CORE JAVA (THEORY)
INTERVIEW QUESTIONS

BY- RAHUL CHAUHAN

TOPIC-1. JAVA BASICS

1. What is the programming language?
2. Who created Java programming language and when? And why?
3. What is platform Independency?
4. Why Java is a secure programming language?
5. What is the difference between JRE & JVM?
6. What is JDK?
7. Is java compiled or interpreted?
8. Is java slow then C programming language?
9. Write down the minimal tools required for creating the first java application?
10. What is the name of the compiler used in java?
11. How many methods we have in java to print something on the console?
12. What is the difference between printf and println method?
13. What is the difference between format & printf method?
14. What is the difference between print & println method?
15. What is the token? Explain all its types.
16. What are the rules of an identifier?
17. What is Literal and its types?
18. What is DataType?
19. Write down all primitive DataTypes?

20. Write down all non-primitive DataTypes?

21. What is the memory size of “char”?

22. What is Unicode?

23. Difference between ASCII code and Unicode?

24. Write down all the list of operators in java(category size)?

25. What are Uniary, Binary, and Ternary Operators? Explain with syntax.

26. Is it possible to use increment operators with data directly? Ex. 3++;

27. Is it possible to use ++ or -- operator with character and floating-point data variables?

28. What is TypeCasting?

29. Give the syntax of Implicit and explicit Typecasting?

30. What is upcasting and downcasting?

31. Write down the typecasting flow diagram?

32. What is widening & Narrowing?

33. Explain the meaning of System.out.println();

34. Is it possible to leave print method blank? Ex. System.out.print();

35. Is it possible to leave println method blank? Ex. System.out.println();

36. Differences between: "System.in" and "System.out" and "System.err".

37. What is the decision-making statement?

38. What are the types of decision-making statements?

39. What are the allowed data types in switch condition and switch cases?

40. What is the loop control statement?

41. What are the types of the loop?

42. What is enhanced for loop?

43. How can you exit anticipatedly from a loop?

44. Is it possible to use a break in if or else statement?

45. What is a labeled and unlabeled loop? Explain with the program.

46. Write a program that explains the need for "for-loop" and "while-loop"?

47. Difference between "for-loop" and "enhanced for-loop"?

48. What is an infinite Loop?

49. What is the difference between double and float variables in Java?

50. Write a program to generate random numbers in Java?

51. Can we use “goto” in Java?

52. Is JDK required on each machine to run a Java program?

53. List any five features of Java?

54. What is the Java API?

55. Why java does not support global variables?

TOPIC-2. OOP BASICS

1. What is OOP mean?
2. What are the concepts that come under the OOP?
3. Differentiate Encapsulation & Abstraction?
4. Who developed OOP concepts?
5. Is it possible to use “;” at the end of class?
6. Describe all types of Initializers in a class?
7. How garbage collection is done in Java?
8. Which method is used to garbage collect an object?
9. Can we have two methods in a class with the same name?
10. Can we have same method name and same variable name in a class?
11. Can we have method name same as class name?
12. What’s the difference between comparison done by equals method and == operator?

TOPIC-3. STATIC KEYWORD

1. What is static-initializer?
2. Is it possible to make a class static?
3. Is it possible to create a local static variable?
4. Write a program that contains static-variable and static-method?
5. When exactly the static variable and non-static variable (instance variable) got memory allocation?
6. If we are having two variable (static **x** & non-static **y**) in class and creating 5 instances of that class. Then tell how many times x and y variable created in memory?
7. Is it possible to directly access the non-static member inside the static initializer of the same class?
8. Is it possible to directly access the non-static member inside the static method of the same class?
9. Write a program to access the non-static member of a class inside the static method of the same class.
10. Is it possible to overload two static methods? If yes then write a program.
11. Is it possible to declare more than one static initializer in a class? If yes then write a program and output.
12. Is it possible to use "**this.**" With the static variable?

TOPIC-4. CONSTRUCTOR

1. What is Constructor?
2. Differentiate constructor and method?
3. Is it possible to write a return type with Constructor?
4. Is it possible to pass the argument in the constructor?
5. Is it possible to have a different name of the contractor than the class name?
6. Is it possible to make a constructor private or protected?
7. What is default access modifier of constructor?
8. Is it possible to have more than one constructor in a class?
9. Write a program in which two constructors executed with one instance.
10. Name the keyword used to chain two constructors in a class.
11. What is constructor overloading?
12. Write a program of constructor overloading.
13. Is it possible to make a constructor "static" or "abstract" or "final"?
14. Is it possible to override the constructor?
15. Is it possible to call constructor explicitly?
16. What is the default constructor?
17. Write the syntax of the default constructor.
18. It is possible to directly access the static-member inside, the constructor of the same class?

