

CSE TRAINING-II LABORATORY

TR-102



Submitted by: Guneet Kohli

D3 CSEA1

U.R.No.: 1805172

C.R.No.: 1815017

INDEX

S.No	Program
1.	Program to swap two numbers input by the user without using third variable.
2.	Program to print the sum of digits of a number entered by the user.
3.	Program to print Fibonacci series up to first ten terms.
4.	Program to print the reverse of the number entered by the user.
5.	Program to find the largest of three numbers using conditional statement.
6.	Program to create a class and call member functions.
7.	Program to implement employee class with data members i.e. name and salary and member functions i.e. getdata and showdata.
8.	Program to enter elements in array and display the array.
9.	Program to show implicit and explicit type casting.
10.	Program to demonstrate the use of static variables and static methods.
11.	Program to show the use of 'final' keyword.
12.	Program to illustrate the concept of default and parameterized constructors.
13.	Program to illustrate the concept of 'this' pointer.
14.	Program to illustrate the use 'super' keyword in inheritance.
15.	Program to create an interface and implement it using class.
16.	Program to illustrate the concept of packages.
17.	Program to demonstrate the concept of exception handling.
18.	Program to demonstrate the concept of inheritance.
19.	Program to create an abstract class.
20.	Program to demonstrate the concept of method overloading.

PROGRAM-1

Program to swap two numbers input by the user without using third variable.

```
public class swap {  
    public void main(String args[]){  
        System.out.println("before swapping");  
        int x=15;  
        int y=20;  
        System.out.println("value of x is:" +x);  
        System.out.println("value of y is:" +y);  
        System.out.println("after swapping");  
        x=x+y;  
        y=x-y;  
        x=x-y;  
        System.out.println("value of x is:" +x);  
        System.out.println("value of y is:" +y);  
    }  
}
```

The screenshot displays an IDE window titled 'swap - MyProject'. The code editor shows the Java program for swapping two numbers without a third variable. The code is as follows:

```
public class swap {  
    public void main(String args[]){  
        System.out.println("before swapping");  
        int x=15;  
        int y=20;  
        System.out.println("value of x is:" +x);  
        System.out.println("value of y is:" +y);  
        System.out.println("after swapping");  
        x=x+y;  
        y=x-y;  
        x=x-y;  
        System.out.println("value of x is:" +x);  
        System.out.println("value of y is:" +y);  
    }  
}
```

Below the code editor, a 'BlueJ: Terminal Window - MyProject' is open, showing the output of the program:

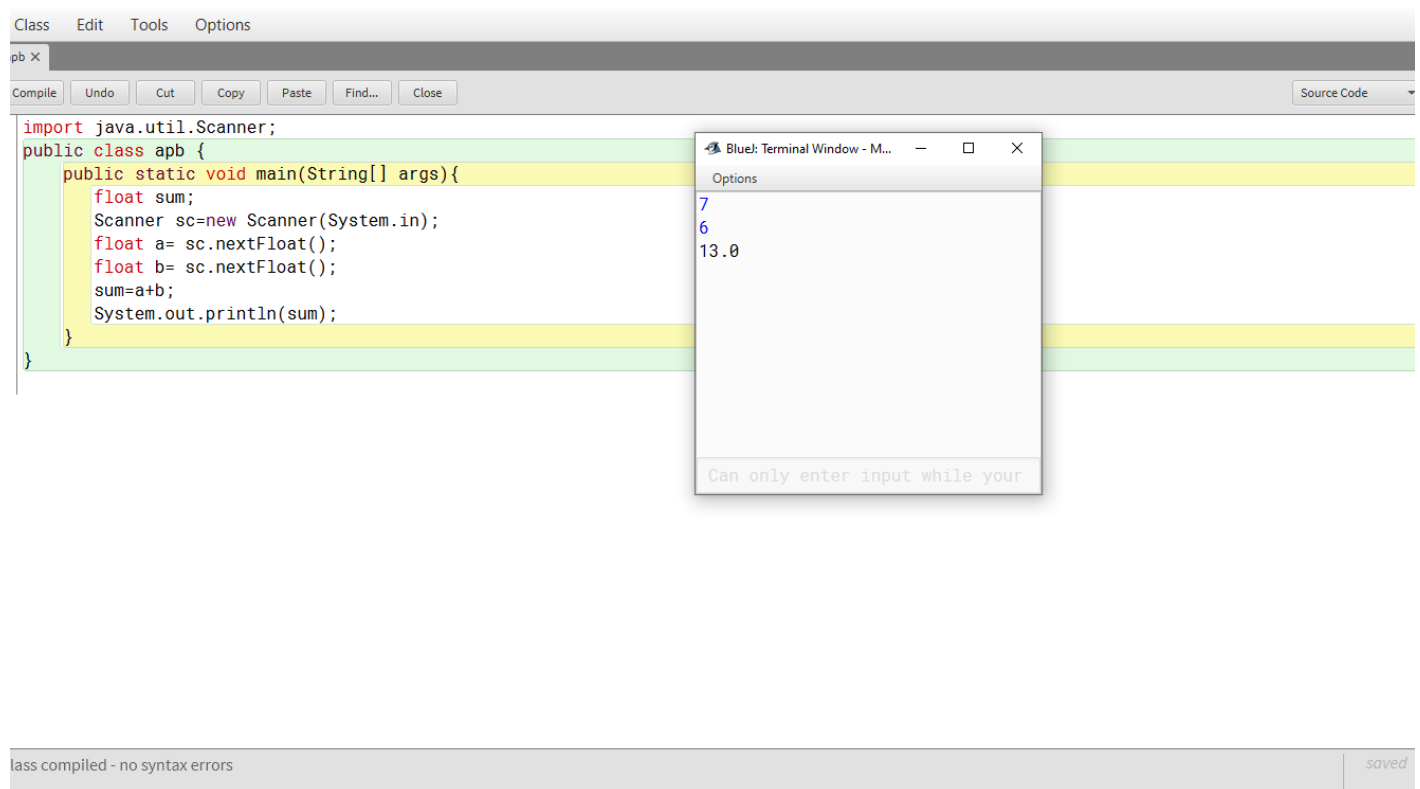
```
before swapping  
value of x is:15  
value of y is:20  
after swapping  
value of x is:20  
value of y is:15
```

