**B1. How are the exceptions handled in java?**

Customized Exception Handling : Java exception handling is managed via five keywords: try, catch, throw, throws, and finally. ... Any exception that is thrown out of a method must be specified as such by a throws clause. Any code that absolutely must be executed after a try block completes is put in a finally block.

**B2. Difference between Error and Exception in Java.**

An Error "indicates serious problems that a reasonable application should not try to catch." An Exception "indicates conditions that a reasonable application might want to catch." Error along with RuntimeException & their subclasses are unchecked exceptions. All other Exception classes are checked exceptions.

**B3. What are checked and unchecked exceptions?**

The main difference between checked and unchecked exception is that the checked exceptions are checked at compile-time while unchecked exceptions are checked at runtime.

**B4. How do we handle more Than One Type of Exception using catch block in Java?**

Java catch multiple exceptions

A try block can be followed by one or more catch blocks. Each catch block must contain a different exception handler. So, if you have to perform different tasks at the occurrence of different exceptions, use java multi-catch block.

**B5. What happens if an exception is thrown from the finally or catch block in Java?**

In normal case when there is no exception in try block then the finally block is executed after try block. However if an exception occurs then the catch block is executed before finally block. 4. An exception in the finally block, behaves exactly like any other exception.

**B6. Will the finally block be executed when the catch clause throws exception in Java?**

Java finally block is always executed whether exception is handled or not.

**B7. What is a user defined/custom exception in Java?**

User-defined Custom Exception in Java. Java provides us facility to create our own exceptions which are basically derived classes of Exception. For example MyException in below code extends the Exception class. ... The constructor of Exception class can also be called without a parameter and call to super is not mandatory.

**B8. Does a finally block always run in Java?**

The finally block will not be executed only if program exits (either by calling System.exit() or by causing a fatal error that causes the process to abort). Java finally block is always executed whether exception is handled or not.

**B9. Does return statement allow finally block to execute in Java?**

The answer is yes. finally block will execute. The only case where it will not execute is when it encounters System.exit().

**B10. Should a catch block always follow try block in Java for Exception handling?**

The try block contains set of statements where an exception can occur. A try block is always followed by a catch block, which handles the exception that occurs in associated try block. A try block must be followed by catch blocks or finally block or both.

**B11. Difference between Error and runtime exceptions in Java. B12. Difference between throw and throws clause in Java.**

Throws clause is used to declare an exception, which means it works similar to the try-catch block. On the other hand throw keyword is used to throw an exception explicitly. 2. If we see syntax wise than throw is followed by an instance of Exception class and throws is followed by exception class names.

**B13. Can try block exist without any catch and finally block in Java?**

Can we have try without catch block in java. Yes, we can have try without catch block by using finally block. You can use try with finally. As you know finally block always executes even if you have exception or return statement in try block except in case of System.exit().

**B14. What is stack trace?**

Java: Stack Traces. A stack trace is the list of methods that the program was in the middle of when the stack trace was printed. It's typically printed to the console when an unexpected error occurs (an exception is thrown but never caught).

**B15. What is the order of catch blocks when catching more than one exception?**

Points to remember. At a time only one exception occurs and at a time only one catch block is executed. All catch blocks must be ordered from most specific to most general, i.e. catch for ArithmeticException must come before catch for Exception.

**B16. Can we use FileNotFoundException and IOException in Java multi catch?**

Yes, as the javadoc shows it, FileNotFoundException is a subclass of IOException . If you really want FileNotFoundException , you must catch only this execption, otherwise catching IOException will also catch any exception subclassing it, like FileNotFoundException any many others.

**B17. Give few examples of checked exceptions.**

Example of checked exceptions are : ClassNotFoundException, IOException, SQLException and so on.

**B18. Give few examples of unchecked exceptions.**

Unchecked Exceptions are subclasses of RuntimeException. Example of unchecked exceptions are : ArithmeticException , ArrayStoreException , ClassCastException and so on.

**B19. Explain exception handling when overriding a method?**

How to handle exceptions in overriding methods?

Rule 4: If parent-class method declares combination of both checked & unchecked exceptions

1. Then child-class overriding-method can declare any type of unchecked exception.
2. Then child-class overriding-method can declare same type of checked-exception or one of its sub-class or no exception.

**B20. Can overridden method throw RuntimeException when original method throw ArithmeticException?**

So at compile time itself, the Java compiler checks the type of exception the overridden method is throwing. Since which overridden method will be executed can be decided only at runtime, we cannot know what kind of Exception we have to catch.

**B21. Can I write only try block without any catch and finally block?**

Yes, we can have try without catch block by using finally block. You can use try with finally. As you know finally block always executes even if you have exception or return statement in try block except in case of System.exit().

**B22. Difference between final, finally and finalize in Java.**

Final class can’t be inherited, final method can’t be overridden and final variable value can’t be changed. Finally is used to place important code, it will be executed whether exception is handled or not. Finalize is used to perform clean up processing just before object is garbage collected.

**B23. What is rethrowing an exception?**

An exception can be rethrown in a catch block. ... The purpose of the rethrow operation is to get the attention of the outside world that an exception has occurred and at the same time perform any contingency logic (such as logging) in the catch block.

**B24. Explain the rules of Exception Handling in terms of Method Overriding?**

Rule: An overriding method (the method of child class) can throw any unchecked exceptions, regardless of whether the overridden method (method of base class) throws exceptions or not. ... The overriding method can throw those checked exceptions, which have less scope than the exception(s) declared in the overridden method.