******

***Sonargaon University(SU)***

**Department of Computer Science and Engineering**

**B.Sc Engineering in Computer Science and Engineering**

Assignment name : Lab Final Assignment

Course title : State of Art Programming Sessional

Course code : CSE334

Section : 20B

Session : Spring 2023

|  |  |
| --- | --- |
| **Submitted by** | **Submitted to** |
| Name : Riyaz Hossain  ID : CSE1901016170 | Name: Md. Ashfakur Rahman  Designation: Lecturer |

**Date: 1st April 2023**

**Task No. : 01**

**Task name: Interface**

**Source code:**

**myAssignment.java**

package com.mycompany.myassignment;

import java.util.Scanner;

import Interface\_Assignment.Rectangle;

public class MyAssignment {

public static void main(String[] args) {

Scanner scan = new Scanner(System.in); // Create Reader

System.out.print("Enter Your height of the ractangle : "); // Ask the user for something

int a = scan.nextInt(); // Read value from user

System.out.println("Enter the width of the ractangle : ");

int b = scan.nextInt();

Rectangle re = new Rectangle();

re.Area(a, b);

}

}

**Area.java**

package Interface\_Assignment;

public interface Area {

public void Area(float a, float b);

}

**Rectangle.java**

package Interface\_Assignment;

public class Rectangle implements Area {

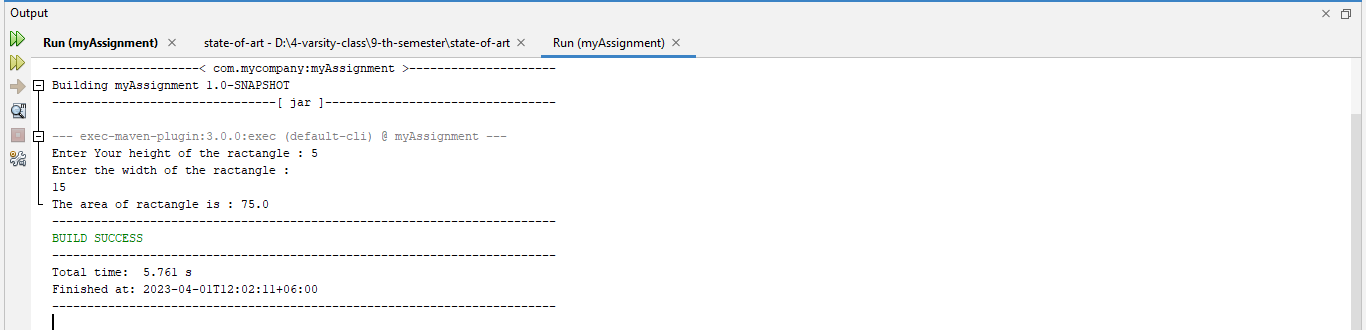
public void Area(int a, int b) {

System.out.println("The area of ractangle is : " + a \* b);

}

}

**Output:**



**Task No. : 02**

**Task name: Polymorphism(Overloading and overriding)**

**Source code:**

**myAssignment.java**

package com.mycompany.myassignment;

import java.util.Scanner;

import Interface\_Assignment.Rectangle;

import polymorphism\_assignment.\*;

public class MyAssignment {

public static void main(String[] args) {

Scanner scan = new Scanner(System.in);

System.out.print("Enter the value of A : ");

int a = scan.nextInt();

System.out.println("Enter the value of B : ");

int b = scan.nextInt();

System.out.println("Enter the value of C : ");

int c = scan.nextInt();

Overloading ovl = new Overloading();

System.out.println("Addition in overloading is : " + ovl.add(a, b));

System.out.println("Addition in overloading is : " + ovl.add(a, b,c));

Overriding ovr = new Overriding();

System.out.println("Addition from overriding : "+ovr.add(a, b, c));

}

}

**Overloading.java**

package polymorphism\_assignment;

public class Overloading {

public int add(int a, int b)

{

return a + b;

}

public int add(int a, int b, int c)

{

return a + b + c;

}

}

**Overriding.java**

package polymorphism\_assignment;

public class Overriding extends Overloading {

public int add(int a, int b, int c)