At the bottom of the IDE, a status bar indicates 'Class compiled - no syntax errors' and 'saved'.

PROGRAM-2

Program to print the sum of digits of a number entered by the user.

```
import java.util.Scanner;
public class apb {
    public void main(String[] args){
        float sum;
        Scanner sc=new Scanner(System.in);
        float a= sc.nextFloat();
        float b= sc.nextFloat();
        sum=a+b;
        System.out.println(sum);
    }
}
```



PROGRAM-3

Program to print Fibonacci series up to first ten terms.

```
public class fib {  
  
    public static void main(String[] args) {  
  
        int n = 10, t1 = 0, t2 = 1, sum;  
  
        System.out.print("First " + n + " terms: ");  
  
        for (int i = 1; i <= n; ++i) {  
  
            System.out.print(t1 + " + ");  
  
            sum = t1 + t2;  
  
            t1 = t2;  
  
            t2 = sum; }  
  
    }  
  
}
```



The screenshot shows a Java IDE with a file named 'fib.x'. The code is as follows:

```
public class fib {  
    public static void main(String[] args) {  
        int n = 10, t1 = 0, t2 = 1, sum;  
        System.out.print("First " + n + " terms: ");  
        for (int i = 1; i <= n; i++)  
        {  
            System.out.print(t1 + " + ");  
  
            sum = t1 + t2;  
            t1 = t2;  
            t2 = sum;  
        }  
    }  
}
```

A terminal window titled 'Blue: Terminal Window - MyProject' is open, showing the output: 'First 10 terms: 0 + 1 + 1 + 2 + 3 + 5 + 8 + 13 + 21 + 34'. A message at the bottom of the terminal says 'Can only enter input while your programming is running'.