19. Is it possible to declare more than one initialize block in a class? If yes then write a program and output.

20. What is a singleton class? Explain with practical example and use.

21. Is it possible to write more than one constructor chaining statement in a constructor?

22. Can we call the constructor of a class more than once for an object?

TOPIC-5. INHERITANCE

1. What is Inheritance?
2. What are the benefit of using inheritance?
3. What are sub-class and super-class?
4. Does the instance/object of the sub-class contain the memory of the private-member of the super-class?
5. Is it possible to access the private member of the super-class inside the sub-class?
6. Is it possible to call the constructor of super-class inside the sub-class without creating a separate object of super-class?
7. Is it possible to write **this()** and **super()** both statements together in a constructor?
8. Name the class, which is the default super-class of every java class.
9. What is IS-A ad HAS-A relation? Explain with programs.
10. Is it possible to inherit a class that has only one constructor which is private?
11. Is it possible to inherit a class that has only one constructor that is protected?
12. Is it possible to inherit more than one class in a class?
13. Does static variable and method involve in inheritance?
14. Is it possible to use "**this.**" and "**super.**" Inside the static context of a class?
15. Is it possible to use "**super.**" with a static variable?
16. What type of inheritance is not supported in java classes and why?
17. Define all types of inheritance?
18. What is toString() method?

19. Can a class be a super-class and a sub-class at the same time? Give example.

20. How can we restrict the inheritance of a class?

21. What is the order of constructor calling in inheritance?

22. Differentiate Composition and Inheritance.

TOPIC-6. POLYMORPHISM

1. What is method overloading?
2. What are the rules for method overloading?
3. Is it possible to overload the static method?
4. Does Java support operator overloading?
5. What is the working of "**instanceof**" keyword?
6. What is method overriding?
7. What are the rules of method overriding?
8. Is it possible to override data?
9. Is it possible to override the static method?
10. Is it possible to override constructor?
11. What is data hiding?
12. What is method hiding?
13. What are the rules for method hiding?
14. Is it possible to override the private method?
15. Is it possible to override the protected method?
16. What is polymorphism?
17. What is compile-time polymorphism?
18. What is runtime polymorphism?
19. Write a program for compile-time polymorphism?

20. Write a program for runtime polymorphism?
21. Is it possible to achieve compile-time polymorphism with inheritance? If yes, then write a program.
22. Is it possible to achieve runtime polymorphism without inheritance?
23. What is the covariant return type? Explain with a program.
24. What is cohesion? Explain with a program.
25. What is coupling? Explain with a program.
26. What is the abstract class?
27. What is the abstract method?
28. Is it possible to make a variable abstract?
29. Is it possible to make a class abstract and final both?
30. Is it possible to make a method abstract and final both?
31. Is it possible to make a method static and abstract both?
32. Is it possible to make a method private and final both?
33. Is it possible to make a method private and abstract both?
34. Is it possible to make a method static and final both?
35. What is non-primitive upcasting?
36. What is non-primitive down-casting?
37. Is it possible to hold the object of super-class in the reference of the sub-class?
38. Is it possible to hold the object of the sub-class in the reference of the super-class?
39. Is it possible to access the member of the super-class from the reference of the sub-class?

