Question 1

Which of the following is the correct syntax to output "Hello, World!" in Java?

1) `System.out.println("Hello, World!");`

2) `print("Hello, World!");`

3) `echo("Hello, World!");`

4) `Console.WriteLine("Hello, World!");`

Answer: 1) System.out.println("Hello, World!");

Question 2

Which of the following is a valid keyword in Java?

1) `interface`

2) `string`

3) `Float`

4) `unsigned`

Answer: 1) `interface`

Question 3

Which of the following is the correct way to declare an array in Java?

1) `int arr[] = new int[5];`

2) `int[] arr = new int[5];`

3) `int arr[];`

4) All of the above

Answer: 4) All of the above

Question 4

Which of the following loops is guaranteed to execute at least once?

1) `for` loop

2) `while` loop

3) `do-while` loop

4) None of the above

Answer: 3) `do-while` loop

Question 5

Which method is used to get the length of a string in Java?

1) `length()`

2) `getSize()`

3) `getLength()`

4) `size()`

Answer: 1) `length()`

Question 6

Which of these is the correct format for a main method in Java?

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

2) `public static main(String[] args)`

3) `public void main(String args[])`

4) `static public void main(String args)`

Answer: 1) `public static void main(String[] args)`

Question 7

What is a jagged Array ?

1) Rows will be constant for each row

2) Columns will differ for each row

3) Columns will be constant for each row

4) Rows and columns will be same for each row

Answer: 2) Columns will differ for each row

Question 8

Which of the following is not a feature of Java?

1) Platform independent

2) Object-oriented

3) Use of pointers

4) Multi-threaded

Answer: 3) Use of pointers

Question 9

Predict the output:

```
public class Variable
{
     int x =10;
     static int y = 23;
    public static void main(String[] args)
    {
        Variable d1 = new Variable();
        Variable d2  = new Variable();
        d1.x = d1.x+3;
        d2.y = d2.y+3;
       System.out.println(d1.x);
       System.out.println(d1.y);
       System.out.println(d2.x);
       System.out.println(d2.y);
     }
}
```

1.) 13 10 10 23

2.)23 10 10 13

3.) 13 23 13 23

4.) 13 23 10 23

Answer: 4.) 13 23 10 23

Question 10

What is the output of the following code?

```
int a = 5;
int b = 10;
System.out.println(a + b);
```

1) 5

2) 10

3) 15

4) 510

Answer: 3) 15

Question 11

What will be the output of the following code?

```
int x = 5;
int y = ++x * 10 / x++ + --x;
System.out.println(y);
```

1) 10

2) 11

3) 12

4) 13

Answer: 2) 11

Question 12

What will be the output of the following code?

int a = 5, b = 10, c = 15;

System.out.println(a + " " + b + " " + c);

1) 5 10 15

2) 51015

3) 5, 10, 15

4) None of the above

Answer: 1) 5 10 15

Question 13

What will be the output of the following code?

int[] arr = {1, 2, 3, 4, 5};

System.out.println(arr[2]+arr[4]++);

```

1) 10

2) 21

3) 9

4) 8

**Answer:** 4) 8
```

Question 14

What will be the output of the following code?

```
for(int i = 0; i < 5; i++) {
    if(i == 3) {
        continue;
    }
    System.out.print(i + " ");
}
```

1) `0 1 2 3 4`

2) `0 1 2 4`

3) `0 1 2 4 5`

4) `0 1 2 3 5`

Answer: 2) `0 1 2 4`

Question 15

Predict the output of this code:

```
abstract class Car{
   abstract void changeGear();
   abstract void pressClutch();
    void startEngine() {
          System.out.println("car has started");
    }
    void accelerate()
```

```
    {
                System.out.println("car is accerlerating");
    }

}

 abstract class NewCar extends Car
 {
            void changeGear()
            {
                  System.out.println("changing gear");
             }
            public static void main(String[] args) {
                  NewCar obj = new NewCar();
                  obj.startEngine();
                  obj.accelerate();
                  obj.changeGear();
          }
 }
```

