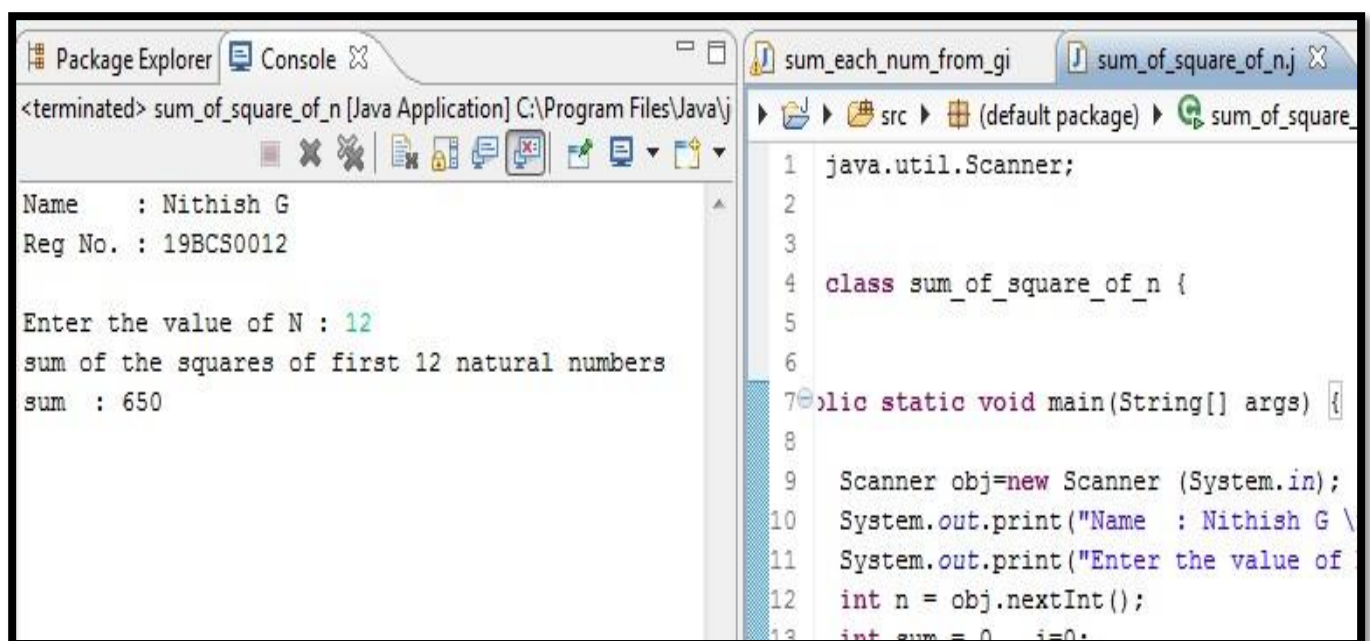
```
import java.util.Scanner;

public class sum_of_square_of_n {

    public static void main(String[] args) {

        Scanner obj=new Scanner (System.in);
        System.out.print("Name   : Nithish G \nReg No.       : 19BCS0012\n\n");
        System.out.print("Enter the value of N : ");
        int n = obj.nextInt();
        int sum = 0 , i=0;
        for(i=1;i<=n;i++)
        {
            sum += i * i;
        }
        System.out.println("sum of the squares of first " +n+ " natural numbers " );
        System.out.println("sum : " +sum);
    }
}
```

Output:

The screenshot shows an IDE with two windows. The left window, titled 'sum_of_square_of_n [Java Application]', displays the program's output in the console. The output shows the user's name 'Nithish G', registration number '19BCS0012', and the input value '12' for N. It then calculates and displays the sum of the squares of the first 12 natural numbers as 650. The right window, titled 'sum_of_square_of_n.j', shows the source code of the program, which is the same code as provided in the 'Source code' section. The code is written in Java and uses a Scanner to take input and System.out to print the results.

```
<terminated> sum_of_square_of_n [Java Application] C:\Program Files\Java\j
Name   : Nithish G
Reg No. : 19BCS0012

Enter the value of N : 12
sum of the squares of first 12 natural numbers
sum : 650

sum_each_num_from_gi
sum_of_square_of_n.j
1 java.util.Scanner;
2
3
4 class sum_of_square_of_n {
5
6
7 public static void main(String[] args) {
8
9     Scanner obj=new Scanner (System.in);
10    System.out.print("Name   : Nithish G \
11    System.out.print("Enter the value of
12    int n = obj.nextInt();
13    int sum = 0 , i=0;
```

3. Write a Java program to find the maturity value of a principal(P) due to the rate of compound interest(r).

Source Code:

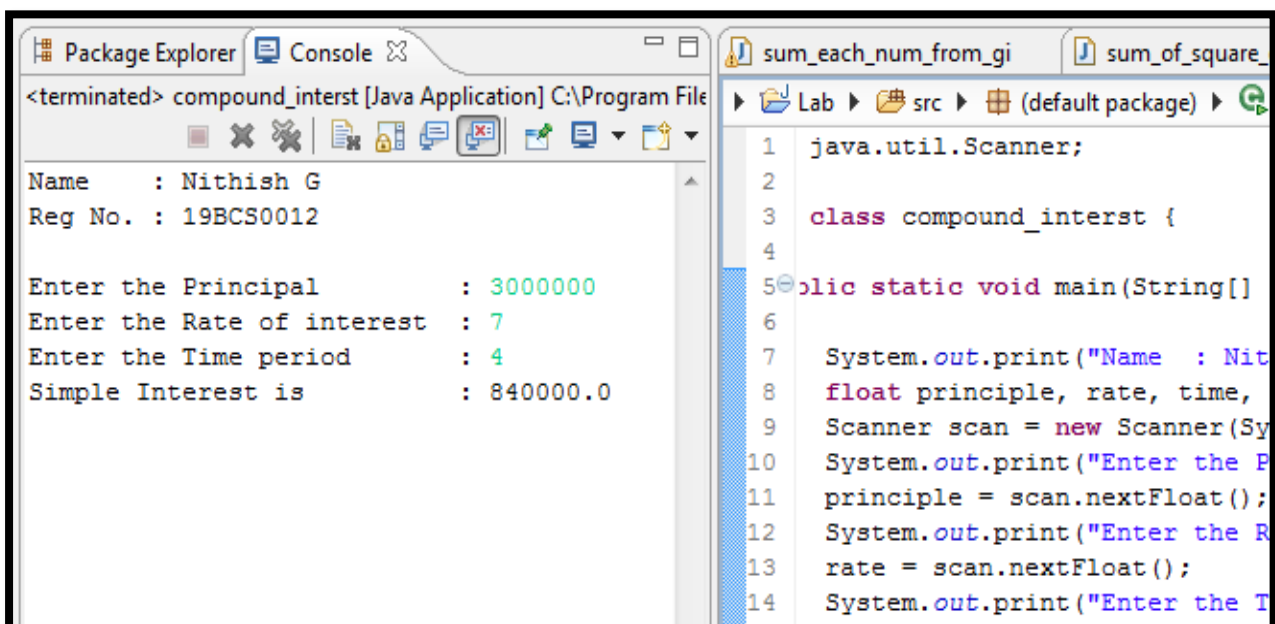
```
import java.util.Scanner;

public class compound_interst {

    public static void main(String[] args) {

        System.out.print("Name : Nithish G \nReg No. : 19BCS0012\n\n");
        float principle, rate, time, interest;
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter the Principal : ");
        principle = scan.nextFloat();
        System.out.print("Enter the Rate of interest : ");
        rate = scan.nextFloat();
        System.out.print("Enter the Time period : ");
        time = scan.nextFloat();
        scan.close();
        interest = (principle * rate * time) / 100;
        System.out.print("Simple Interest is : " + interest);
    }
}
```

Output:

The screenshot shows a Java IDE with two panes. The left pane, titled 'Console', displays the output of the program: 'Name : Nithish G', 'Reg No. : 19BCS0012', followed by prompts for 'Enter the Principal', 'Enter the Rate of interest', and 'Enter the Time period', with user inputs of 3000000, 7, and 4 respectively. The final output is 'Simple Interest is : 840000.0'. The right pane shows the source code of the 'compound_interst' class, which matches the code provided in the 'Source Code' section. The code is written in a standard Java IDE syntax with color-coding for keywords and strings. The package explorer on the left shows the project structure with 'Lab' and 'src' folders.

Lab 6:

1. Get a string from the user and perform the following
 - (i) Take the last char and return a new string with the last char added at the front and back. ("bat""tbatt")
 - (ii) Return a new string where the first and last chars have been exchanged. ("bat""tab")

Source code:

(i)

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

```
public class last_char_in_front {
```

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

```
        System.out.print("Name      : Nithish G \nReg No. : 19BCS0012\n\n");
```

```
        Scanner obj=new Scanner (System.in);
```

```
        System.out.print("Enter the string : ");
```

```
        String a = obj.next();
```

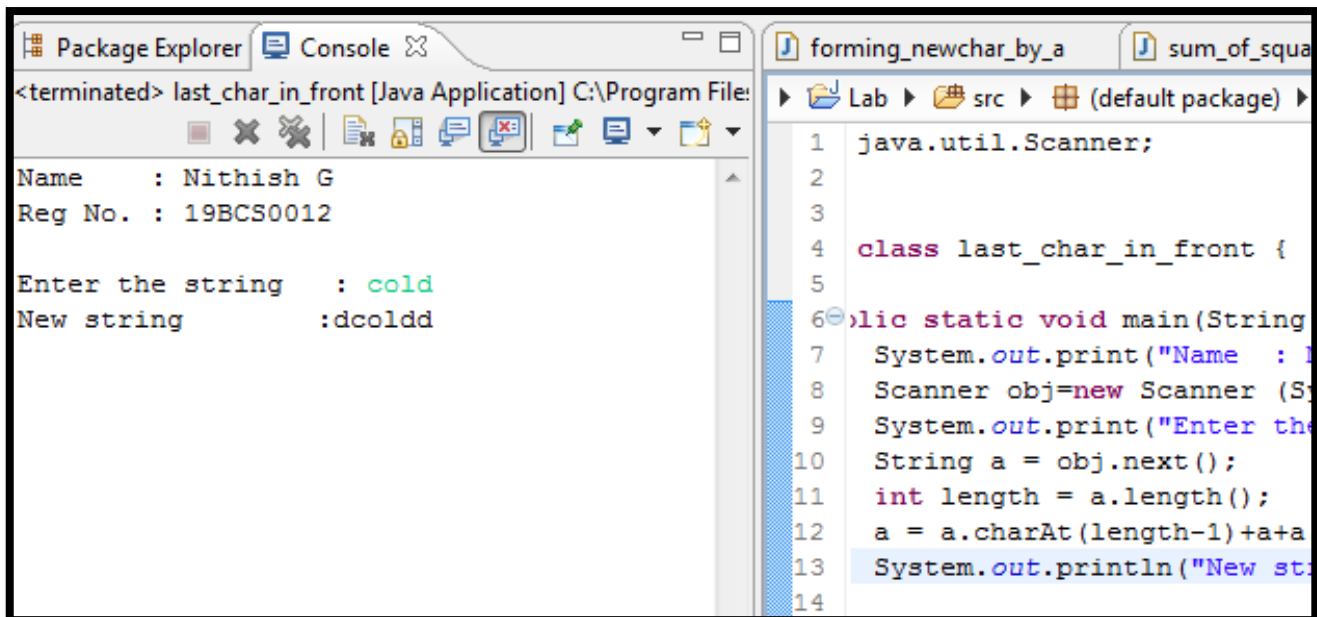
```
        int length = a.length();
```

```
        a = a.charAt(length-1)+a+a.charAt(length-1);
```

```
        System.out.println("New string      : " +a);
```

```
    }
```

