

Chapter 09:

Q What will be the area of circle if radius is 1, 25, 125. Write a program of it.

Algorithm:-

1) Main Class.

2) Sample Class

↓
getRadius

setRadius

3) Method of calculating area of circle

4) The calculating area of circle by calling it in a main class.

Code:-

// Using Getter Setter //

class Sample {

 int radius;

 public int getRadius() {

 return radius;

}

 public void setRadius (int radius) {

 this.radius = radius;

}

 public void calcArea {

 System.out.println ("Area of circle is " +
 Math.PI * radius * 2);

```
public class Circle Calc {  
    psvm {  
        Sample obj = new Sample();  
        obj.setRadius(1);  
        obj.getRadius();  
  
        obj.setRadius(25);  
        obj.getRadius();  
    }  
}
```

Using Constructor

```
class Sample {  
    float radius;
```

```
Sample(float radius){  
    this.radius = radius;  
}
```

```
public void CalcArea(){  
    sout(2 * Math.PI * radius);  
}
```

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

```
        Sample obj1 = new Sample(1);  
        Sample obj2 = new Sample(25);
```

```
        obj1.CalcArea();
```

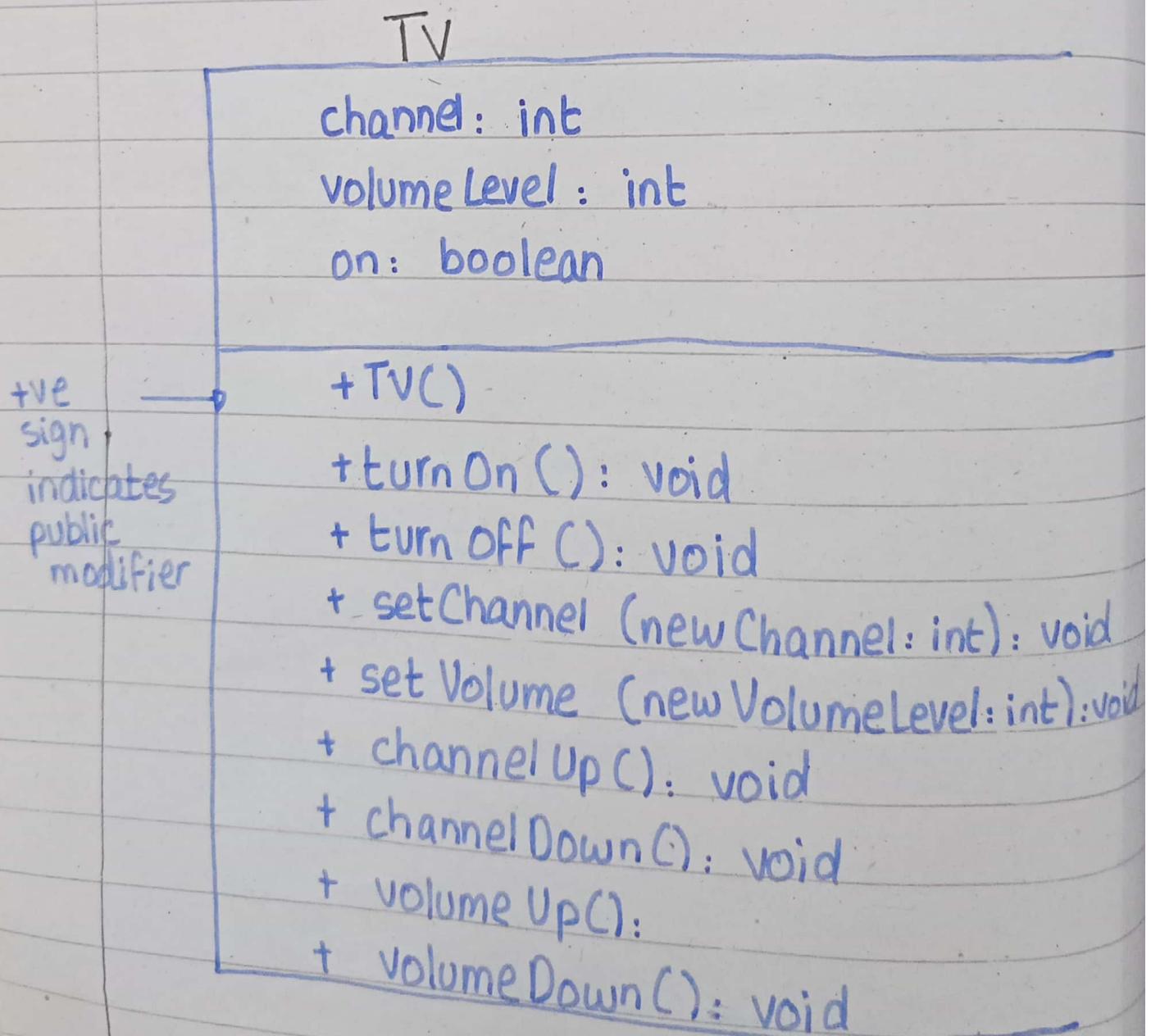
```
        obj2.CalcArea();
```

```
}
```

```
}
```

