Java Interview Question and Answers

➢ Topic – Java Exception Handling:

ava Exception Handling Interview Questions and Answers

---

1. What is Exception in Java?
Ans: An exception is an error that affects the normal execution of program. If an exception is not handled, the program abruptly terminates.

2. What is the super or base class of all exceptions in Java?
Ans: Exception is the superclass of all exceptions in Java.

3. What is the superclass for error and exception classes in Java?
Ans: Throwable is the superclass for error and exception classes in Java.

4. What are the types of exceptions in Java API?
Ans: There are two types of exceptions available in Java API. They are:

a. **Predefined Exceptions (Built-in-Exceptions):** Predefined exceptions are those exceptions that are already defined by Java system. These exceptions are also called built-in-exceptions.
The built-in-exception is further divided into two categories: checked and unchecked exceptions.

b. **Custom Exceptions:** Custom exceptions are those exceptions that are created by users or programmers according to their own needs. The custom exceptions are also called user-defined exceptions that are created by extending the exception class.

5. Why an exception occurs in the program?
Ans: There can be many reasons that might generate an exception in a Java program.

- Opening a non-existing file in your program.
- Reading a file from a disk but the file does exist there.
- Writing data to a disk but the disk is full or unformatted.

- When the program asks for user input and the user enters invalid data.
- When a user attempts to divide an integer value by zero, an exception occurs.
- When a data stream is in an invalid format, etc.

6. What is Exception handling in Java?

Ans: The mechanism of handling unexpected errors in a java program is called exception handling. It is a powerful mechanism to handle runtime errors, ClassNotFoundException, FileNotFoundException, IOException, etc. so that the normal execution flow of the program can be maintained.

7. What is exception handler in Java?

Ans: The code that catches the exception thrown by JVM is called exception handler in Java. In other words, an exception handler is a block of code that performs an action when an exception occurs in a program.

8. What is the advantage of using exception handling in Java?

Ans: There are several advantages of using exception handling in java. They are:

- The main advantage of exception handling technique is to maintain the normal flow of the program.
- It provides flexibility in handling situations of errors.
- It allows us to define a user-friendly message to handle the exception.
- It helps to separate "Error-Handling code" from "Regular code."

9. Which of the following statements will throw an exception?

a. System.out.println(1/0);
b. System.out.println(2.0/0);

Ans: a will throw an exception named ArithmeticException. The second statement will give output infinity.

10. Which exception may be thrown if the given code is executed?

a.

```
public class Test

{

public static void main(String[] args)
```

```
{

int[ ] list = new int[4];

System.out.println(list[4]);

  }

}
```

Ans: ArrayIndexOutOfBoundsException will be thrown if the above code is executed.

b.

```
public class Test

{

public static void main(String[] args)

{

int a = 20;

int b = 30;

int c = 10;

int x = (a * b)/(a - b + c);

System.out.println("Result: " +x);

  }

}
```

Ans: ArithmeticException will be thrown if code is executed.

11. Explain the Java exception hierarchy.

Ans: The hierarchy of exception in Java is shown in the below diagram.