} output:



The screenshot displays a Java IDE with two panels. The left panel, titled 'Console', shows the output of the program: 'Name : Nithish G', 'Reg No. : 19BCS0012', 'Enter the string : cold', and 'New string : dcoldd'. The right panel, titled 'forming_newchar_by_a', shows the source code for the 'last_char_in_front' class. The code imports 'java.util.Scanner', defines the class, and implements the 'main' method which prompts the user for a string and prints the result of the character manipulation logic.

```
Package Explorer Console
<terminated> last_char_in_front [Java Application] C:\Program File:
Name      : Nithish G
Reg No.   : 19BCS0012

Enter the string      : cold
New string           : dcoldd

forming_newchar_by_a sum_of_squa
1  java.util.Scanner;
2
3
4  class last_char_in_front {
5
6  public static void main(String
7      System.out.print("Name : 1
8      Scanner obj=new Scanner (S
9      System.out.print("Enter th
10     String a = obj.next();
11     int length = a.length();
12     a = a.charAt(length-1)+a+a
13     System.out.println("New st
14
```

(ii) Source code:

```
import java.util.Scanner;

public class forming_newchar_by_adding_lastchar_on_both_sides {

    public static void main(String[] args) {

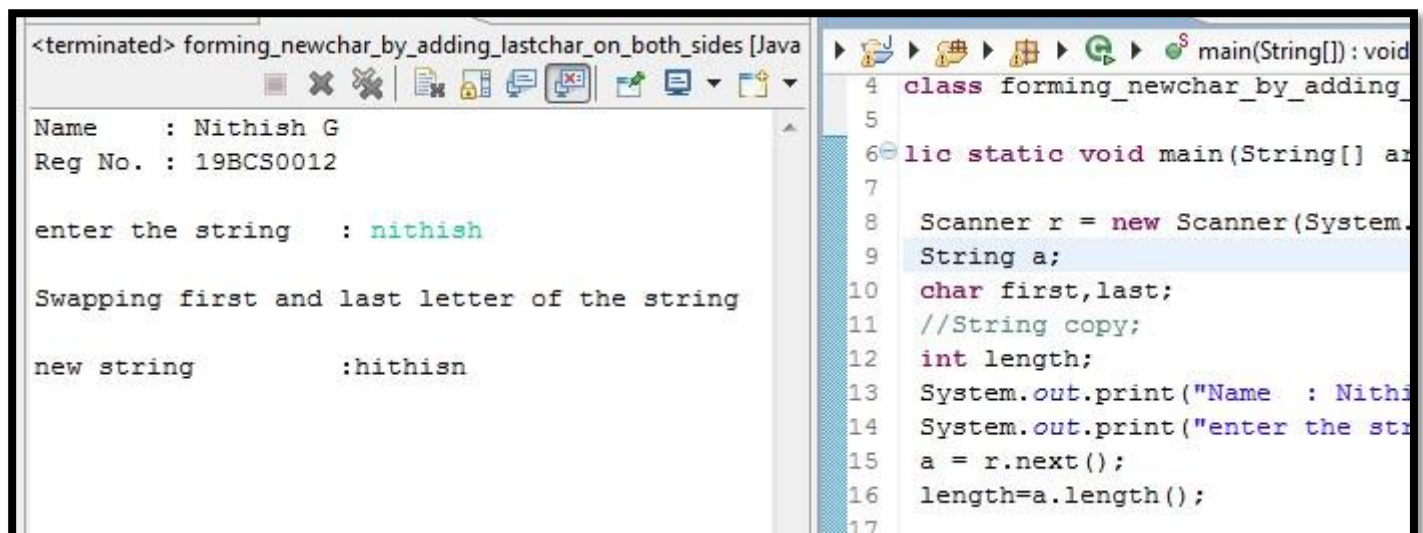
        Scanner r = new Scanner(System.in);
        String a;
        char first,last;

        int length;
        System.out.print("Name      : Nithish G \nReg No. : 19BCS0012\n\n");
        System.out.print("enter the string : ");
        a = r.next();
        length=a.length();

        char[]temp = a.toCharArray();
        first=a.charAt(0);
        last=a.charAt(length-1);
        temp[0] = last;
        temp[length-1]=first;
        a= String.valueOf(temp);
        System.out.println(a);

    }
}
```