Q2 Consider television sets. Each TV is an object with states (current channel, adjust volume level, turn on or off) and behaviors (change channels, adjust volume, turn on/off). You can use a class model TV sets.

The UML diagram for class is :-



The Current channel (1 to 120) of this TV.
The Current volume (1 to 7) of this TV
Indicates whether TV is on or off.

Construct default TV object

Turn on this TV.

Turn off

Set a new channel for TV.

Set a new volume level

Channel number by 1.

_____ X _____ X _____

Code :-

```
public class TV {  
    int channel = 1;           // Default value //  
    int volumeLevel = 1;       // Default value //  
    boolean on = false;        // TV is off
```

TV() { // Constructor //

}

void turnOn() {

on = true;

}

void turnOff() {

on ~~off~~ = false;

}

```
public void setChannel(int newChannel){  
    if (on 44 newChannel >= 1 44  
        new Channel <= 120)  
        channel = newChannel;  
}
```

```
public void setVolumeLevel(int newVolumeLevel)  
    if (on 44 newVolumeLevel >= 1 44  
        new Volume level <= 7)  
        volumeLevel = newVolumeLevel;  
}
```

```
public void channelUp(){  
    if (on 44 channel < 120);  
        channel ++;  
}
```

```
public void channelDown(){  
    if (on 44 channel > 1)  
        channel --;  
}
```

```
Public void volumeUp(){  
    if (on 44 volumeLevel < 7)  
        volumeLevel ++;
```

```
public void volumeDown() {
    if (on && volumeLevel > 1)
        volumeLevel--;
}
```

```
}
```

— TestTV class —

```
public class TestTV {
    public static void main (String [] args) {
        TV obj = new TV ();
        obj.turnOn ();
        obj.setChannel (44);
        obj.setVolumeLevel (3);
    }
}
```

```
TV obj2 = new TV ();
obj2.turnOn ();
obj2.channelUp ();
obj2.volumeDown ();
obj2.channelUp ();
```

```
cout ("TV1's channel is " + obj.channel
      + " volume is " + obj2.volume);
```

```
}
```

```
}
```

- Question 9.1:
Describe relationship b/w an object and its defining class.
- An object is a representation of class which acts as a template for defining objects and their behaviour.
- Object Properties are represented by datafields while behaviour is represented by methods.
- Question 9.2:
How do you define a class

public class ClassName {
.....
}

The class usually contains constructor, various data field and method in its body.

Question 9.3 :-

How do you declare an object's reference variable?

If the class is defined with name Class Name, then object is referenced with following syntax:

Class Name mathematics;

Question 9.4:-

How do you create an object

ClassName mathematics = new ClassName;

If object reference variable is already declared with name mathematics the creation syntax is following.

Q Question 9.5:

What are differences b/w constructor and methods?

Method

- A method is used to expose behaviour of an object.
- ~~Method~~ Constructor must have return type
- It invoked ~~ex~~ plicity.
- Not have same name as class name

Constructor

- Constructor is used to initialize state of an object.
- Constructor must not have return type.
- Constructor is invoked implicitly having or must be same as class name

Q Question 9.6:

When will a class have a default constructor?

A class has a default constructor only if programmer does not define any constructor explicitly.

A default constructor is always empty and does not accept any argument.

Q Question 9.7

Which operator is used to access a datafield or invoke a method from an object?

The dot operator(.) is used to access a datafield or invoke a method from an object. Its alternative name is object member access operator.

Q 9.7

What is an anonymous object?

An anonymous object is an object without a variable that is referencing it. In other words, it is created with new operator, but is not stored in any variable.

System.out.println(new Calculation().getSquare())

Q Question 9.9:

What is null pointer exception?
Null pointer exception is a runtime exception.
It is thrown when program attempts
to use an object reference that has
null value.

