Core java interview Questions.

**OOPS Concepts and Keywords:**

1. What are the characteristics of object?(T)
2. //Difference between overloading and overriding
3. //Define OOP concepts in java:-
4. //How can objects be organized into a hierarchy?
5. How is inheritance put to use in java:- (d)
6. //What happens if you remove static modifier from main() method?
7. Is JVM platform independent? (t)
8. What is JVM, JRE and JDK? (t)
9. What is the difference between JDK and JVM? (t)
10. What is pointer? Why java doesn’t support pointers? (t)
11. What is the difference between path and classpath? (t)
12. Explain public static void main(String[] args):- (t)
13. //Can we declare main() method as final?
14. //Can a source file contain more than 1 class declaration?
15. //Can a class be declared as protected?
16. //What is the impact of declaring a method as final?
17. //How can I prevent my class to be inherited by some other class?
18. Can a class be declared static? (d)
19. What are the restrictions imposed on a static method or static block of code?(t)
20. //How can I print something even before main() method is executed?
21. Examples of final class defined in java API?(t)
22. Difference between parameter and arguments(t)
23. What is Unicode?(t)
24. What are transient and volatile modifiers?
25. //How do you allocate memory to an object?
26. What is the use of bin and lib in JDK?(t)
27. //What is garbage collection and how can we call it?
28. What is difference between this() and super()?
29. Why java is platform independent?(t)
30. Why java is not pure OOP language?(t)
31. //Difference b/w early binding and late binding
32. //Can we call main method from static block ?

**Stream :**

1. Difference between Collection API and Stream API.
2. When does Java throws IllegalStateException with respect to Streams?
3. What is the benefit of using Parallel Stream?
4. How does Parallel Stream work?
5. How can we use streams with primitives?
6. What is collect() method in Java Stream?
7. What are the types of Stream operations?
8. What are Stateless and Stateful operations in Java Stream?

**Collections:**

1. What are concurrent Collection Classes?
2. What is difference between Comparable and Comparator interface?
3. What is difference between ArrayList and LinkedList?
4. What is different between Iterator and ListIterator?
5. Why Map interface doesn’t extend Collection interface?
6. Differentiate between HashMaps and HashTables
7. What are Generics? Why do we use them?
8. Benefits of Generics
9. What is wildcard in Generics?
10. What are the different forms of Generics
11. What are the features of collections

**Lambda Expression**

* 1. Why lambdas expressions?
  2. How lambdas support functional programming?
  3. OOP Vs Functional Programming.
  4. How lambdas achieve function inline?
  5. How can you implement lambdas in legacy Java code?
  6. What is the target type inference inherited by lambdas?
  7. How will you override methods of in-built functional interfaces?
  8. How will you call default method of an interface?
  9. Advantages of lambdas over inner classes?
  10. How does Java 8 support backward compatibility with legacy integration?