(ii) Output



```
<terminated> forming_newchar_by_adding_lastchar_on_both_sides [Java]
Name      : Nithish G
Reg No.   : 19BCS0012

enter the string   : nithish

Swapping first and last letter of the string

new string        : hithisn

main(String[]) : void
4  class forming_newchar_by_adding_
5
6  public static void main(String[] a
7
8      Scanner r = new Scanner(System
9      String a;
10     char first,last;
11     //String copy;
12     int length;
13     System.out.print("Name  : Nithi
14     System.out.print("enter the st
15     a = r.next();
16     length=a.length();
17
```

2. Write a Java Program to sort the string in a given array.

Source Code:

```
import java.util.Arrays;

import java.util.Scanner;

public class sort_array_string {

    public static void main(String[] args) {

        Scanner obj = new Scanner (System.in);

        int n ;

        System.out.print("Name      : Nithish G \nReg No. : 19BCS0012\n\n");
        System.out.print("Enter the Size of array : ");

        n = obj.nextInt();

        String[] a = new String[n];

        for(int i = 0 ;i< n ; i++)

        {

            System.out.print("String "+(i+1)+" : ");

            a[i]=obj.next();

        }

        for(int i = 0; i<n; i++)

        {

            for (int j = i+1; j<n; j++)

            {

                if(a[i].compareTo(a[j])>0)

                {

                    String temp = a[i];
```

```

        a[i] = a[j];
        a[j] = temp;
    }
}

System.out.println("\nThe Sorted String Array \n");

for(int i = 0 ;i< n ; i++)
{
    System.out.println("String "+(i+1)+" : "+a[i]);
}
}
}

```

Output:

The screenshot displays an IDE with two panes. The left pane, titled 'Console', shows the execution of a Java application named 'sort_array_string'. It prompts the user to enter the size of the array (5) and then five strings: 'java', 'program', 'file', 'save', and 'completed'. The output shows the sorted array: 'completed', 'file', 'java', 'program', and 'save'. The right pane shows the source code of the 'sort_array_string.java' file, which includes a 'main' method that reads user input and sorts the array.

```

using_wrapper_class.  sort_array_string.java 23
sort_array_string ▶ main(String[]):
1 java.util.Arrays;
3
4 class sort_array_string {
5
6 public static void main(String[] args) {
7
8
9 Scanner obj = new Scanner (System.in);
10 int n ;
11 System.out.print("Name : Nithish G \nR
12 System.out.print("Enter the Size of arra
13 n = obj.nextInt();
14 String[] a = new String[n];
15 for(int i = 0 ;i< n ; i++) {
16 {
17     System.out.print("String "+(i+1)+"
18     a[i]=obj.next();
19 }
20 for(int i = 0; i<n; i++)
21 {

```

Package Explorer Console

<terminated> sort_array_string [Java Application] C:\

Name : Nithish G
Reg No. : 19BCS0012

Enter the Size of array : 5
String 1 : java
String 2 : program
String 3 : file
String 4 : save
String 5 : completed

The Sorted String Array

String 1 : completed
String 2 : file
String 3 : java
String 4 : program
String 5 : save

Lab 7:

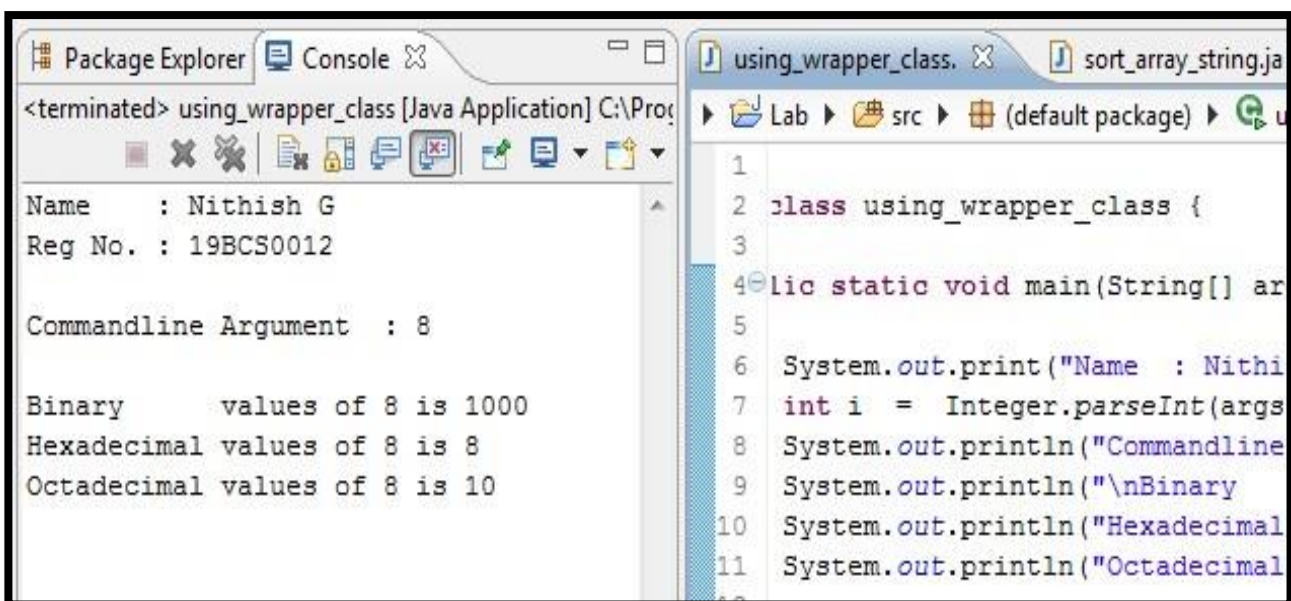
1. Write a program to receive an integer number as a command line argument, and print the binary, octal and hexadecimal equivalent of the given number using Wrapper Class.