1) car has started

    car is accerlerating

    changing gear

    clutch is pressed

2) car has started

   car is accerlerating

3) car has started
   car is accerlerating
   changing gear
4) compile-time error

Answer: 4) compile-time error

Question 16

```
interface A
{
          int a = 5;
         void show();
}
class Main implements A
{
        int b = 10;
    public void show()
    {
                System.out.println("show() method");
     }
   void print()
   {
        System.out.println("Print() method");
   }
   public static void main(String[] args)
```

```
    {
        A obj = new Main();
        obj.show();
        obj.print();
        System.out.println(A.a);
        System.out.println(obj.b);
    }}
```

1.) Show() method 5

2.) Print() Method 10

3.) Show() method Print() Method 5 10

4.) Compile-time error

Answer: 4.) Compile-time error

Question 17

What will be the output of the following code?

```
public class Test {
    public static void main(String[] args) {
        String str = "123";
        str += "456";
        System.out.println(str);
    }
}
```

1) `123`

2) `123456`

3) `456`

4) `Error`

Answer: 2) `123456`

Question 18

Which of the following is a valid constructor for the `Test` class?

1) `public Test() {}`

2) `public void Test() {}`

3) `public static Test() {}`

4) `void Test() {}`

Answer: 1) `public Test() {}`

Question 19

What does the following Java code snippet do ?

```
public static void main(String[] args) {
    int[] arr = {3, 1, 4, 1, 5, 9, 2, 6};
    bubbleSort(arr);
    System.out.println(Arrays.toString(arr));
}

public static void bubbleSort(int[] arr) {
    int n = arr.length;
```

```
    for (int i = 0; i < n - 1; i++) {
        for (int j = 0; j < n - i - 1; j++) {
            if (arr[j] > arr[j + 1]) {
                int temp = arr[j];
                arr[j] = arr[j + 1];
                arr[j + 1] = temp;
            }
        }
    }
}
```

What will be printed by this code?

1) [1, 1, 2, 3, 4, 5, 6, 9]

2) [9, 6, 5, 4, 3, 2, 1, 1]

3) [3, 1, 4, 1, 5, 9, 2, 6]

4) [1, 1, 2, 3, 4, 5, 6, 9]

Answer: 4) [1, 1, 2, 3, 4, 5, 6, 9]

Question 20

Which of the following is not a valid statement in Java?

1) `int a = 10;`

2) `float b = 5.5;`

3) `double c = 5;`

4) `boolean d = 0;`

Answer: 4) `boolean d = 0;`

Question 21

```
class Demo {
        int add(int a, int b)
          {
            float c = a+b;
           System.out.println(c);
          }
        public static void main(String[] args)
       {
            new Demo().add(12,28);
       }
}
```

1.) 40.0

2.) 40

3.) Blank Output

4.) Compile-time error

Answer: 4.) Compile-time error

Question 22

Question: What does the following Java code snippet do?

```
public static void main(String[] args) {
    int[] arr = {2, 5, 7, 8, 1};
```

```
    insertionSort(arr);
    System.out.println(Arrays.toString(arr));
}

public static void insertionSort(int[] arr) {
    int n = arr.length;
    for (int i = 1; i < n; i++) {
        int key = arr[i];
        int j = i - 1;
        while (j >= 0 && arr[j] > key) {
            arr[j + 1] = arr[j];
            j--;
        }
        arr[j + 1] = key;
    }
}
```

What will be printed by this code?

1) [1, 2, 5, 7, 8]

2) [8, 7, 5, 2, 1]

3) [2, 5, 7, 8, 1]

4) [2, 1, 5, 7, 8]

Answer: 1) [1, 2, 5, 7, 8]

Question 23

What does the following Java code snippet do?

```
public static void main(String[] args) {
    int[] arr = {5, 2, 8, 1, 6};
    selectionSort(arr);
    System.out.println(Arrays.toString(arr));
}

public static void selectionSort(int[] arr) {
    int n = arr.length;
    for (int i = 0; i < n - 1; i++) {
        int minIndex = i;
        for (int j = i + 1; j < n; j++) {
            if (arr[j] < arr[minIndex]) {
                minIndex = j;
            }
        }
        int temp = arr[minIndex];
        arr[minIndex] = arr[i];
        arr[i] = temp;
    }
}
```

What will be printed by this code?