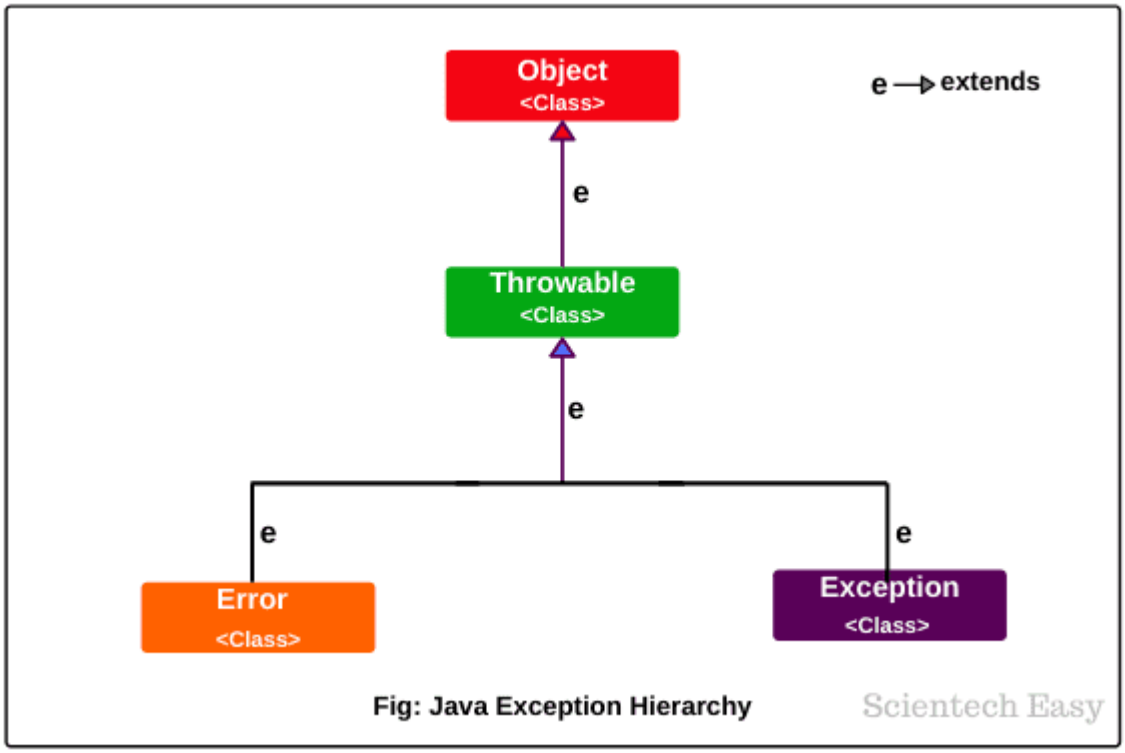

Fig: Java Exception Hierarchy

Throwable class is the parent class of all exception types. It is an immediate subclass of the Object class. Below Throwable class, there are two subclasses (two child objects) Error and Exception.

Error: Error class is the subclass of Throwable class and a superclass of all the runtime error classes. It terminates the program if there is a problem related to a system or resources (JVM).

Exception: An exception is an abnormal condition that is caused by runtime error in the program. It is the superclass of all exceptions in Java. It is further divided into checked and unchecked (runtime) exceptions.
The exception class hierarchy has been shown in the below figure.

