Interview Questions:

1. **Why java is not a 100% object oriented programming language.**

Because of primitive datatype java is not 100% object oriented programming language.

2. **Why pointers are not allowed in java?**

In java Jvm is responsible for the memory allocation in java to avoid the direct access of the user to the memory pointers are discouraged in java.

3. **What is Jit compiler in java?**
4. **Why String is immutable in java?**

Because of String constant pool string is immutable in java, In string constant pool once an object is stored then multiple references can points to the same object if they are having same content in their object instead of creating a new object they will refer to the object present in the string constant pool because of this reason string is immutable. If immutability is not there then if change the object content for one reference then for all other which are pointing towards that object their content will also get change which will rise a data inconsistency issue.

5. **What is a marker interface?What is the need of marker interface?**

- An empty interface is known as marker interface.
- They provide the runtime type information about the object to compiler so that compiler have some additional information about objects.

e.g: clonable,serializable,Rendom Access.

6. **Can you Override private or static method in java?**

- No we can’t override private and static methods in java.

7. **Does finally always execute in java?**

- Finally is a block associated with try for cleanup activity.
- If we use System.exit then finally block will not be executed because this will stop the jvm explicitly from the execution of program.

8. **What methods the object class have?**

- Clone()//it will create a copy of the object
- equals()//it will compare two objects
- Finalize()//Invoked by garbage collector before destroying an object to perform cleanup activity.
- hashCode()//it will return the hashcode of an object
- wait()//
- notify()//
- notifyAll()//
- getClass()//return the current running class
- toString()//return the string representation of an object

9. **How to create a Immutable Class?**

- Make the class final so that no one can inherit that class.
- Make all the fields private so that no one can access the datamembers directly.
- Don't provide setters for variables so that no one can set the values.
- Make all mutable fields final so the value can be assigned only once.
- Initialize the field with a constructor performing a deep copy.
- Perform cloning on the getter method so that it will return a copy of the reference rather than the actual reference.

10. **What is Singleton class in java and how can we make a singleton class?**
11. **Explain collection hierarchy?**
12. **What are wrapper classes?**
13. **What are constructors in java?**
14. **What is default constructor?**
15. **What is parameterized constructor?**
16. **Explain java 8 features?(default,FunctionalInterface,Lamdas,Optional,streamApi)?**
17. **What is abstract class?**
18. **What is Interface?**
19. **Difference between interface and abstract class?**
20. **Why we use interface over abstract class to achieve abstraction?**
21. **What is the use of constructor in abstract class?**
22. **Explain checked and unchecked exception?**
23. **Parent class of Exception?**
24. **Explain Error?**
25. **Difference between error and exception?**

26.**Difference between Throw and Throws?**

27.**Can we write try without catch block?**

28.**What should be the hierarchy if we write multi catch block?**

29.**Exception propagation?**

30.**Share some example of check and unchecked exception?**

31.**String vs StringBufffer vs StringBuilder?**

32.**Why string is Immutable in java?**

33.**Difference between final,finally and finalize?**

34.**Wrapper classes?**

35.**Serialaization and deserialization?**

36.**Jdk vs jre vs jvm?**

37.**Explain public static void main(String args[])**

38.**What is command line argument?**

39.**Why main method is static?**

40.**Sib vs iib?**

41.**Stack vs heap memory?**

42.**Describe garbage collector?**

43.**What is optional class?**

44.**Before java 8 how null pointer exceptions are handled?**

45.**When null pointer exception occurs?**