PROGRAM-4

Program to print the reverse of the number entered by the user.

```
import java.util.Scanner;

public class Rev {

    public static void main(String args[]) {

        int n, reverse = 0;

        System.out.println("Enter an integer to reverse");

        Scanner in = new Scanner(System.in);

        n = in.nextInt();

        while(n != 0){

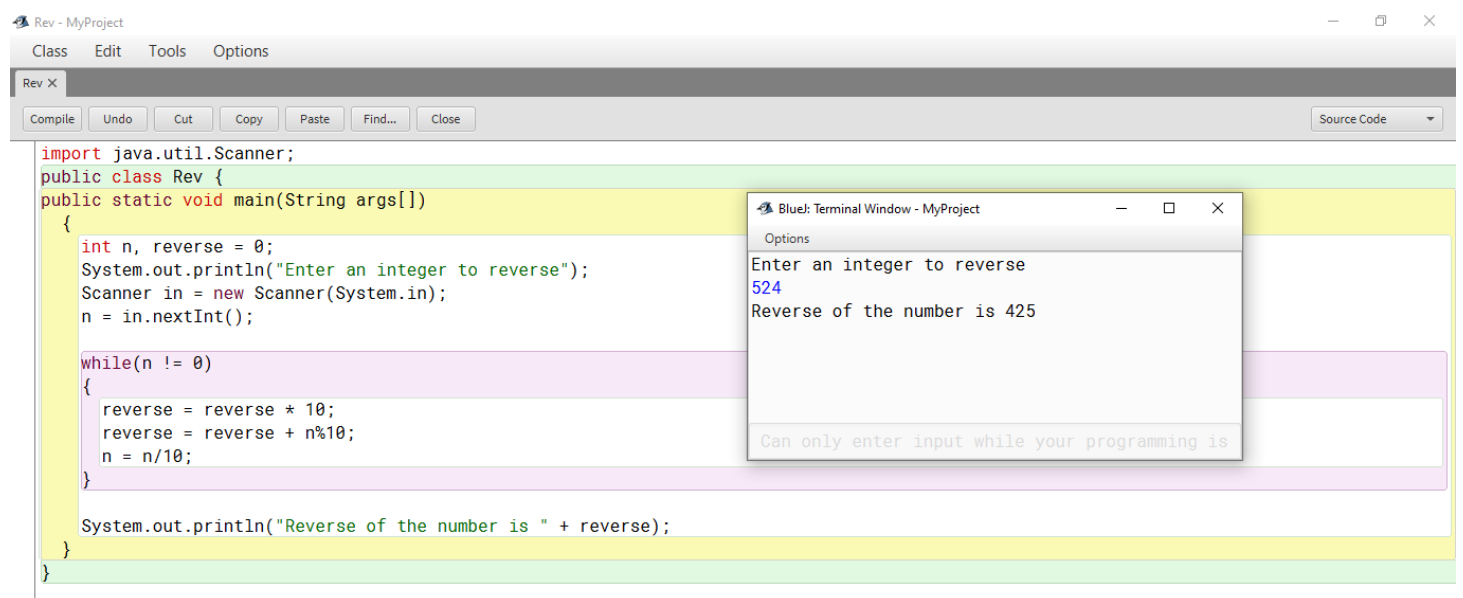
            reverse = reverse * 10;

            reverse = reverse + n%10;

            n = n/10;}

        System.out.println("Reverse of the number is " + reverse);}

}
```



The screenshot shows a Java IDE window titled "Rev - MyProject". The code editor displays the same Java code as shown in the previous block. A terminal window titled "Blue: Terminal Window - MyProject" is open, showing the execution output. The prompt "Enter an integer to reverse" is followed by the input "524". The output then shows "Reverse of the number is 425". A message at the bottom of the terminal says "Can only enter input while your programming is". The IDE interface includes a menu bar (Class, Edit, Tools, Options) and a toolbar with buttons for Compile, Undo, Cut, Copy, Paste, Find..., and Close. The status bar at the bottom indicates "Class compiled - no syntax errors" and "saved".

```
import java.util.Scanner;
public class Rev {
    public static void main(String args[])
    {
        int n, reverse = 0;
        System.out.println("Enter an integer to reverse");
        Scanner in = new Scanner(System.in);
        n = in.nextInt();

        while(n != 0)
        {
            reverse = reverse * 10;
            reverse = reverse + n%10;
            n = n/10;
        }

        System.out.println("Reverse of the number is " + reverse);
    }
}
```