1) [1, 2, 5, 6, 8]

2) [8, 6, 5, 2, 1]

3) [5, 2, 8, 1, 6]

4) [1, 2, 5, 6, 8]

Answer: 1) [1, 2, 5, 6, 8]

Question 24

Which of the following is not a principle of OOP?

- a) Encapsulation
- b) Abstraction
- c) Inheritance
- d) Compilation

Answer: d) Compilation

Question 25

Which concept of OOP is used to achieve runtime polymorphism?

- a) Encapsulation
- b) Abstraction
- c) Inheritance
- d) Method Overriding

Answer: d) Method Overriding

Question 26

Which of the following statements is true about inheritance in Java?

- a) A subclass can inherit from multiple superclasses.
- b) A subclass can inherit from only one superclass.
- c) A class can be a superclass without having any subclasses.
- d) Inheritance is not supported in Java.

Answer: b) A subclass can inherit from only one superclass.

Question 27

What is the main purpose of a constructor in a class?

- a) To allocate memory for the class.
- b) To initialize objects of the class.
- c) To call other methods of the class.
- d) To inherit properties from another class.

Answer: b) To initialize objects of the class

Question 28

```
class A
{
  public static int hello()
  {
    int a = 10;
    System.out.println(a);
    return a;
  }
  public static void main(String[] args)
```

```
        System.out.println(hello());
    }
}
```

1.) 10

2.) 10 10

3.) Compile time error

4.) Blank output

Answer: 2.) 10 10

Question 29

“this” keyword is implicitlt present in:(select all the options that apply)

- 1) instance method
- 2) static method
- 3) default constructor
- 4) parameterized constructor

Answer: 1,3,4

Question 30

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

```
{
        final int x;
        x = x+3;
        System.out.println(x);
    }
}
```

1.) 1

2.) Blank output

3.) Compile-time error

4.) 4

Answer: 3.) Compile-time error

Question 31:

When Overloading does not occur?

a) More than one method with same name but different method signature and different number or type of parameters
b) More than one method with same name, same signature but different number of signature
c) More than one method with same name, same signature, same number of parameters but different type
d) More than one method with same name, same number of parameters and type but different signature

Answer: d) More than one method with same name, same number of parameters and type but different signature

32.

Which concept of Java is a way of converting real world objects in terms of class?
ab) Encapsulation
c) Abstraction
d) Inheritance

Answer: c) Abstraction

33.

When the instance variables memory is allocated in RAM ?

a) When object is created
b) When class is loaded
c) When constructor is called
d) When we execute the program

Answer: a) When object is created

34.

Which concept of OOP encapsulates the properties (data) and behaviors (methods) of a real-world object into a single entity?

a) Abstraction
b) Encapsulation
c) Inheritance
d) Polymorphism

Answer: b) Encapsulation

35.

A programmer defines a class but fails to create an instance of that class. They try to call a method of the class directly.

What type of error will occur?

a.) Compile-time error

b.) Logical error
c.) Runtime error
d.) No error

Answer: a)Compile-time error

36.

If a class Car has a method drive() that requires no arguments, which of the following calls is correct after creating a Car object named myCar?

a.) Car.drive()

b.) myCar.drive()

c.) drive(myCar)

d.) Car().drive()

Answer: b) myCar.drive()

37.

Given the following Java code, what is the output?

```
class Box {
int width;
Box(int w) {
width = w;
}
}
public class Test {
public static void main(String[] args) {
Box myBox = new Box(5);
```

```
System.out.println(myBox.width);
}
}
```

a.) 5
b.) 0
c.) Error
d.) None

Answer: a.) 5

38.

