1) What is meant by Exception?

Answer:

An Exception is a problem that can occur during the normal flow of execution.

A method can throw an exception when something wails at runtime.

If that exception couldn’t be handled, then the execution gets terminated before it completes the task.

If we handled the exception, then the normal flow gets continued. Exceptions are a subclass of java.lang.Exception.

2) What are the types of Exceptions?

Answer: There are two types of Exceptions.

a) Checked Exception:

These exceptions are checked by the compiler at the time of compilation. Classes that extend Throwable class except Runtime exception and Error are called checked Exception.

Checked Exceptions must either declare the exception using throws keyword (or) surrounded by appropriate try/catch.

For Example, ClassNotFound Exception.

b) Unchecked Exception:

These exceptions are not checked during the compile time by the compiler. The compiler doesn’t force to handle these exceptions.

It includes: Arithmetic Exception, ArrayIndexOutOfBounds Exception

3) What are the different ways to handle exceptions?

Answer:

a) Using try/catch:

The risky code is surrounded by try block.

If an exception occurs, then it is caught by the catch block which is followed by the try block.

b) By declaring throws keyword:

At the end of the method, we can declare the exception using throws keyword.

4) What are the advantages of Exception handling?

Answer:

The normal flow of the execution won’t be terminated if an exception gets handled

We can identify the problem by using catch declaration

5) What are the Exception handling keywords in Java?

Answer:

a) try:- When a risky code is surrounded by a try block.

An exception occurring in the try block is caught by a catch block.

Try can be followed either by catch (or) finally (or) both. But any one of the blocks is mandatory.

b) catch:- This is followed by a try block. Exceptions are caught here.

c) finally:- This is followed either by try block (or) catch block.

This block gets executed regardless of an exception. So generally clean up codes are provided here.

6) Explain about Exception Propagation.

Answer:

Exception is first thrown from the method which is at the top of the stack.

If it doesn’t catch, then it pops up the method and moves to the previous method and so on until they are got.

7) What is the final keyword in Java?

Answer:

Final variable: Once a variable is declared as final, then the value of the variable could not be changed. It is like a constant.

Final method: A final keyword in a method, couldn’t be overridden. If a method is marked as a final, then it can’t be overridden by the subclass.

Final class: If a class is declared as final, then the class couldn’t be subclassed. No class can extend the final class.

8) What is a Thread?

Answer:

In Java, the flow of execution is called Thread. Every java program has at least one thread called the main thread, the main thread is created by JVM.

The user can define their own threads by extending the Thread class (or) by implementing the Runnable interface. Threads are executed concurrently.

9) How do you make a thread in Java?

Answer:

a) Extend Thread class:- Extending a Thread class and override the run method. The thread is available in java.lang.thread.

The disadvantage of using a thread class is that we cannot extend any other classes because we have already extended the thread class. We can overload the run () method in our class.

b) Implement Runnable interface:- Another way is by implementing the runnable interface. For that, we should provide the implementation for the run () method which is defined in the interface.

10) Explain about join () method.

Answer:

Join () method is used to join one thread with the end of the currently running thread.

It is a non-static method. The Join () method has an overloaded version. So we can mention the time duration in join () method also “.s”.

11) What does the yield method of the Thread class do?

Answer:

A yield () method moves the currently running thread to a runnable state and allows the other threads for execution. So that equal priority threads have a chance to run. It is a static method. It doesn’t release any lock.

Yield () method moves the thread back to the Runnable state only, and not the thread to sleep (), wait () (or) block.

12) Explain about wait () method.

Answer:

wait () method is used to make the thread to wait in the waiting pool.

When the wait () method is executed during a thread execution then immediately the thread gives up the lock on the object and goes to the waiting pool.

Wait () method tells the thread to wait for a given amount of time.

Then the thread will wake up after notify () (or) notify all () method is called.

Wait() and the other above-mentioned methods do not give the lock on the object immediately until the currently executing thread completes the synchronized code.

It is mostly used in synchronization.

13) Difference between notify() method and notifyAll() method in Java.

Answer:

notify() notifyAll()

This method is used to send a signal to This method sends the signal to wake up

wake up a single thread in the waiting pool. all the threads in a waiting spool.