Blue: Terminal Window - MyProject

Options

Enter an integer to reverse

524

Reverse of the number is 425

Can only enter input while your programming is

Class compiled - no syntax errors | saved

PROGRAM-5

Program to find the largest of three numbers using conditional statement.

```
import java.util.Scanner;

public class LargestTernary {

    public static void main(String[] args) {

        int a, b, c, d;

        Scanner s = new Scanner(System.in);

        System.out.println("Enter all three numbers:");

        a = s.nextInt();

        b = s.nextInt();

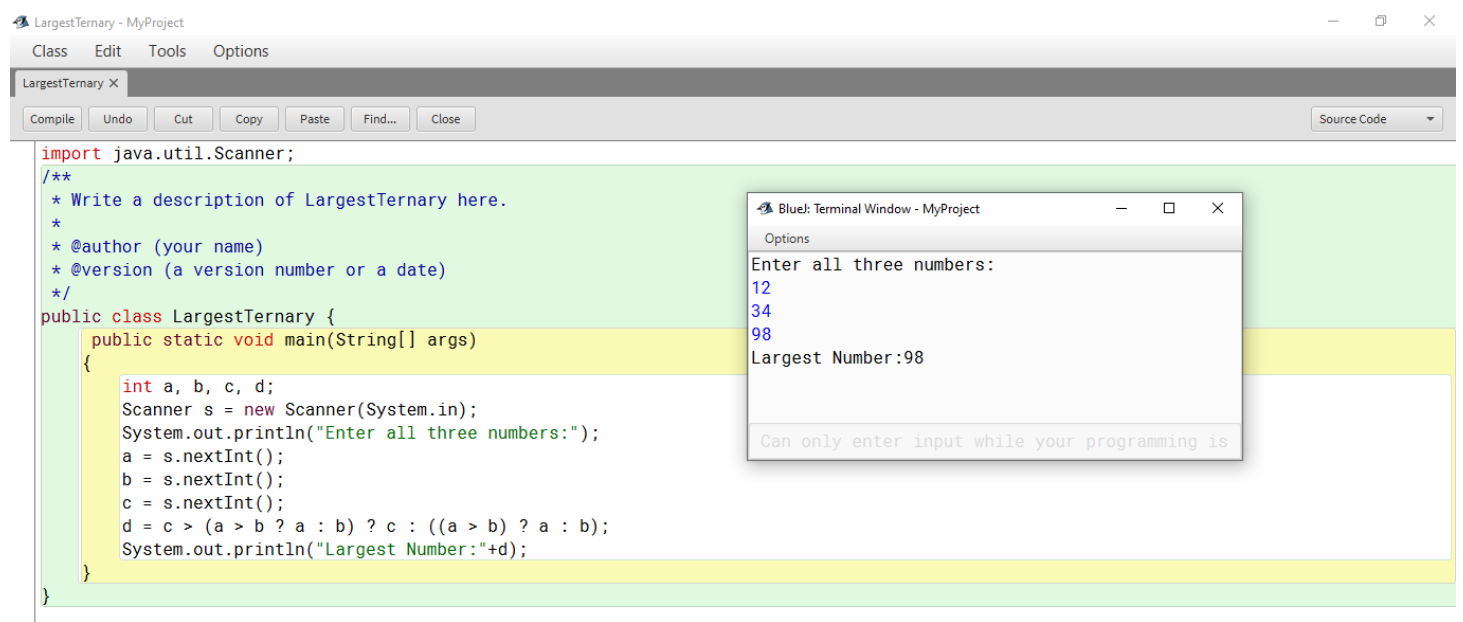
        c = s.nextInt();

        d = c > (a > b ? a : b) ? c : ((a > b) ? a : b);

        System.out.println("Largest Number:"+d);

    }

}
```