Sample Output: java Test 20
Given Number :20
Binary equivalent :10100
Octal equivalent :24
Hexadecimal equivalent :14

Source Code:

```
public class using_wrapper_class {  
  
    public static void main(String[] args) {  
  
        System.out.print("Name      : Nithish G \nReg No. : 19BCS0012\n\n");  
        int i = Integer.parseInt(args[0]);  
        System.out.println("Commandline Argument : "+args[0]);  
        System.out.println("\nBinary    values of " +i+ " is " +Integer.toBinaryString(i));  
        System.out.println("Hexadecimal values of " +i+ " is " +Integer.toHexString(i));  
        System.out.println("Octadecimal values of " +i+ " is " +Integer.toOctalString(i));  
    }  
}
```

Output:



2. Write a Java code to find the distance from VIT University to major cities of India.

Hint: Create String array of major cities and integer array of distances. User gives the city name and the same is searched (use binary search) in the respective array and displays result.)

Source Code:

```
import java.util.Arrays;
import java.util.Scanner;

public class strign_binary_search {
    String city;
    int cities()
    {
        String[] cities = new String [10];
        cities[0]="agra";           cities[1]="ahmedabad";       cities[2]="bengaluru";
        cities[3]="chennai";        cities[4]="delhi";         cities[5]="calcutta";
        cities[6]="indore";         cities[7]="jaipur";       cities[8]="kanpur";
        cities[9]="lucknow";
        Arrays.sort(cities);

        System.out.print("\tName    : Nithish G \n\tReg No.      : 19BCS0012\n\n");
        System.out.println("\tFind the Distance from ");
        System.out.println("VIT University to major cities of India\n ");
        int min = 0,mid;
        int max = cities.length - 1;
        String key;
        Scanner obj = new Scanner (System.in);

        for(int i = 0 ;i<10 ;i++)
        {
            System.out.println("\t"+(i+1)+". "+(cities[i]).toUpperCase());
        }
        System.out.println("\n");
        System.out.print(" Enter the citie name :");
        key = obj.next();
        key = key.toLowerCase();
        while (min <= max) {
```

```

        mid = (min + max) / 2;
        if (cities[mid].compareTo(key) < 0)
        {
            min = mid + 1;
        }
        else if (cities[mid].compareTo(key) > 0)
        {
            max = mid - 1;
        }
        else
        {
            city=cities[mid];
            return mid;
        }
    }
    return -1;
}

```

```

public static void main(String[] args) {

```

```

    strign_binary_search obj = new strign_binary_search();

```

```

    int[] d = new int[10];

```

```

    d[0]= 1584;    d[1]= 1315;    d[2]= 211;
    d[3]= 137;     d[4]= 2222;    d[5]= 1750;
    d[6]= 1557;    d[7]= 2192;    d[8]= 1981;
    d[9]= 1975;

```

```

    int index = obj.cities();

```

```

    if(index!=-1)

```

```

        System.out.println("\nDistance from VIT UNIVERSITY To " + (obj.city).toUpperCase()+" :
"+d[index]+" Km.");

```

```

    else

```

```

        System.out.println("You Entered the wrong city name ");

```

```

    }

```

```

}

```

Output:

The screenshot shows a Java IDE with two windows. The left window, titled 'strign_binary_search [Java Application]', displays the program's output. It shows the user's name 'Nithish G' and registration number '19BCS0012'. The program prompts the user to find the distance from VIT University to major cities of India, listing 10 cities: AGRA, AHMEDABAD, BENGALURU, CALCUTTA, CHENNAI, DELHI, INDORE, JAIPUR, KANPUR, and LUCKNOW. The user enters 'Indore', and the program outputs 'Distance from VIT UNIVERSITY To INDORE : 1557 Km.'.

The right window, titled 'using_wrapper_class.', shows the source code for the program. It imports 'java.util.Arrays' and 'java.util.Scanner'. The class 'strign_binary_search' contains a 'city' variable and a 'cities()' method. The 'cities()' method initializes a 'String[] cities' array with 10 elements, sorts it using 'Arrays.sort(cities)', and prints the user's name and the task. It then prompts the user to enter a city name and starts a loop to find the distance.

```
<terminated> strign_binary_search [Java Application] C:\Program Files\Java\jdk1
Name      : Nithish G
Reg No.   : 19BCS0012

Find the Distance from
VIT University to major cities of India

1. AGRA
2. AHMEDABAD
3. BENGALURU
4. CALCUTTA
5. CHENNAI
6. DELHI
7. INDORE
8. JAIPUR
9. KANPUR
10. LUCKNOW

Enter the citie name   :Indore

Distance from VIT UNIVERSITY To INDORE : 1557 Km.

using_wrapper_class.
area.java
sort_array_st
Lab > src > (default package) > strign_bin
1 java.util.Arrays;
2 java.util.Scanner;
3
4
5 class strign_binary_search {
6     ng city;
7     cities()
8
9     String[] cities = new String [10];
10    cities[0]="agra";      cities[1]="ah
11    cities[3]="chennai";   cities[4]=
12    cities[6]="indore";    cities[7]="ja
13    cities[9]="lucknow";
14    Arrays.sort(cities);
15
16
17    System.out.print("\tName      : Nithish
18    System.out.println("\tFind the Distan
19    System.out.println("VIT University to
20        int min = 0,mid;
21        int max = cities.length - 1;
22        String key;
23        Scanner obj = new Scanner (System
24
25        for(int i = 0 ;i<10 ;i++)
26        {
27            System.out.println("\t"+(
```

3. Consider an int array first with possibly repeated values given by the user. Create a new array second that has each number in the first appear exactly once in their order of appearance. Display the second array.
For example, if values in first are 10, 20, 6, 7, 10, 8, 5, 6, 4, 7, 1, then the second has 10, 20, 6, 7, 8, 5, 4, 1.