40. Is it possible to access the member of the sub-class from the reference of the super-class?

41. Is it true, method calling bound at compile-time but executed at runtime?

42. Is it true, method calling bound and executed at compile time?

43. What is the interface?

44. Differentiate the abstract class and interface?

45. What is the keyword used to inherit an interface in class?

46. What is the keyword used to inherit an interface in the interface?

47. Is it possible to create the instance/object of the abstract class directly?

48. Is it possible to create the instance/object of the interface?

49. Does abstract class support constructor?

50. Does interface support constructor?

51. Is it possible to declare a static method and a static variable in abstract class?

52. Is it possible to declare a static method and a static variable in the interface?

53. Does an interface is also the child of object class?

54. Is it possible to declare a private variable in the interface?

55. Is it possible to declare a private method in the interface?

56. Is it possible to declare a non-static method with body in the interface?

57. Is it compulsory to use abstract keyword to declare abstract method in the interface?

58. Is it possible to use abstract keyword with the interface?

59. Is it possible to inherit more than one interface in a class?

60. Is it possible to inherit more than one interface in an interface?

61. Can we declare a class as Abstract without having any abstract method in it?

62. Is it possible to inherit a class in an interface?

63. Is it possible to inherit both (a class and an interface) in a class?

64. When you are inheriting a class and an interface than is it true, that class inheritance must come first?

65. Is it possible to declare a static nested interface?

66. Is it possible to make an interface final?

67. Is it possible to declare a private nested interface?

68. Can we override a method by different return types?

TOPIC-7. FINAL KEYWORD

1. What is the final class?
2. What is the final method?
3. What is the final variable?
4. Is it possible to declare the final local variable?
5. What happens, if the final variable is not initialized?
6. Does the final variable receive default value?
7. Does the local variable receive default value?
8. What is the blank final variable?
9. Is it possible to declare a non-static final variable and initialize it from the static block?
10. Is it possible to declare a static final variable and initialize it from the constructor?
11. Is it possible to declare a static final variable and initialize it from the static block?
12. Is it possible to declare a non-static final variable and initialize it from the constructor?

TOPIC-8. NESTED CLASS

1. What is Nested Class?
2. What is the difference between Nested Class and Sub Class?
3. Write the signature of creating the object of non-static (inner) nested class from outside the outer/upper class?
4. Write the signature of creating the object of static Nested Class from outside/upper class?
5. Is it possible to declare a nested class private or protected?
6. Is it possible to access the private data of the outer class inside the inner class directly?
7. Is it possible to access the non-static private data of the outer class inside the static nested class directly?
8. Is it possible to access the non-private member of the inner class from the reference of the outer class?
9. Is it possible to access the non-private member of the outer class from the reference of the inner class?
10. Is it possible to declare the static variable in the inner class? If yes, then write a program.
11. Is it possible to declare a static method and static initializer inside the inner class?
12. Is it possible to declare the static method and static initialize inside the inner class?
13. Is it possible to declare the static initializer inside the static nested class?
14. Is it possible to access a local nested class from outside that local block?
15. Is it possible to access the private data of the outer class inside the local nested class?

16. What is an anonymous nested class?

17. Is it possible to declare a variable inside the anonymous nested class?

18. Is it possible to declare a non-overridden method inside the anonymous nested class?

19. Is it possible to declare a class inside the anonymous nested class?

20. Write the signature of creating the object of anonymous nested class?

21. Is it possible to declare a static variable in the anonymous nested class?

22. How many classes the following program contains? Write the byte-code names of all classes:

```
class A{

class B{ }

void m(){

    class C{}

    class B{}

}

A(){

    class B{}

    class E{}

    B b=new B(){};

    B b2=new B();

}

B b=new B(){};

}
```

TOPIC-9. PACKAGES

1. What is the role of packages?
2. What is the name of by-default imported package?
3. What is the import keyword?
4. Is it possible to have an import statement before the package statement?
5. Differentiate import and import static?
6. Is it possible to access the default member of a public class outside the package?
7. Is it possible to access the public member of the default class outside the package? If yes, then write the program.
8. Is it possible to access the protected member of a class outside the package?
9. Is it possible to declare a method final and private both?
10. Write the compilation command that creates the byte-code and package folder also?
11. Does Importing a package imports its sub-packages as well?
12. There are two classes named A and B. Both classes are in the same package. Can a private member of A can be accessed by an object of B?
13. What's the access scope of Protected Access modifier?
14. What is package private access specifies means?
15. Is it possible to have multiple public classes in a single java source file?

TOPIC-10. EXCEPTION HANDLING

1. What is Exception?
2. Differentiate Exception & Error?
3. What is Exception Handling?
4. Is it possible to write a catch block without a try block?
5. Is it possible to write a try block without a catch block?
6. Is it possible to write finally block without a catch block?
7. Is it possible to write finally block without a try block?
8. What is finally block?
9. Is it possible to write more than one catch block with a try block?
10. Is it possible to write more than one finally block with a try block?
11. Is it possible to declare the final variable inside try or catch or finally block?
12. What happens, when there is a try-catch block but it does not contain any exception throwing statement?
13. What is reported (unchecked) and unreported(checked) exceptions?
14. Differentiate throw and throws?
15. If we are using throw then is it compulsory to use throws?
16. If we are using throws then is it compulsory to use throw?
17. What is try with Resource?
18. What are the rules of throws in method overriding?