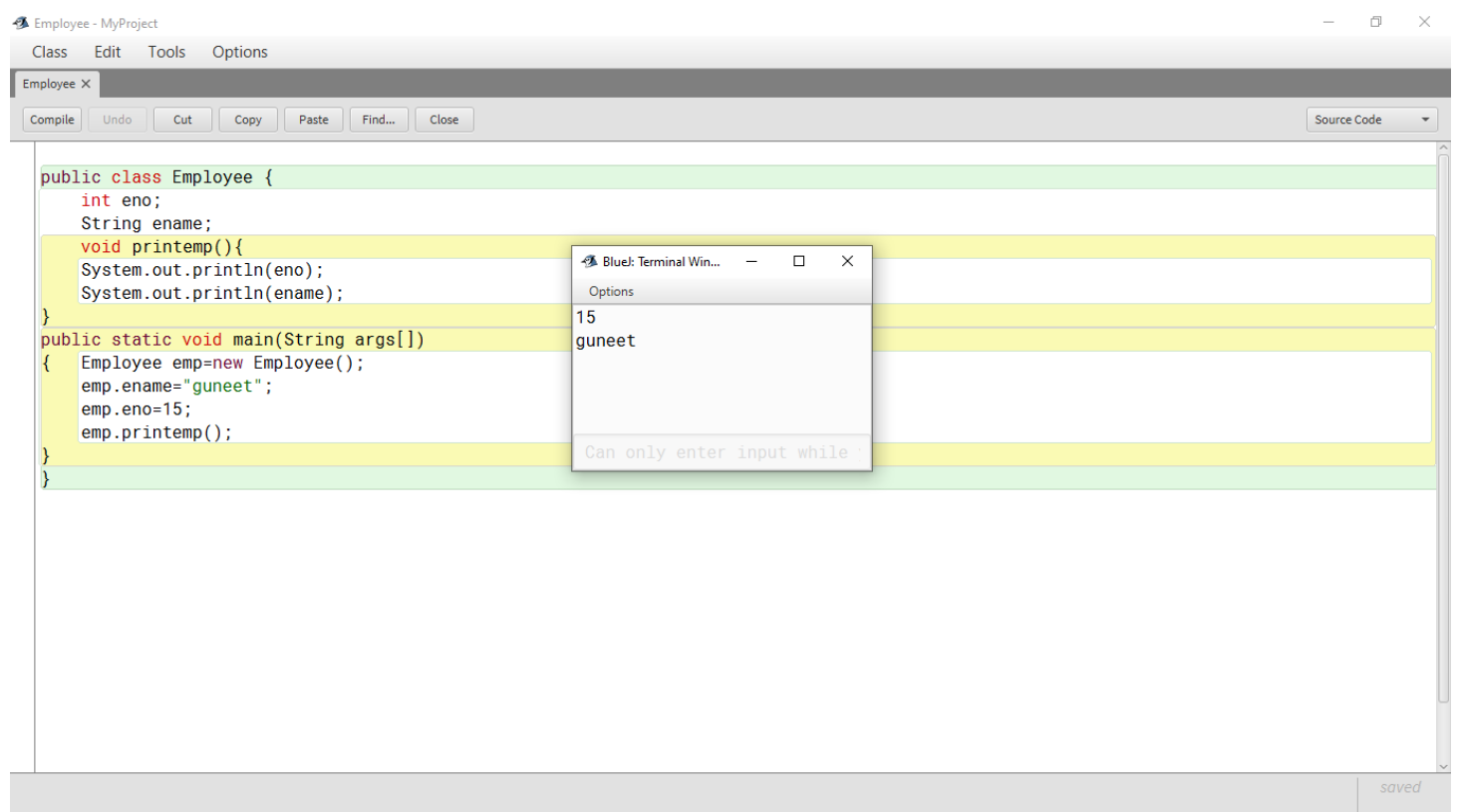
```
import java.util.Scanner;

/**
 * Write a description of LargestTernary here.
 *
 * @author (your name)
 * @version (a version number or a date)
 */
public class LargestTernary {
    public static void main(String[] args)
    {
        int a, b, c, d;
        Scanner s = new Scanner(System.in);
        System.out.println("Enter all three numbers:");
        a = s.nextInt();
        b = s.nextInt();
        c = s.nextInt();
        d = c > (a > b ? a : b) ? c : ((a > b) ? a : b);
        System.out.println("Largest Number:"+d);
    }
}
```

PROGRAM-6

Program to create a class and call member functions.

```
public class Employee {  
  
    int eno;  
  
    String ename;  
  
    void printemp(){  
  
        System.out.println(enno);  
  
        System.out.println(ename);  
  
    }  
  
    public static void main(String args[]) {  
  
        Employee emp=new Employee();  
  
        emp.ename="guneet";  
  
        emp.eno=15;  
  
        emp.printemp();}  
  
}
```



The screenshot displays an IDE window titled "Employee - MyProject". The code editor shows the following Java code:

```
public class Employee {  
    int eno;  
    String ename;  
    void printemp(){  
        System.out.println(enno);  
        System.out.println(ename);  
    }  
    public static void main(String args[])  
    {  
        Employee emp=new Employee();  
        emp.ename="guneet";  
        emp.eno=15;  
        emp.printemp();  
    }  
}
```

Below the code editor, a terminal window titled "Blue: Terminal Win..." is open, showing the output of the program:

```
15  
guneet
```