Q Question 9.11:

What is wrong with each of following
programs?

```
1 public class ShowErrors{  
2     public static void main (String [] args){  
3         ShowErrors t = new ShowErrors(5);  
4     }  
5 }
```

Error is in Line 3, when we are
initializing a new object, in the
code it passes an argument and
there is no need for an argument.

```
1 public class ShowErrors {  
2     public static void main (String [] args) {  
3         ShowErrors t = new ShowErrors ();  
4         t.x ();  
5     }  
6 }
```

Error is in line 4. The program calls method x and there is no method x is defined in that class.

```
1 public class ShowErrors {  
2     public void method {  
3         Circle c ;  
4         System.out.println ("What is radius"  
5                             + c.getRadius () );  
6         c = new Circle ();  
7     }  
8 }
```

The error is in Line 4 and Line 6. In the given code, program calls for a method of an object c before object has been initialized.

```
1 public class ShowErrors {
2     public static void main (String [] args) {
3         C c = new C (5.0);
4         System.out.println (c.value);
5     }
6 }
7 class C {
8     int value = 2;
9 }
```

: Error is in 3 line because we
couldn't create constructor
of class C and we are putting
a double 5 in an argument,

Question 9.12:-

```
1 class Test {  
2     public static void main (String [] args) {  
3         A a = new A();  
4         a.print();  
5     }  
6 }  
7 class A {  
8     String s;  
9  
10    A (String news) {  
11        s = news;  
12    }  
13    public void print () {  
14        System.out.print (s);  
15    }  
16 }
```

Answer:-

Error is in line 3; It is calling a default constructor (parameterless) and there is no default constructor in Class A.

Q

9.13:

What is the output of following ~~data~~ code.

```
public class A {  
    boolean x;  
  
    public static void main (String [] args){  
        System.out.println  
            A a = new A();  
        System.out.println(a.x);  
    }  
}
```

Answer:

Since java have a default value
of boolean which is False

Output:

False

Ans

Q How do you create a Date for current time?
How do you display current time?

```
java.util.Date date = new java.util.Date()  
System.out.println(date.toString());
```

Question 9.15:-

How do you create a Point2D ? suppose
p1 and p2 are two instances of Point2D?
How do you obtain distance b/w two points?

```
import java.util.Scanner;  
import javafx.geometry.Point2D;
```

```
public class Test TestPoint2D {  
    public static void main(String[] args) {  
        Scanner input = new Scanner(System.in);
```

```
        System.out.print("Enter point1's x, y: ");  
        double x1 = input.nextDouble();  
        double y1 = input.nextDouble();
```

```
        System.out.print("Enter point2's x, y: ");  
        double x2 = input.nextDouble();  
        double y2 = input.nextDouble();
```

```
Point 2D p1 = new Point2D(x1, y1);  
Point 2D p2 = new Point2D(x2, y2);
```

```
System.out.println("P1 is " + p1.toString());  
System.out.println("P2 is " + p2.toString());
```

```
System.out.println("Distance b/w  
P1 and P2 is: " + p1.distance(p2));
```

```
}
```

Output:-

Enter point1's x,y: 1.5 5.5

Enter point2's x,y: -5.3 -4.4

p1 is Point2D [x = 1.5, y = 5.5]

p2 is Point2D [x = -5.3, y = -4.4]

distance b/w P1 and P2 is 12.01042149

Question 9.16:

Which package contains classes, Date, Random, Point2D, System and Math

java.util.Date;
java.util.Random;
java.lang.System;
java.lang.Math;
java.Fx.geometry.Point2D;

Question 9.17:

Suppose that class F is defined in (a)
let F be an instance of F. Which of the
following statements in (b) are correct?

public class F {
 int i;
 static String s;
 void imethod(){
 }
 static void smethod(){
 }
}

v_a

sout(F.i);
sout(F.s);
F.imethod();
F.smethod();
sout(F.i);
sout(F.s);
F.imethod();
F.smethod();
v_b

Answer: (Incorrect Statement):

sout(F.i); // i is not static
F.imethod(); // i method is not static
 // static

Question 9.18:-

Add static keyword in the place of ?
if appropriate.

```
1 public class Test {  
2     int count;  
3     public ? void main(String[] args){  
4         ....  
5     }  
6     public ? int getCount(){  
7         return count;  
8     }  
9  
10    public ? int factorial(int n){  
11        int result = 1;  
12        for (int i = 1; i <= n; i++)  
13            result *= i;  
14        return result;  
15    }  
16}
```

At Line 3 and Line 10 because static is used for variables and methods that are not dependent on a specific instance of a class so they should be static.