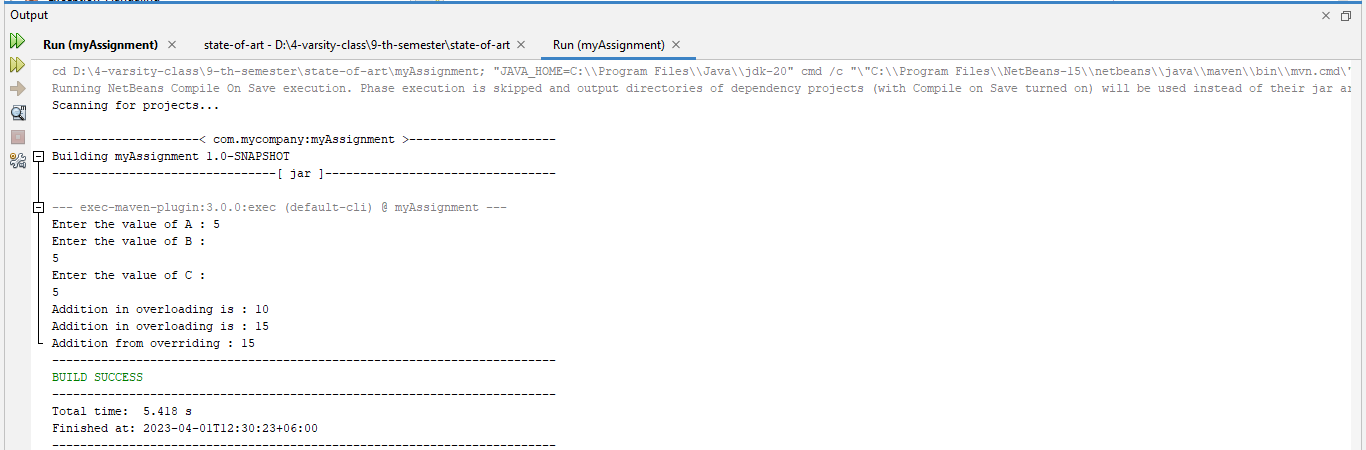
{

return a + b + c;

}

}

**Output:**



**Task No. : 03**

**Task name: Inheritance**

**Source code:**

**myAssignment.java**

package com.mycompany.myassignment;

import java.util.Scanner;

import Interface\_Assignment.Rectangle;

import polymorphism\_assignment.\*;

public class MyAssignment {

public static void main(String[] args) {

Scanner scan = new Scanner(System.in);

System.out.print("Enter the value of A : ");

int a = scan.nextInt();

System.out.println("Enter the value of B : ");

int b = scan.nextInt();

System.out.println("Enter the value of C : ");

int c = scan.nextInt();

Overloading ovl = new Overloading();

System.out.println("Addition Form Parent class : " + ovl.add(a, b));

System.out.println("Addition From Parent class : " + ovl.add(a, b,c));

Overriding ovr = new Overriding();

System.out.println("Addition Child Class : "+ovr.add(a, b, c));

}

}

**Overloading.java**

package polymorphism\_assignment;

public class Overloading {

public int add(int a, int b)

{

return a + b;

}

public int add(int a, int b, int c)

{

return a + b + c;

}

}

**Overriding.java**

package polymorphism\_assignment;

public class Overriding extends Overloading {

public int add(int a, int b, int c)