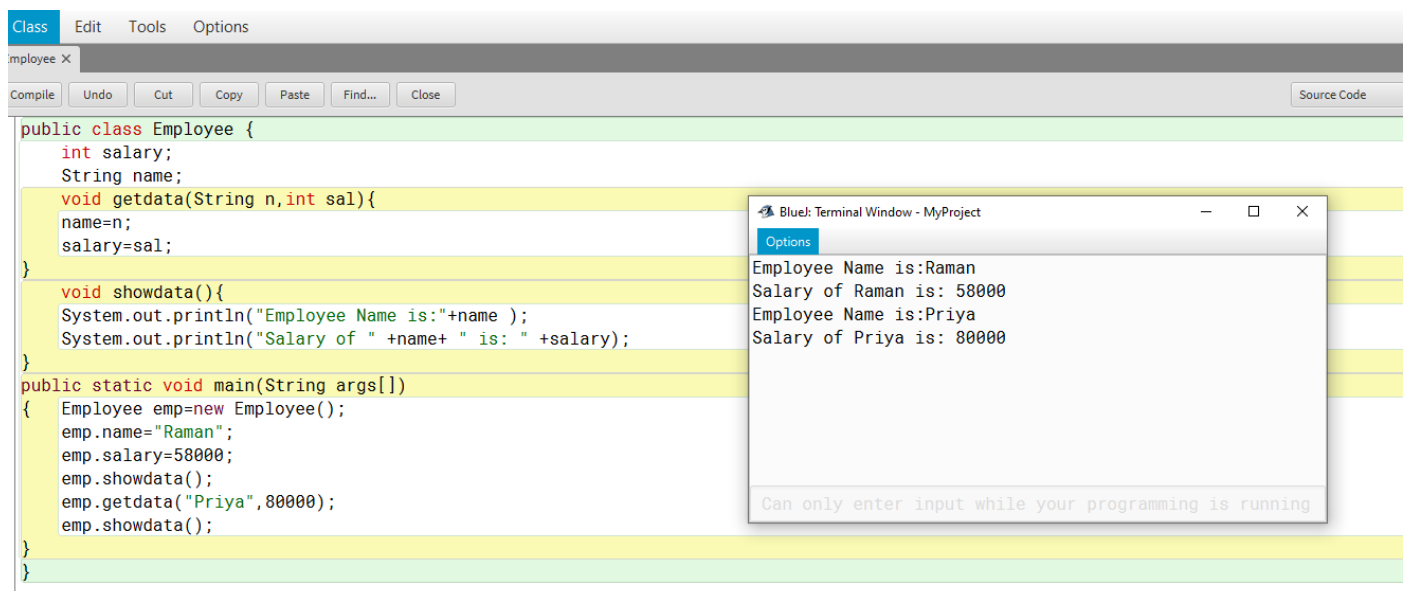
The terminal window also displays the message "Can only enter input while".

Guneet Kohli 1805172

PROGRAM-7

Program to implement employee class with data members i.e. name and salary and member functions i.e. getdata and showdata.

```
public class Employee {  
  
    int salary;  
  
    String name;  
  
    void getdata(String n,int sal){  
  
        name=n;  
  
        salary=sal; }  
  
    void showdata(){  
  
        System.out.println("Employee Name is:"+name );  
  
        System.out.println("Salary of " +name+ " is: " +salary); }  
  
    public static void main(String args[]){  
  
        Employee emp=new Employee();  
  
        emp.name="Raman";  
  
        emp.salary=58000;  
  
        emp.showdata();  
  
        emp.getdata("Priya",80000);  
  
        emp.showdata(); }  
  
}
```



The screenshot shows an IDE window titled 'employee X' with a menu bar (Class, Edit, Tools, Options) and a toolbar (Compile, Undo, Cut, Copy, Paste, Find..., Close). The code is displayed with syntax highlighting. A 'Terminal Window - MyProject' is open, showing the output of the program. Below the terminal, a message states 'Can only enter input while your programming is running'.

```
public class Employee {  
    int salary;  
    String name;  
    void getdata(String n,int sal){  
        name=n;  
        salary=sal;  
    }  
    void showdata(){  
        System.out.println("Employee Name is:"+name );  
        System.out.println("Salary of " +name+ " is: " +salary);  
    }  
    public static void main(String args[])  
    {  
        Employee emp=new Employee();  
        emp.name="Raman";  
        emp.salary=58000;  
        emp.showdata();  
        emp.getdata("Priya",80000);  
        emp.showdata();  
    }  
}
```

Blue: Terminal Window - MyProject

Options

Employee Name is:Raman
Salary of Raman is: 58000
Employee Name is:Priya
Salary of Priya is: 80000