19. What is exception propagation?

20. What is exception chaining?

21. Is there any way to skip execution of Finally block?

22. What is the base class of all exception classes?

23. What is the base class of exception hierarchy?

TOPIC-11. ARRAYS

1. What is an Array?
2. What is the primitive Array?
3. Why does the Array index start from 0?
4. What type of loop control designed specialty for array & collections?
5. What is the default value of an array type variable?
6. What is the default value of array elements?
7. Is it possible to declare an array with a negative size?
8. Is it possible to declare an array with a floating-point value size?
9. Is it possible to store an array of int type in double type array variable?
   Double a[]=new int[]{2,6};
10. Is it possible to change the size of an array at runtime?
11. Is it possible to access an element of an array by providing a negative index?
12. Is it possible to pass floating-point or string value as an index?
13. Which statements are valid?

    I. Integer a[ ]={2,6,5};

    II. int a[]={2,6,5};

    III. int a[ ]=new int[]{2,6,5};

    IV. int a[]=new Integer[]{2,6,5};

    V. Integer a[]=new int[]{2,6,5};

VI. double a[]={2,6.1,5.0};

VII. Double a[]={2,6.1,5.0};

14. Write a program to print the sum of all the elements of an int type array?

15. Write a program to print maximum value element of an int type array?

16. Is this statement valid? int a[]=new int[3]{2,6,5};

17. Which statements are valid?

I. int a[][] = new int[2][];

II. int a [][] = new int[2][5];

III. int a[][] = new int[][];

IV. int a[][] = new int[2][1];

V. a[1] = new int[3];

VI. int a[][] = new int[2][3];

VII. System.out.println(a[0]);

VIII. System.out.println(a[0][1]);

18. Is this valid?

double a[][] = new double [2][];

a[1] = new int [3];

19. Write the program to print both the diagonal of a 2-D int type Array?

20. Name the method that sort the array elements in one go?

TOPIC-12. STRING

1. What is the String?

2. What is the name of the package that contains the String class?

3. How many constructors String class has?

4. When we create a string class object, internally a char type array created or not?

5. What is the difference between the following statements?

   String s1= “Incapp”

   String s2 = new String (“Incapp”);

6. What is the mutable and immutable String?

7. What is the role of the StringTokenizer class?

8. What are the differences between StringBuilder and StringBuffer class?

9. How many total string objects created?

   String s= “Hello”;

   s= s+ “Hi” +5;

10. How many total string objects created?

    String s= “hello”;

    s=s.concat(“Hi”).concat(5);

11. How many String Objects are created in the following code?

    String s1="Java Language";

    String s2="C Language";

String s3="Java Language";

12. When lots of changes are required in string data, which one class should be used? String or StringBuffer?

13. Can we compare String and StringBuffer objects in Java?

TOPIC-13. WRAPPER CLASS

1. What is a Wrapper class?

2. Write down all the wrapper classes?

3. What is Boxing? Write the syntax?

4. What Unboxing? Write the syntax.

5. What is AutoBoxing? Write the syntax.

6. What is parsing? Write the syntax.

7. What is the difference between-

   Integer.parseInt()

   Integer.valueOf()

   Integer.decode()

8. Are wrapper classes immutable? If yes then why?

9. What is the output?

```
Integer a=127, b=127;

System.out.println(a==b);

Integer x=128, y=128;

System.out.println(x==y);

Integer p=new Integer(100);

Integer q=new Integer(100);

System.out.println(p==q);
```

10. Write down the ways to convert a primitive type to a wrapper object.

11. Write down the ways to convert wrapper object to primitive type.

TOPIC-14. VAR-ARGS

1. What is var-args?
2. Is this a valid statement?

   int … a={10, 2,3};

3. Is it possible to have two var-args in an argument list?
4. Is it possible to have var-args as the first argument in an argument list?
5. Is it possible to create var-args of object type?
6. What is the command-line argument?

TOPIC-15. PUBLIC STATIC VOID MAIN

1. What is the significance of "public" and "static" in the main-method?
2. What happens, when the main-method does not have "public" and "static" keyword?
3. Is it possible to overload the main-method?
4. Is it possible to have the main-method in more than one class of a program?
5. Can main() method in Java can return any data?
6. Can we declare the main method as private?
7. Can we execute any code before main-method?
8. Is it possible to compile a java program successfully without even having a main method in it?

incapp

CONTACT DETAILS:

Address: 5th Floor, Om Tower, Alpha 1,
Commercial Belt, Greater Noida, UP, INDIA.

Phone: 9811272031, 01204108484

Email: info@incapp.in

Website: www.incapp.in