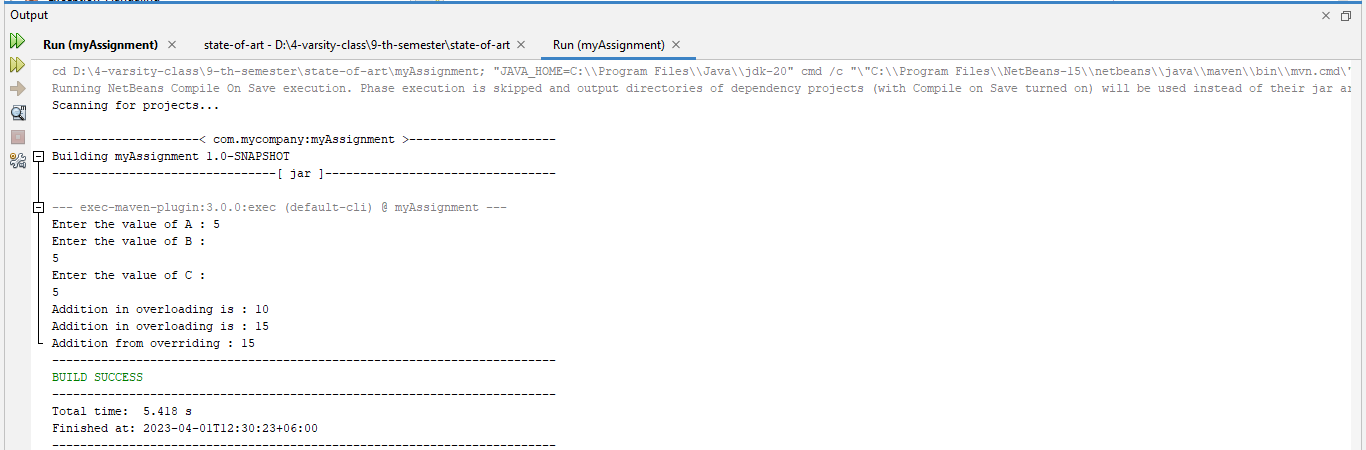
{

return a + b + c;

}

}

**Output:**



**Task No. : 04**

**Task name: Enum and Vector**

**Source code:**

**myAssignment.java**

package com.mycompany.myassignment;

import java.util.Scanner;

import Interface\_Assignment.Rectangle;

import polymorphism\_assignment.\*;

//enum

public class MyAssignment {

enum Level {

LOW,

MEDIUM,

HIGH

}

public static void main(String[] args) {

Level myVar = Level.MEDIUM;

switch(myVar) {

case LOW:

System.out.println("Low level");

break;

case MEDIUM:

System.out.println("Medium level");

break;

case HIGH:

System.out.println("High level");

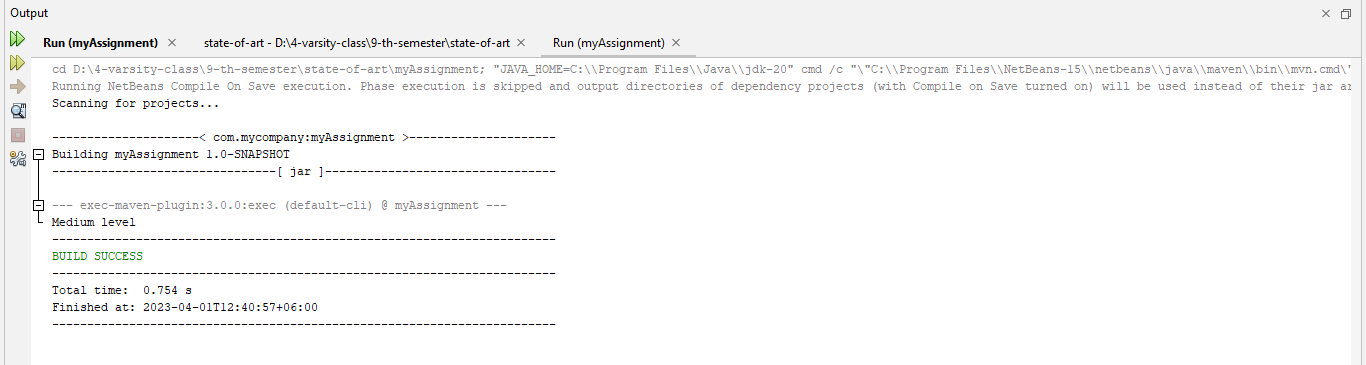
break;

}

}

}

**Output:**



//vector

package com.mycompany.myassignment;

import java.util.Scanner;

import java.io.\*;

import java.util.\*;

import Interface\_Assignment.Rectangle;

import polymorphism\_assignment.\*;

public class MyAssignment {

public static void main(String[] args) {

// Size of the Vector

int n = 5;

// Declaring the Vector with

// initial size n

Vector<Integer> v = new Vector<Integer>(n);

// Appending new elements at

// the end of the vector

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

v.add(i);

// Printing elements

System.out.println(v);

// Remove element at index 3

v.remove(3);

// Displaying the vector

// after deletion

System.out.println(v);

// iterating over vector elements

// using for loop

for (int i = 0; i < v.size(); i++)