Can only enter input while your programming is running

PROGRAM-8

Program to enter elements in array and display the array.

```
import java.util.Scanner;

public class ArraySum {

    public static void main(String[] args) {

        int n, sum = 0;

        Scanner s = new Scanner(System.in);

        System.out.print("Enter no. of elements you want in array:");

        n = s.nextInt();

        int a[] = new int[n];

        System.out.println("Enter all the elements:");

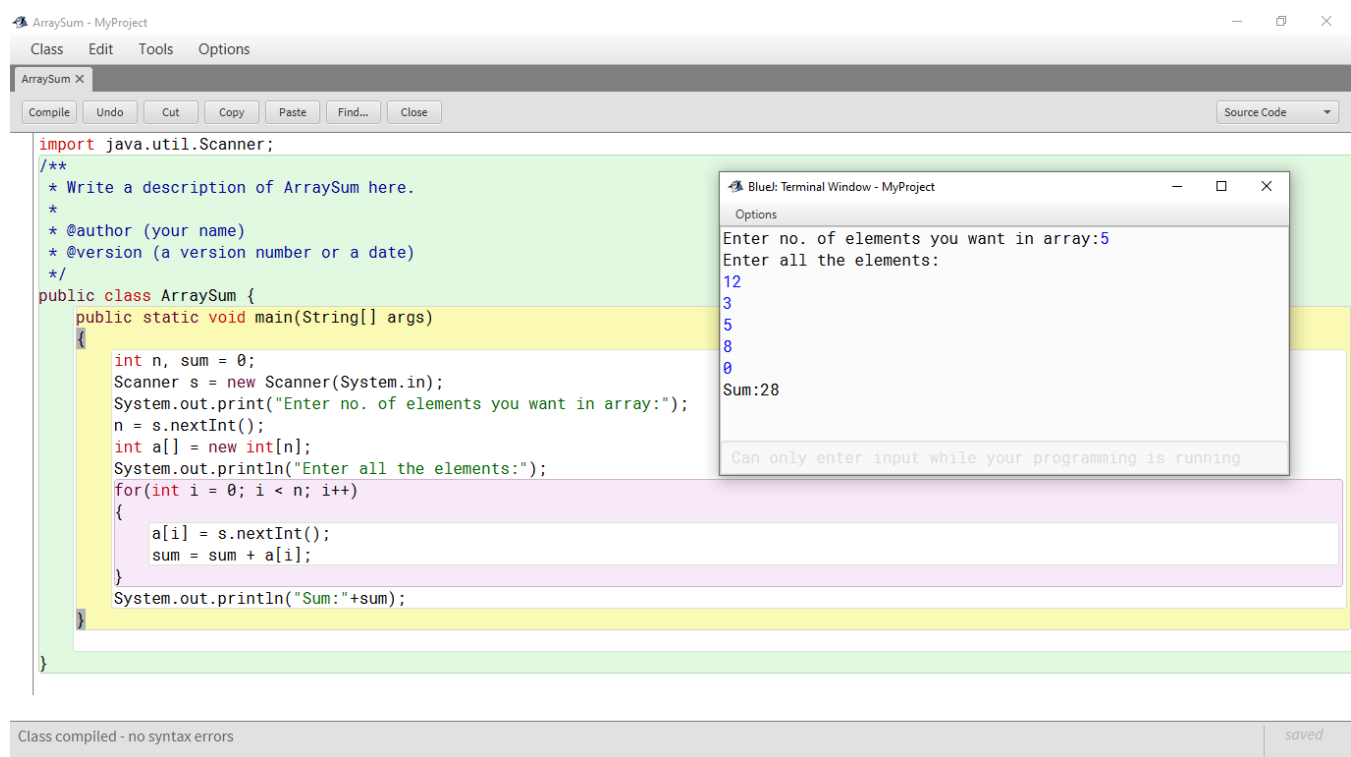
        for(int i = 0; i < n; i++){

            a[i] = s.nextInt();

            sum = sum + a[i];}

        System.out.println("Sum:"+sum); }

}
```