Question 20:-

• What is an accessor method? What is a mutator method? What are naming convention of accessor and mutator method

An accessor method is used to retrieve private instance data outside the class.

Also known as getter method.

A mutator method is used to enable user to change the value of private instance data outside of class. Known as setter

Convention will be same for both mutator and accessor which is camel convention.

Question 9.21:

What are benefits of datafield encapsulation?

- It is a way to achieve data hiding.
- It provides you to control over data
- You can make class readonly or write only.

Question 9.22:

In the following code, radius is private in Circle class and myCircle is an object of Circle class. Does the highlighted code cause any problem? if so why.

```
public class Circle{  
    private double radius = 1;
```

```
    public double getArea(){  
        ....  
    }
```

```
public static void main(String [] args){  
    Circle myCircle = new Circle();  
    System.out.println ("Radius is "+  
        myCircle.radius);  
}  
}
```

Answer:

It doesn't cause any problem because it is inside Circle class and myCircle is an instance of that class. If it were outside the class, it will cause an issue.

• Question 9.23:

Describe difference b/w a parameter of a primitive type and passing a

parameter of a reference type.

Show output of the following code.

```
public class Test {
```

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

```
        int times = 0;
```

```
        for (int i = 0, i < 100, i++)
```

```
            increment (myCount, times);
```

// Count = 99
time

```
        System.out.println ("count is " + myCount);
```

```
        System.out.println ("time is " + times);
```

```
    public static void increment (Count c, int time)
```

```
    { c.count++;
```

```
        times++;
```

```
}
```

```
}
```

```
public class Count {  
    public int count;  
    public Count (int c) {  
        count = c;  
    }  
}
```

```
public Count () {  
    count = 1;  
}  
}
```

Output:

Count is 101

Time is 0

Reason:

For loop in the main method iterates 100 times and in each iteration, increment method increase the count ~~meth~~ variable by 1. Therefore after 100 iteration becomes $1 + 100 = 101$

Time variable in main method is not being updated by increment method due to java pass by value nature. The time variable in the increment is a local copy and changes made to it does not affect main method

- Passing a Parameter of a Primitive type:
 - Pass by Value:

When you pass a parameter of a primitive type (e.g., int, char, double), you are passing a copy of actual value to method.
 - No impact on Original Value:

Any modification made to the parameter inside the method are confined to ~~to~~ the local copy and they do not affect original value of variable outside method.
- Passing a parameter of a reference type
 - When you pass a parameter of a reference type (e.g. objects, arrays), you are passing value of reference to the object not the actual object itself.

Question 9.24: Show output of following program.

```
public class Test {  
    public static void main (String [] args){  
        Circle circle1 = new Circle(1); // r=1  
        Circle circle2 = new Circle(2); // r=2  
  
        // 1           // 2  
        swap1 (circle1 , circle 2);  
        System.out.println ("After swap1: circle  
                           = " + circle1.radius + "circle2=" +  
                           circle2.radius);  
  
        swap2 (circle1., circle 2);  
        Sout ("After swap2: circle1=" +  
              circle1.radius}+ "circle2" +  
              circle2.radius);  
    }  
  
    public static void swap1 (Circle x, Circle y){  
        Circle temp = x; // temp = 1  
        x = y; // 1 = y;  
        y = temp;  
    }  
}
```

```
public static void swap2 (Circle x, Circle y){  
    double temp = x.radius;  
    x.radius = y.radius;  
    y.radius = temp;  
}
```

```
class Circle {  
    double radius;
```

```
    Circle (double newRadius) { //Constructor  
        radius = newRadius;  
    }  
}
```

// Output:

After swap1 circle1= 1.0 circle2 = 2.0

After swap2 circle1 = 2.0 circle2 = 1.0

Reason 1:

Swap1 Method:

- The method is attempting to swap references x and y but these changes are local to the method and do not affect references in main method.

. As a result, output after calling swap1 will not show any change in radii of circle1 and circle2.