// Printing elements one by one

System.out.print(v.get(i) + " ");

}

}

**Vector output:**



**Task No. : 05**

**Task name: Exception Handling**

**Source code:**

**myAssignment.java**

package com.mycompany.myassignment;

import java.util.Scanner;

import java.io.\*;

import java.util.\*;

import Interface\_Assignment.Rectangle;

import polymorphism\_assignment.\*;

public class MyAssignment {

public static void main(String[] args) {

try {

int[] myNumbers = {1, 2, 3};

System.out.println(myNumbers[10]);

} catch (Exception e) {

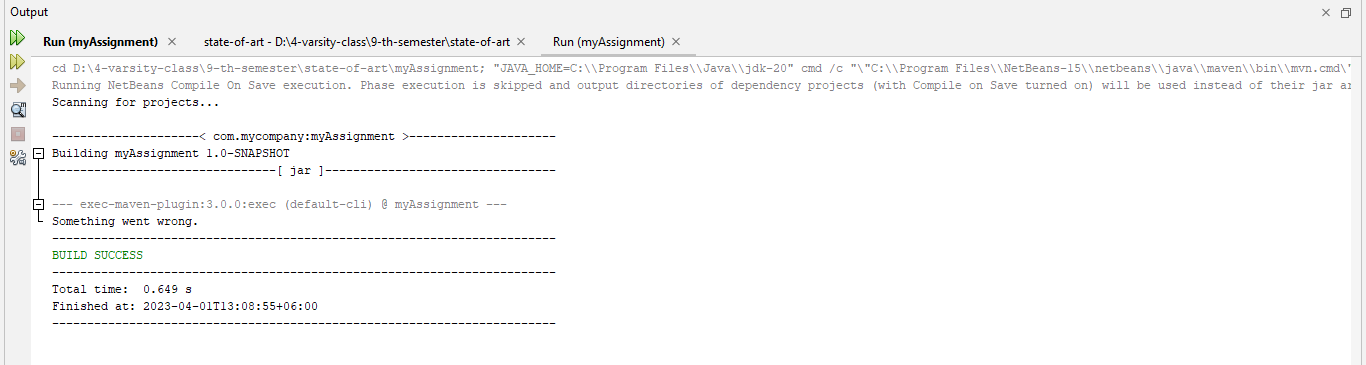
System.out.println("Something went wrong.");

}

}

}

**Output:**



**Task No. : 06**

**Task name: File Operation**

**Source code:**

**FileOperation.java**

package com.mycompany.fileoperation;

import java.io.File;

import java.io.IOException;

public class FileOperation {

public static void main(String[] args) {

try {

File Obj = new File("myfile.txt");

if (Obj.createNewFile()) {

System.out.println("File created: "

+ Obj.getName());

}

else {

System.out.println("File already exists.");

}

}

catch (IOException e) {

System.out.println("An error has occurred.");

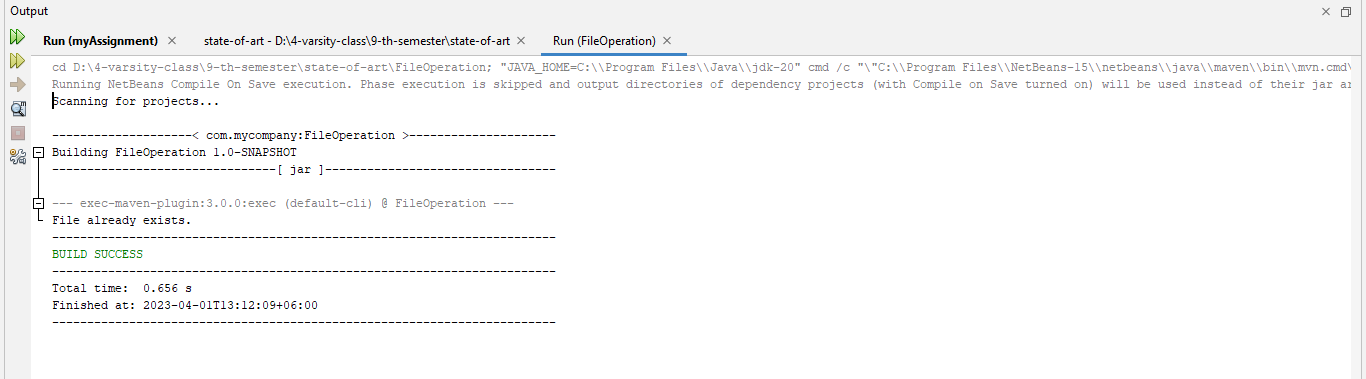
e.printStackTrace();