The screenshot displays an IDE window titled "ArraySum - MyProject". The main editor shows the Java code for "ArraySum.java", which is identical to the code provided in the previous block. The code is color-coded: keywords in blue, comments in green, and identifiers in black. A terminal window titled "BlueJ: Terminal Window - MyProject" is open, showing the program's execution. It prompts for the number of elements (5) and then for each element (12, 3, 5, 8, 0). After processing, it displays "Sum:28". A message at the bottom of the terminal states "Can only enter input while your programming is running". The IDE's status bar at the bottom indicates "Class compiled - no syntax errors" and "saved".

```
import java.util.Scanner;

/**
 * Write a description of ArraySum here.
 *
 * @author (your name)
 * @version (a version number or a date)
 */
public class ArraySum {
    public static void main(String[] args)
    {
        int n, sum = 0;
        Scanner s = new Scanner(System.in);
        System.out.print("Enter no. of elements you want in array:");
        n = s.nextInt();
        int a[] = new int[n];
        System.out.println("Enter all the elements:");
        for(int i = 0; i < n; i++)
        {
            a[i] = s.nextInt();
            sum = sum + a[i];
        }
        System.out.println("Sum:"+sum);
    }
}
```

BlueJ: Terminal Window - MyProject

Options

Enter no. of elements you want in array:5

Enter all the elements:

12

3

5

8

0

Sum:28

Can only enter input while your programming is running

Class compiled - no syntax errors | saved

Guneet Kohli 1805172

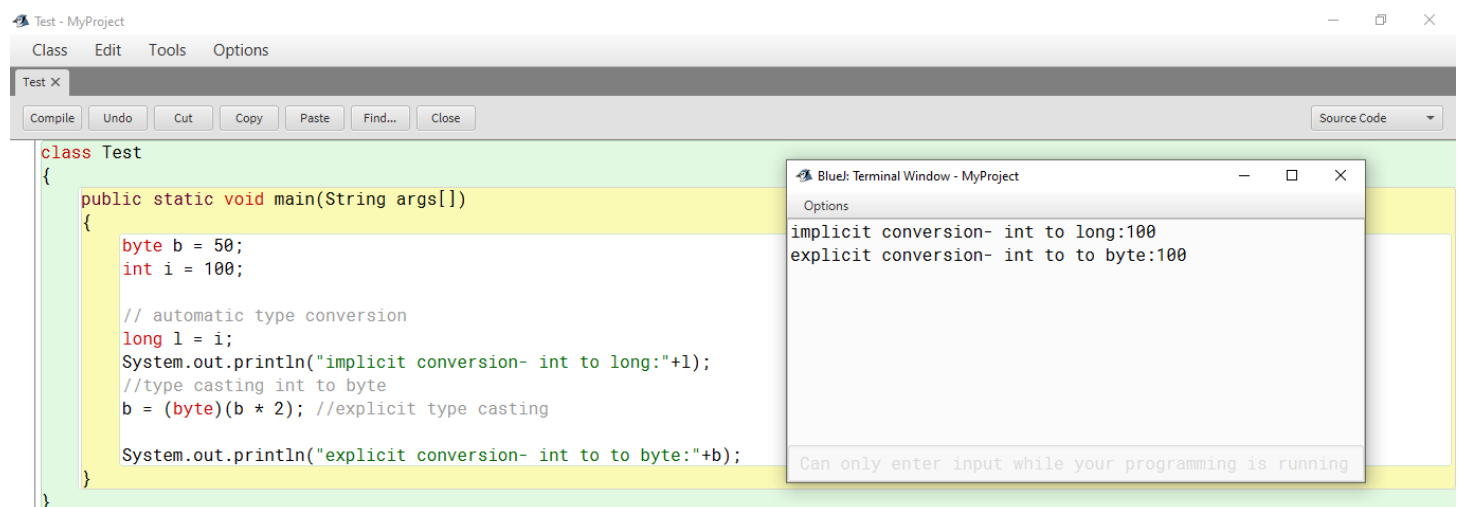
PROGRAM-9

Program to show implicit and explicit type casting.

```
class Test
{
    public static void main(String args[])
    {
        byte b = 50;
        int i = 100;

        // automatic type conversion
        long l = i;
        System.out.println("implicit conversion- int to long:"+l);
        //type casting int to byte
        b = (byte)(b * 2); //explicit type casting

        System.out.println("explicit conversion- int to to byte:"+b);
    }
}
```



Class compiled - no syntax errors

saved

PROGRAM-10

Program to demonstrate the use of static variables and static methods.

```
public class StaticTest {  
  
    static int i;  
  
    static void myMethod()  
  
    {  
  
        System.out.println("myMethod");  
  
    }  
  
  
    public static void main(String args[]){  
  
        for(i=1;i<=5;i++)  
  
        { System.out.println(i);  
  
        }myMethod();  
  
    }  
  
}
```