Swap 2 Method:

This method correctly swaps the 'radius' value of two circle object

Question 9.25

Show the output of the following code:

```
public class Test {  
    public static void main (String [] args){  
        int [] a = {1, 2};  
        swap (a [0] , a [1]);  
        Sout ("a[0] = " + a [0] + "a[1] = " + a [1]);  
    }  
  
    public static void swap (int n1, int n2){  
        int temp = n1;  
        n1 = n2;  
        n2 = temp;  
    }  
}
```

Output:

a[0] = 1 a[1] = 2

b | public class Test {
| psvm{
| int[] a = {1,2};
| swap(a);
| sout("a[0] = "+a[0] + "a[1] = "+a[1]);
| }
| }

public static void swap(int[] a){
 int temp = a[0];
 a[0] = a[1];
 a[1] = temp;
}

Output:

a[0]= 2 a[1] = 1

```
public class Test {  
    public void swap(T t) {  
        T t1 = new T();  
        swap(t1);  
        cout ('e1=' + t.e1 + 'e2=' + t.e2);  
    }  
}
```

```
public static void swap (T t) {  
    int temp = t.e1;  
    t.e1 = t.e2;  
    t.e2 = temp;  
}  
}
```

```
class T {  
    int e1 = 1;  
    int e2 = 2;  
}
```

Output:
e1 = 2 e2 = 1

```
d public class Test {  
    public static void main (String [] args) {  
        T t1 = new T();  
        T t2 = new T();  
        sout ('t1's i = ' + t1.i + " and j = "  
              + t1.j);  
        sout ('t2's i = "' + t2.i + " and j = "  
              + t2.j);  
    }  
}
```

```
class T {  
    static int i = 0;  
    int j = 0;
```

```
T() {  
    i++;  
    j = 1;  
}  
}
```

Output:

t1's i=1 and j = 1
t2's i=12 and j = 1

Question 9.26:-

What is output of following object..

```
import java.util.Date;
```

```
public class Test {  
    psvm {  
        Date date = null;  
        m1(date);  
        sout(date)  
    }  
}
```

```
public static void m1(Date date){  
    date = new Date();  
}  
}
```

Output:-

null.

```
b | import java.util.Date;  
  
public class Test {  
    psvm {  
        Date date = new Date ("1234567");  
        m1(date);  
        sout (date.getTime());  
    }  
  
    public static void m1 (Date date) {  
        date = new Date ("7654321");  
    }  
}
```

Output:-

1234567

Question 9.27

What is wrong in the following code.

```
1 public class Main {  
2     public static void main(String[] args) {  
3         java.util.Date[] dates = new java.util.Date[10];  
4         System.out.println(dates[0]);  
5         System.out.println(dates[0].toString());  
6     }  
7 }
```

Answer

When you created an array its element is initialized to ~~null~~ null for objects. In your code, date [0] is null since you haven't assigned any date object.

mutable → changing
immutable → not changing

• Question 9.28:

If a class contains only private data fields and no setter method, is the class immutable?

For the private data fields to be immutable, no setter and getter method can exist.

• Question 9.29:-

If all data fields in a class are private and of primitive types and class doesn't contain any setter method, is the class immutable?

The class is immutable.

Question 9.30:

Q11 Is the following class immutable?

public class A {

 private int[] values;

 public int[] getValues() {

 return values;

}

This class is mutable because it have a getter method.

Question 9.31: What is output of following program?

public class Test {

 private static int i = 0;

 private static int j = 0;

 psvm {

 int i = 2;

 int k = 3;

{

 int i = 2;

 int k = 3;

{ int j = 3;

 cout (<i+j is " + i+j); // 23

}

```
K = i + j;  
cout << "K is " << K;  
cout << "j is " << j;
```

// 0+2=2

// 2

// 0

}

}

Output:

```
i + j is 23  
K is 2  
j is 0
```

Question 9.32:

Describe role of this keyword:

this keyword is used to refer current object in a method or constructor.

Question 9.33

What is wrong in following code?

```
1 public class C {  
2     private int p;  
3 }
```

```
4     public C() {  
5         cout << "C's no-arg constructor";  
6         this(0);  
7     }  
8 }
```

```
9     public C(int p) {  
10        p = p;  
11    }
```

```
public void setP(int p) {  
    p = p;  
}  
}
```

Answer:

Line 13, Line 9; we have to use this. $p = p$;
otherwise compile will give us error
because of same datafield and arg.
name

line 6:-

Another constructor can be invoked
with "this", but it has to be appear
first before any other executable statement

Q) Last Question:
What is wrong with the following code?

```
public class Test {  
    private int id;  
  
    public void m1() {  
        this.id = 45;  
    }
```

```
    public void m2() {  
        Test.id = 45;  
    }
```

Solution:

Error in above code is that we cannot write `Test.id = 45` because we are calling this datafield from other class or main class so we have to write `id = 45`.