}

}

}

**Output:**



**Task No. : 07**

**Task name: Package**

**Source code:**

**myAssignment.java**

package com.mycompany.myassignment;

import java.util.Scanner;

import Interface\_Assignment.Rectangle;

import polymorphism\_assignment.\*;

public class MyAssignment {

public static void main(String[] args) {

Scanner scan = new Scanner(System.in);

System.out.print("Enter the value of A : ");

int a = scan.nextInt();

System.out.println("Enter the value of B : ");

int b = scan.nextInt();

System.out.println("Enter the value of C : ");

int c = scan.nextInt();

Overloading ovl = new Overloading();

System.out.println("Addition Form Parent class : " + ovl.add(a, b));

System.out.println("Addition From Parent class : " + ovl.add(a, b,c));

Overriding ovr = new Overriding();

System.out.println("Addition Child Class : "+ovr.add(a, b, c));

}

}

**Overloading.java**

package polymorphism\_assignment;

public class Overloading {

public int add(int a, int b)

{

return a + b;

}

public int add(int a, int b, int c)

{

return a + b + c;

}

}

**Overriding.java**

package polymorphism\_assignment;

public class Overriding extends Overloading {

public int add(int a, int b, int c)

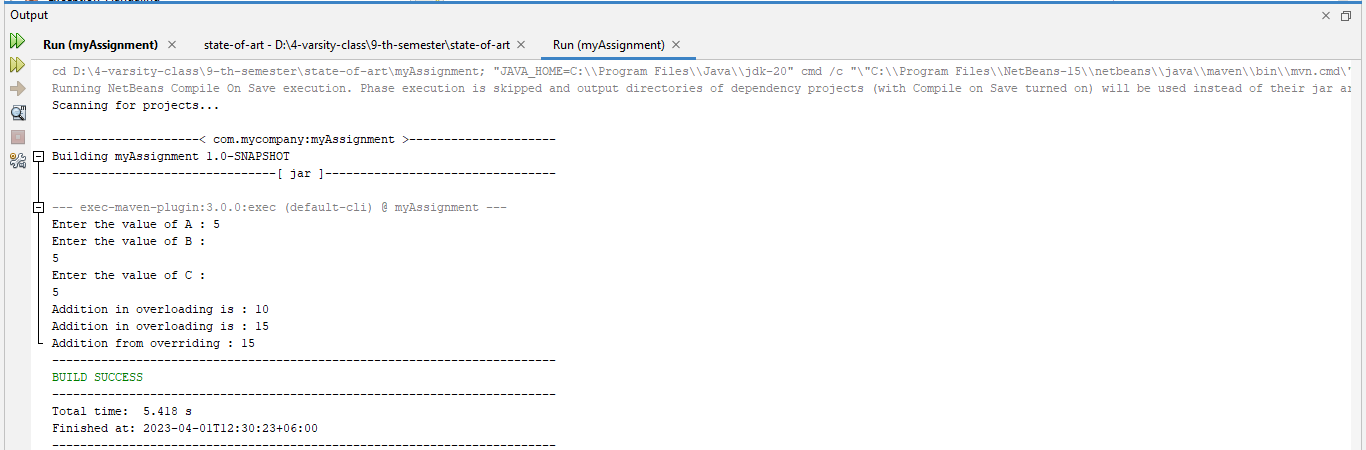
{

return a + b + c;

}

}

**Output:**



**Task No. : 08**

**Task name: Label Loop**

**Source code:**

**LabelLoop.java**

package com.mycompany.labelloop;

public class LabelLoop {

public static void main(String[] args) {

int i, j;

//outer loop

outer: //label

for (i = 1; i <= 5; i++) {

System.out.println();

//inner loop

inner: //label

for (j = 1; j <= 10; j++) {

System.out.print(j + " ");

if (j == 9) {

break inner;

}

}

}

}

}

