Week-7

1. Write a Java program to create an abstract class Animal with an abstract method called sound(). Create subclasses Lion and Tiger that extend the Animal class and implement the sound() method to make a specific sound for each animal.

Code:

abstract class Animal {

abstract void sound();

}

class Lion extends Animal {

void sound() {

System.out.println("Lion roars: Roarrr!");

}

}

class Tiger extends Animal {

void sound() {

System.out.println("Tiger growls: Grrrr!");

}

}

public class AnimalSound {

public static void main(String[] args) {

Lion l=new Lion();

l.sound();

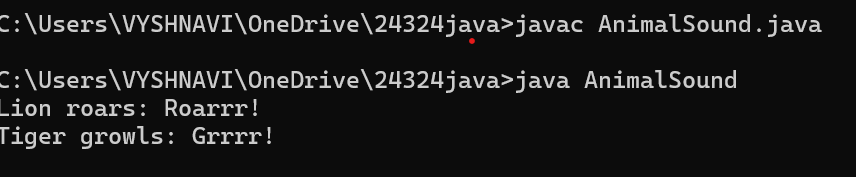
Tiger t=new Tiger();

t.sound();

}

}

Output:



Error Message:

|  |  |
| --- | --- |
| Error Message | Error Rectification |
| Java can’t find the main method to start your program Error rectification Convert static void main(String[ ] args) into | Convert static void main(String[ ] args) into Public static void main(String[ ] args |

Important points:

1)JVM(java virtual machine) is one which runs our program.JVM starts your program by calling the main( ) method from outside your class.JVM runs outside our package.only public method are accessible from outside a class or package.

2. )Write a Java program to create an abstract class Shape3D with abstract methods calculateVolume() and calculateSurfaceArea(). Create subclasses Sphere and Cube that extend the Shape3D class and implement the respective methods to calculate the volume and surface area of each shape.

CODE:

abstract class Shape3D{

abstract double CalculateVolume();

abstract double CalculateSurfaceArea();

}

class Sphere extends Shape3D{

double radius;

Sphere(double r) { this.radius=r;

}

double CalculateVolume() {

return (4.0/3.0)\*Math.PI\*radius\*radius\*radius;

}

double CalculateSurfaceArea() {

return 4\*Math.PI\*radius\*radius;

}

}

class Cube extends Shape3D{

double Side;

Cube(double s) {

this.Side=s;

}

double CalculateVolume() {

return Side\*Side\*Side;

}

double CalculateSurfaceArea() {

return 6\*Side\*Side;

}

}

public class ass18 {

public static void main(String[]args) {

Shape3D S=new Sphere(5.0); Shape3D C=new Cube(4.0);

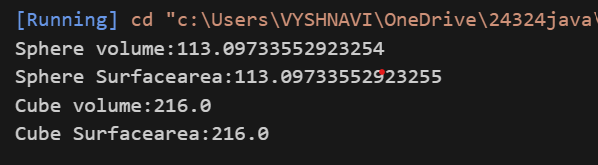
System.out.println("Name :S.Vyshnavi"+"Roll No:AV.SC.U4CSE24324"+"Section:CSE B");

System.out.println("Volume of Sphere:”+S.CalculateVolume()); System.out.println("Surface Area of Sphere:”+S.CalculateSurfaceArea()); System.out.println("volume of Cube:”+C.CalculateVolume()); System.out.println("Surfacearea of Cube:”+C.CalculateSurfaceArea());

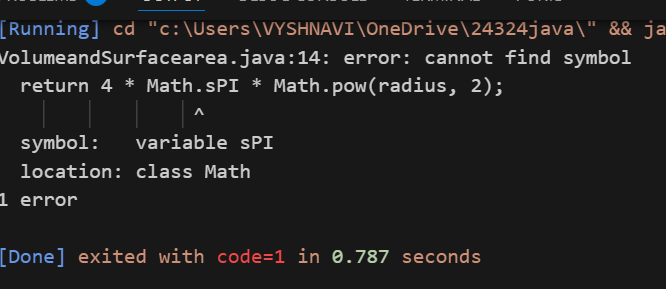
}

}

Output:



Error message:



|  |  |
| --- | --- |
| Error message | Error rectification |
| Cannot find the symbol | Rectified by removing the Math.sPI by removing ‘s’in Math.sPI. |

3)write a java program using an abstract class to define a method for pattern printing

• Create an abstract class named patternprinter with an abstract method printpattern(int n) and a concrete method to display the pattern title

• Implement two subclasses

1. Star pattern prints a right angle triangle of Star( \*)

2. Numberpattern-prints a right angled triangle of increasing numbers

• In the main( ) method ,create objects of both subclasses and print the patterns for a given number of rows

Excepted Output:

Pattern 1:

\*

\* \*

\* \* \*

\* \* \* \*

\* \* \* \* \*

Pattern 2:

1

1 2

1 2 3

1 2 3 4

1 2 3 4 5

CODE:

abstract class PatternPrinter {

public abstract void printPattern(int n);

public void displayTitle(String title) {

System.out.println("\n" + title);

}

}

class StarPattern extends PatternPrinter {

@Override

public void printPattern(int n) {

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

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

System.out.print("\* ");

}

System.out.println();

}

}

}

class NumberPattern extends PatternPrinter {

@Override

public void printPattern(int n) {

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

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

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

}

System.out.println();

}

}

}

public class Pattern {

public static void main(String[] args) {

int rows = 5;

PatternPrinter starPattern = new StarPattern();

starPattern.displayTitle("Pattern 1:");

starPattern.printPattern(row 1);

PatternPrinter numberPattern = new NumberPattern();

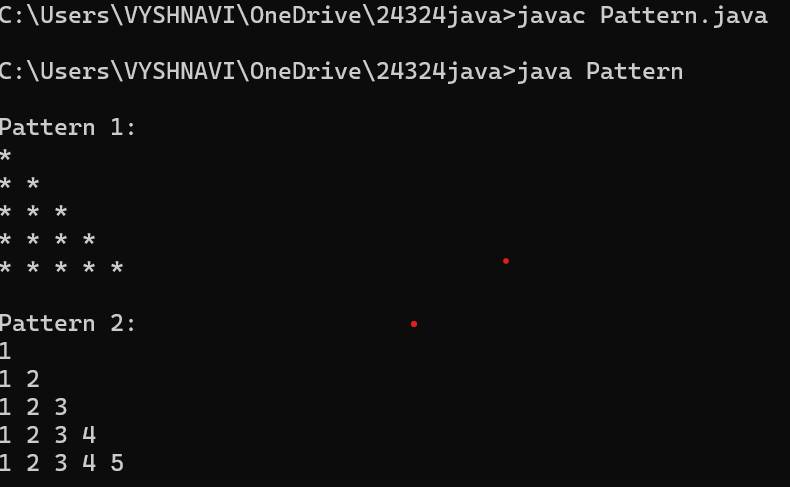
numberPattern.displayTitle("Pattern 2:");

numberPattern.printPattern(rows);

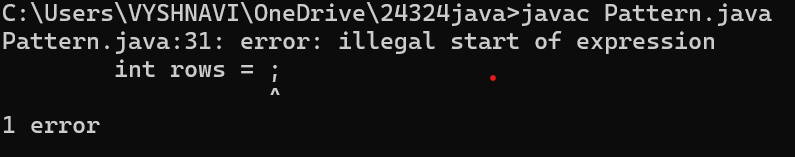
}

}

Output:



Error message:



Error Table:

|  |  |
| --- | --- |
| Error Message | Error Rectification |
| Iilegal start of expression | Rectified by mentioning the No.of rows i.e 5 rows. |