What is the result of compiling and running the following code in Java?

```
class Counter {
private int count = 0;
void increment() {
count++;
}
int getCount() {
return count;
}
}
```

a.) Error
b.) 0
c.) Compile-time error
d.) None of the above

Answer: d.) None of the above

39.

Identify the mistake in the following Java code snippet:

```
class Student {
private int score;
public static void setScore(int s) {
score = s;
}
}
```

a.) score should not be private

b.) s is not correctly passed to score

c.) setScore cannot be static as it accesses an instance variable

d.) None of the above

Answer: c.) setScore cannot be static as it accesses an instance variable

40.

A programmer attempts to compile the following Java code but encounters an error:

```
class MyClass { void MyClass() { } }.
```

What is the mistake?

A The constructor has a return type

B The class name is incorrect

C The constructor is defined as a method

D None of the above

Answer: C) The constructor is defined as a method

41.

Which keyword is used in Java to inherit a interface?

A extends
B implements
C inherits
D super

Answer: B implements

42.

Which keyword is used in Java to inherit a class?

A extends
B implements
C inherits
D super

Answer: A extends

43.

Polymorphism in OOP allows for:

A objects to take on multiple forms

B methods to execute concurrently

C classes to encapsulate data and methods

D objects to be instantiated without classes

Answer: A objects to take on multiple forms

44.

What does the super keyword in Java do?

A It is used to call superclass methods

B It defines a superclass

C It overrides a method in the superclass

D It creates an instance of a superclass

Answer: A It is used to call superclass methods

45.

Which of the following best describes "dynamic polymorphism"?

A Compile-time method binding

B The ability to create dynamic classes at runtime

C Method overriding where method calls are resolved at runtime

D The creation of objects at runtime

Answer: C Method overriding where method calls are resolved at runtime

46.

What is an "abstract class" in Java?

A A class that cannot be instantiated and must be inherited

B A class that can only contain abstract methods

C A class without any implementation

D A final class that cannot be extended

Answer: A A class that cannot be instantiated and must be inherited

47.

To achieve polymorphism in Java, one must use:

A Interfaces only

B Abstract classes only

C Either interfaces or abstract classes

D Direct class inheritance only

Answer: C Either interfaces or abstract classes

48.

Which feature distinguishes interfaces from abstract classes in Java?

A Interfaces cannot have any method implementations

B Interfaces can have only final static fields

C Abstract classes can contain non-final methods

D Interfaces support multiple inheritance

Answer: D Interfaces support multiple inheritance

49.

What is the output of the following code snippet in Java?

```
class Animal {
public void sound() {
System.out.println("Animal sound");
}
```

```
}
class Dog extends Animal {
public void sound() {
System.out.println("Bark");
}
}
public class Test {
public static void main(String[] args) {
Animal myAnimal = new Dog();
myAnimal.sound();
}
}
```

A "Animal sound"
B "Bark"
C Compilation error
D None of the above

Answer: B "Bark"

50.

Given the following Java code, which principle is demonstrated?

```
interface Flyable {
void fly();
}
class Bird implements Flyable {
public void fly() {
System.out.println("Bird flies");
}
}
```

```
class Airplane implements Flyable {
public void fly() {
System.out.println("Airplane flies");
}
}
```

A

Inheritance

B

Encapsulation

C

Polymorphism

D

Abstraction

Answer: C. Polymorphism

51.

In Java, which method signature indicates that a method is overriding a method from its superclass?

A. A method with the same name but a different return type

B. A method with the same name and parameter list as in the superclass

C. A static method with the same name as in the superclass

D. A method with the same name but different parameter types

Answer: A method with the same name and parameter list as in the superclass

52.

What will happen if a subclass in Java tries to override a superclass method marked as final?

A

The subclass method will successfully override the superclass method

B
The compiler will generate an error
C
The superclass method will be hidden
D
None of the above

Answer: The compiler will generate an error

53.
Given two classes, Base and Derived, where Derived extends Base and both classes have a method show() with different implementations.
How is polymorphic behavior achieved when calling show() on a Derived object referenced by a Base type?