Source code:

```
import java.util.Scanner;

public class store_1st_appearance_to_array {

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

```

int i,j=0,count=0,k=0,present=0,n=0,next = 0,increment=0;
    System.out.print("\tName : Nithish G \n\tReg No.      : 19BCS0012\n\n");
    Scanner obj = new Scanner(System.in);
    System.out.print("Enter the Array Size : ");
    n = obj.nextInt();
    int[] a = new int[n];
    int[] b = new int[n];

    System.out.println("\nEnter the Array elements \n");
    for (i=0;i<n;i++)
    {
        System.out.print("\t" + (i+1) + ". ");
        a[i]= obj.nextInt();
    }
    System.out.println();
    do
    {
        for (i=0;i<=next;i++)
        {
            if(b[i]==a[j])
            {
                present++;
            }
        }
        if(present==0)
        {
            b[k]=a[j];
            k++;
            next++;
            count++;
        }
        if(present>0)
        {
            present=0;
        }
        j++;
        increment++;
    }

```

```

    }while(increment<n);

    System.out.println("First appear exactly once in their order of appearance\n");
    for(i=0;i<count;i++)
    {
        System.out.println("\t"+(i+1)+". "+b[i]);
    }

}
}

```

Output:

The screenshot displays a Java IDE with two panels. The left panel shows the output of a Java application, and the right panel shows the corresponding source code.

Source Code (Right Panel):

```

9  int i,j=0,count=0,k=0,present=0,n=0,next =
10 System.out.print("\tName      : Nithish G \n
11 Scanner obj = new Scanner(System.in);
12 System.out.print("Enter the Array Size :
13 n = obj.nextInt();
14 int[] a = new int[n];
15 int[] b = new int[n];
16
17 System.out.println("\nEnter the Array elem
18 for (i=0;i<n;i++)
19 {
20     System.out.print("\t"+(i+1)+". ");
21     a[i]= obj.nextInt();
22 }
23 out.println();
24 do
25 {
26     for (i=0;i<=next;i++)
27     {
28         if(b[i]==a[j])
29         {
30             present++;
31         }
32     }
33     if(present==0)
34     {
35
36         b[k]=a[j];

```

Output (Left Panel):

```

<terminated> store_1st_appearance_to_array [Java Application] C:\
Name      : Nithish G
Reg No.   : 19BCS0012

Enter the Array Size : 8

Enter the Array elements

1. 43
2. 77
3. 65
4. 43
5. 87
6. 77
7. 65
8. 54

5 Elements First appear exactly
once in their order of appearance

1. 43
2. 77
3. 65
4. 87
5. 54

```


LAB – 8

1. Create a class named Employee. Include data fields to hold the Employee's first name, last name, and hourly pay rate. Create an array of five Employee objects. Prompt the user to enter data for each Employee. Then prompt the user for the number of an Employee to view (1 through 5), and display the corresponding Employee's data.

SOURCE CODE:

```
import java.util.Scanner;

public class display_particular_employee {
    String First_name;
    String Last_name;
    int hourly_pay;

    public static void main(String[] args) {

        display_particular_employee em[] = new
display_particular_employee[5];
        Scanner read = new Scanner (System.in);

        int i=0;
        for(i=0;i<5;i++)
        {
            em[i]=new display_particular_employee();
System.out.println("          Employee : "+ (i+1) + "\n");

System.out.print("Employee First name          : ");
em[i].First_name = read.next();
System.out.print("Employee Last name          : ");
em[i].Last_name = read.next();
System.out.print("Enter the employee hourly pay : ");
    em[i].hourly_pay =read.nextInt();
        System.out.println();
        }
System.out.print("Enter 1 to 5 to show the details of
Employee:");
        int show=0;
        show = read.nextInt();
    }
```

```

        show--;
System.out.println("Employee First name          : "+
em[show].First_name);
System.out.println("Employee Last name          : " +
em[show].Last_name);
System.out.println("Enter the employee hourly pay :
"+em[show].hourly_pay);

    } }

```

OUTPUT:

The screenshot displays a Java IDE with two panels. The left panel shows the console output of a Java application, and the right panel shows the source code of the application.

Console Output:

```

<terminated> display_particular_employee [Java Application] C:\Program Files\
Name : Nithish G
Reg No.: 19BCS0012

Employee : 1

Employee First name : vikram
Employee Last name : vedha
Enter the employee hourly pay : 600

Employee : 2

Employee First name : nithish
Employee Last name : g
Enter the employee hourly pay : 1000

Employee : 3

Employee First name : rajesh
Employee Last name : kumar
Enter the employee hourly pay : 450

Employee : 4

Employee First name : karthi
Employee Last name : raj
Enter the employee hourly pay : 550

```

Source Code:

```

1  .util.Scanner;
2
3
4  s display_particular_employee {
5  First_name;
6  Last_name;
7  rly_pay;
8
9  static void main(String[] args) {
10 em.out.print("\t Name : Nithish G
11 play_particular_employee em[] = new d
12 inner read = new Scanner (System.in);
13
14 i=0;
15 (i=0;i<5;i++)
16
17 em[i]=new display_particular_employee
18 System.out.println(" Emp
19
20 System.out.print("Employee First name
21 em[i].First_name = read.next();
22 System.out.print("Employee Last name
23 em[i].Last_name = read.next();
24 System.out.print("Enter the employee
25 em[i].hourly_pay =read.nextInt();
26 System.out.println();
27

```

The screenshot shows an IDE with a Java program running. The left pane displays the output of the program, and the right pane shows the source code.

Output (Left Pane):

```
Employee : 4
Employee First name      : karthi
Employee Last name       : raj
Enter the employee hourly pay : 550

Employee : 5
Employee First name      : ramesh
Employee Last name       : kumar
Enter the employee hourly pay : 550

Enter 1 to 5 to show the details of Employee:2
Employee First name      : nithish
Employee Last name       : g
Enter the employee hourly pay : 1000
```

Source Code (Right Pane):

```
15 (l=0;l<5;l++)
16
17 em[i]=new display_particular_employee();
18 System.out.println("      Employee : "+ (i+1) + "\n");
19
20 System.out.print("Employee First name      : ");
21 em[i].First_name = read.next();
22 System.out.print("Employee Last name       : ");
23 em[i].Last_name = read.next();
24 System.out.print("Enter the employee hourly pay : ");
25 em[i].hourly_pay =read.nextInt();
26 System.out.println();
27
28
```

IDE Interface:

- Buttons: Problems, Javadoc, Declaration
- Status: 0 errors, 10 warnings, 0 others
- Table with columns: Description, Resource, Path
- Warning: 10 items

2. Create a class named Employee. Include data fields to hold the Employee's ID number, first name, last name, and hourly pay rate. Create an array of five Employee objects. Prompt the user to enter data for each Employee. Do not allow duplicate ID numbers to be entered. Then prompt the user to choose whether to search for an Employee by (1) ID number, (2) last name, or (3) hourly pay. After the user chooses the field on which to search, prompt the user for the search value. Display an error message if there is no Employee with matching criteria, otherwise display all the data for every matching Employee (more than one Employee might have the same last name or pay rate)

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

```
public class employee {
```

```
    String First_name;
```

```
    String Last_name;
```

```
    int hourly_pay,Id;
```

```

public static void main(String[] args) {

    employee em[] = new employee[5];
    Scanner read = new Scanner (System.in);
    System.out.print("\t Name      : Nithish G \n\t Reg No.:
19BCS0012\n\n");
    int i=0,c =0;
    int s = 0;
    for(i=0;i<5;i++)
    {

        em[i]=new employee();
        System.out.println("      Employee : " + (i+1) + "\n");

        System.out.print("Employee ID      : ");
        em[i].Id = read.nextInt();

        System.out.print("Employee First name      : ");
        em[i].First_name = read.next();
        System.out.print("Employee Last name      : ");
        em[i].Last_name = read.next();
        System.out.print("Enter the employee hourly pay : ");
        em[i].hourly_pay =read.nextInt();
        for(c=0;c<i;c++)
        {
            if(em[i].Id==em[c].Id)
            {
                System.out.println("ID is repated Plese enter unique id
again");

                i--;
            }

        }
        System.out.println("\n");
    }
    int count =1;
    for(int r = 0; r<count; r++)
    {
        System.out.println("Choose the option to search the employee by ");
    }
}

```

```

        System.out.println("1. Employee ID\n2. Employee Last
Name\n3.Employee hourly pay");
        System.out.print("choice : ");
        int ch=0;
        ch = read.nextInt();

        switch(ch)
        {
        case 1 :

                System.out.print("Enter the Employee ID : ");
                int sId = read.nextInt();

                for(i=0;i<5;i++)
                {
                        if(sId==em[i].Id)
                        {
                                System.out.println("\nThis Employee ID belong's to ");
                                System.out.println("Employee ID      : "+em[i].Id);
                                System.out.println("Employee First name   : "+em[i].First_name);
                                System.out.println("Employee Last name    : "+em[i].Last_name);
                                System.out.println("Employee Hourly pay   : "+em[i].hourly_pay);
                                s=1;
                        }
                }
                if(s==0)
                {
                        System.out.println(" The data is not found ");
                }
                break;

        case 2:

                System.out.print("Enter the Employee Last name : ");
                String sln = read.next();
                for(i=0;i<5;i++)
                {
                        if(sln.equals(em[i].Last_name))
                        {
                                System.out.println("\nThis Employee Last name
belongs to ");

```

```

System.out.println("Employee First name  : "+em[i].First_name);
System.out.println("Employee Last name   : "+em[i].Last_name);
System.out.println("Employee Hour pay    : "+em[i].hourly_pay);
                s=1;
            }
        }
        if(s==0)
        {
            System.out.println(" \nThe data is not found ");
        }

        break;

    case 3:

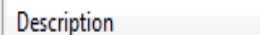
        System.out.print("\nEnter the Employee hourly pay : ");
        int shp = read.nextInt();
        for(i=0;i<5;i++)
        {
            if(shp ==em[i].hourly_pay)
            {
System.out.println("\nThis Employee Hourly belongs to ");
System.out.println("Employee First name  : "+em[i].First_name);
System.out.println("Employee Last name   : "+em[i].Last_name);
System.out.println("Employee Hour pay    : "+em[i].hourly_pay);
                s=1;
            }
        }
        if(s==0)
        {
            System.out.println(" \nThe data is not found ");
        }
        break;
    default:
        System.out.print("\nPlease enter the correct choice :");

    }

    System.out.println("\n Press 1 to continue else press 0 to stop");
    System.out.print("choice : ");

```

OUTPUT:



Package Explorer
Console X

employee [Java Application] C:\Program Files\Java\jdk1.7.0_76\bin\javaw.exe (2)

Employee : 3

Employee ID : 13
Employee First name : rajesh
Employee Last name : kumar
Enter the employee hourly pay : 450
ID is repated Plese enter unique id again

Employee : 3

Employee ID : 14
Employee First name : rajesh
Employee Last name : kumar
Enter the employee hourly pay : 450

Employee : 4

Employee ID : 15
Employee First name : karthi
Employee Last name : raj
Enter the employee hourly pay : 600

Employee : 5

Employee ID : 16
Employee First name : yash
Employee Last name : kumar
Enter the employee hourly pay : 800

wish_based_on_presen
display_particular_e

Lab ▶ src ▶ (default package) ▶ employee

```

1  a.util.Scanner;
2
3  ss employee {
4
5      First_name;
6      Last_name;
7      urly_pay,Id;
8
9  static void main(String[] args) {
10
11      ployee em[] = new employee[5];
12      anner read = new Scanner (System.in)
13      stem.out.print("\t Name : Nithish
14      t i=0,c =0;
15      t s = 0;
16      r(i=0;i<5;i++)
17
18
19      em[i]=new employee();
20      System.out.println("
21
22      System.out.print("Employee ID
23      em[i].Id = read.nextInt();
24
25
26      System.out.print("Employee First n
27      em[i].First_name = read.next();
28      System.out.print("Employee Last na

```

Problems X
@ Javadoc
Declaration

0 errors, 10 warnings, 0 others

Description

Package Explorer

Console

employee [Java Application] C:\Program Files\Java\jdk1.7.0_76\bin\javaw.exe (2)

Employee : 5

Employee ID : 16
Employee First name : yash
Employee Last name : kumar
Enter the employee hourly pay : 800

Choose the option to search the employee by
1. Employee ID
2. Employee Last Name
3. Employee hourly pay
choice : 1
Enter the Employee ID : 12

This Employee ID belong's to
Employee ID : 12
Employee First name : nithish
Employee Last name : g
Employee Hourly pay : 1000

Press 1 to continue else press 0 to stop
choice : 1
Choose the option to search the employee by
1. Employee ID
2. Employee Last Name
3. Employee hourly pay
choice : 2
Enter the Employee Last name : kumar

wish_based_on_presen

display_par

Lab > src > (default package)

```
1  a.util.Scanner;  
2  
3  ss employee {  
4  
5      First_name;  
6      Last_name;  
7      urly_pay,Id;  
8  
9  static void main(String[]  
10  
11  ployee em[] = new employee  
12  anner read = new Scanner  
13  stem.out.print("\t Name  
14  t i=0,c =0;  
15  t s = 0;  
16  r(i=0;i<5;i++)  
17  
18  
19      em[i]=new employee();  
20      System.out.println("  
21  
22      System.out.print("Employ  
23      em[i].Id = read.nextInt(  
24  
25  
26      System.out.print("Employ  
27      em[i].First_name = read.  
28      System.out.print("Empl
```

Problems @ Javadoc Declarati

Package Explorer Console X

employee [Java Application] C:\Program Files\Java\jdk1.7.0_76\bin\javaw.exe (2

2. Employee Last Name
3. Employee hourly pay
choice : 2
Enter the Employee Last name : kumar

This Employee Last name belongs to
Employee First name : rajesh
Employee Last name : kumar
Employee Hour pay : 450

This Employee Last name belongs to
Employee First name : yash
Employee Last name : kumar
Employee Hour pay : 800

Press 1 to continue else press 0 to stop
choice : 1
Choose the option to search the employee by
1. Employee ID
2. Employee Last Name
3. Employee hourly pay
choice : 3
Enter the Employee hourly pay : 1000

This Employee Hourly belongs to |
Employee First name : nithish
Employee Last name : g
Employee Hour pay : 1000

Press 1 to continue else press 0 to stop
choice :

wish_based_on_presen display_particular_e

Lab > src > (default package) > employee

```
1 a.util.Scanner;  
2  
3 ss employee {  
4  
5     First_name;  
6     Last_name;  
7     urly_pay,Id;  
8  
9 static void main(String[] args) {  
10  
11     ployee em[] = new employee[5];  
12     anner read = new Scanner (System.in);  
13     stem.out.print("\t Name : Nithish  
14     t i=0,c =0;  
15     t s = 0;  
16     r(i=0;i<5;i++)  
17  
18  
19     em[i]=new employee();  
20     System.out.println("      Em  
21  
22     System.out.print("Employee ID  
23     em[i].Id = read.nextInt();  
24  
25  
26     System.out.print("Employee First na  
27     em[i].First_name = read.next();  
28     System.out.print("Employee Last nam
```

Problems X @ Javadoc Declaration

0 errors, 10 warnings, 0 others

Description