A
By marking show() in Base as final
B
By using the static keyword in Derived's show() method
C
By overriding show() in Derived
D
None of the above

Answer: C
By overriding show() in Derived

54.

What will be the output of the following Java code snippet?
class Encapsulate {
private int num = 10;
public int getNum() {
return num;
}
} public class Test {
public static void main(String[] args) {

```
Encapsulate obj = new Encapsulate();
System.out.println(obj.getNum());
}
}
```

A 10
B 0
C Compilation error
D Runtime error

Answer: A) 10

55.

What is the output of the following Java program?

```
public class Vehicle {
    public void move() {
        System.out.println("The vehicle moves");
    }
}

public class Car extends Vehicle {
    public void move() {
        System.out.println("The car moves");
    }
}
public class Main {
    public static void main(String[] args) {
```

```
        Vehicle vehicle = new Car();
        vehicle.move();
    }
}
```

A. "The vehicle moves"
B. "The car moves"
C. The code does not compile
D. None of the above

Answer:

B. "The car moves"

56.

What is the output of the following Java program?

```
class Parent {
    String name = "parent";
    String message() {
        return "from parent";
    }
}

class Child extends Parent {
    String name = "child";
    String message() {
        return "from child";
    }
```

```
}

public class Main {
    public static void main(String[] args) {
        Parent p = new Child();
        System.out.println(p.name + " " + p.message());
    }
}
```

A. "parent from parent"

B. "child from child"

C. "parent from child"

D. "child from parent"

Answer:

C. "parent from child"

57.

```
package main;
class Base {
        public void Print()
        {
                System.out.println("Base");
        }
}
```

```
class Derived extends Base {
	public void Print()
	{
		System.out.println("Derived");
	}
}

class Main {
	public static void DoPrint(Base o)
	{
		o.Print();
	}
	public static void main(String[] args)
	{
		Base x = new Base();
		Base y = new Derived();
		Derived z = new Derived();
		DoPrint(x);
		DoPrint(y);
		DoPrint(z);
	}
}
```

A. Base
   Derived
Derived

B. Base

Derived

C. compile-time error

D. exception

Answer: A

58.

Which of these attributes can we use if we want to get the actual size of any array?

a. Array_name.length

b. Size.Array

c. length.Array_name

d. Array.Size

Answer: (a) Array_name.length

59.

What is the output of the following Java code snippet?

```
public class MyClass {
   private int number = 10;
   public void display() {
      System.out.println(number);
   }
}
```

```
public class Main {
    public static void main(String[] args) {
        MyClass obj = new MyClass();
        System.out.println(obj.number);
    }
}
```

A. 10

B. Error: Access Denied

C. 0

D. Compilation Error

Answer: D. Compilation Error

60.

Which access modifier restricts access of a class member to only its subclasses and classes in the same package?

A. Private

B. Default

C. Public

D. Protected

Answer: D. Protected

Problem 1.

Create an Employee class with fields Employee Name, Employee Name, Employee experience, salary and company he belongs to.

Assign skill and salary to each based on the experience and skill:

If an employee experience is in the range of 0-2 and skill is not java : it should be <30000. and

if it is Java: it should be between 30000 and 50000.

If an employee experience is in the range of 2-5 and skill is not java: it should be between 30000 and 50000

and if skill is Java : then it should be in between 50000 to 70000.

If an employee experience is in the range of 5-10 and skill is not java: it should be between 50000 and 70000

and if skill is Java : then it should be in between 70000 to 100000

and if experience is 10+ and skill is not java it should be less than 120000

and if it is Java it should be above 150000

Problem 2.

Write a Java class BankAccount that has attributes name, address, account_number

and balance, and methods to deposit and withdraw money

Problem 3.

Given an integer n, return the number of prime numbers that are strictly less than n.

Example 1:

Input: n = 10

Output: 4

Explanation: There are 4 prime numbers less than 10, they are 2, 3, 5, 7.

Example 2:

Input: n = 0

Output: 0

Example 3:

Input: n = 1

Output: 0

Constraints:

0 <= n <= 5 * 106