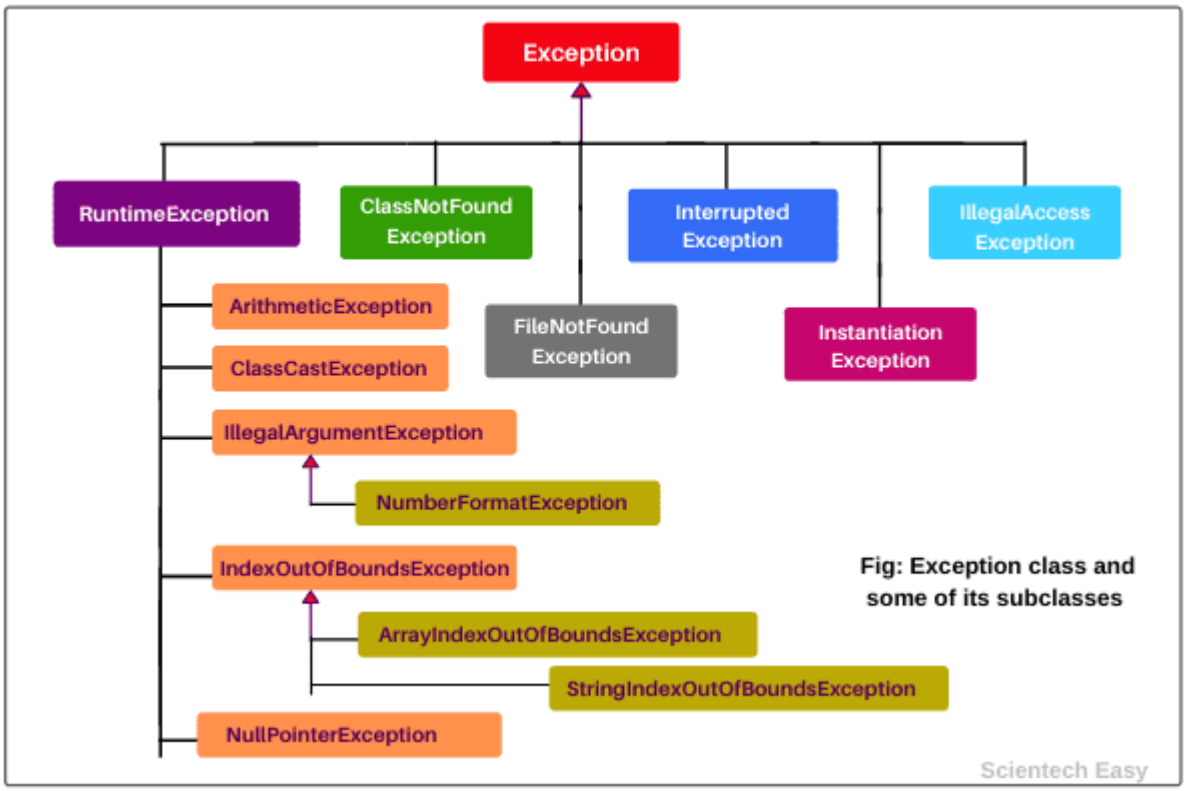

Fig: Exception class and some of its subclasses

Checked exceptions are those exceptions that are checked by Java compiler at compilation. A list of some important checked exceptions are given below:

- ClassNotFoundException
- InterruptedException
- InstantiationException
- IOException
- SQLException
- IllegalAccessException
- FileNotFoundException, etc

Runtime exceptions are those exceptions that are checked by JVM at runtime. Some important examples of runtime exceptions are given below:

- ArithmeticException
- ClassCastException
- NullPointerException
- ArrayIndexOutOfBoundsException
- NegativeArraySizeException
- ArrayStoreException
- IllegalThreadStateException
- SecurityException, etc.

12. What does JVM do when an exception occurs in a program?

Ans: When JVM faces an exception in a program, it creates an exception object and throws it to inform us that an error has occurred. If the exception object is not caught and handled properly, JVM will display an error message and will terminate the rest of the program abnormally.

13. How do we catch an exception?
Ans: We can catch an exception in either of two ways. They are:

- By try-catch block
- By using throws clause

14. What is throwing an exception in Java?
Ans: When an exception occurs inside a method in java program, the method in which exception has occurred creates an exception object (i.e, an object of exception class) internally with the help of JVM and hands it over to the java runtime system (JVM). This process is called throwing an exception in java.

15. What is catching an exception in Java?
Ans: The process of finding a handler by JVM to handle thrown exception is called catching an exception.

16. What will happen to exception object after exception handling is done?
Ans: Once exception handling is done, the exception object will be garbage collected.

17. What is the difference between checked and unchecked exceptions in Java?
Ans: Refer to answer to question 11.

18. How will you handle the checked exception?
Or What are the different ways to handle checked exceptions?
Ans: A checked exception can be handled either by using try and catch block or by using throws clause in the method declaration. If not handles properly, it will give a compile-time error.

19. Which exception class can you use in the catch block to handle both checked and unchecked exceptions?
Ans: Exception class

20. Can we throw checked exceptions from the static block?
Ans: We cannot throw because there is no specific place to catch it and it is called only once.

21. Do checked exceptions occur at compile time?
Ans: Checked exceptions do not occur at compile-time. All exceptions always occur at runtime only but some exceptions are detected at compile-time and some other at runtime.

22. Are compile-time errors exceptions?
Ans: Compile-time errors are not exceptions. They come under errors. In Java, only runtime errors come under exceptions.

23. Does Java compiler check Runtime exceptions at compilation?
Ans: Java compiler does not check runtime exception at compile time whether the programmer handles them or not.

24. What happens when runtime exception occurs in a program?
Ans: When a runtime exception occurs in a method and the programmer does not handle it, JVM terminates program without the execution of the rest of the code.

25. What are the keywords to handle an exception in Java?

Ans: Java provides five essential keywords to handle an exception. They are:

- try
- catch
- finally
- throw
- throws

These keywords can be used to handle exceptions properly.

26. What happens when an exception is thrown by the main method?
Ans: When an exception is thrown by the main() method, JVM terminates the program and prints the exception message and stack trace in system console.

27. What is try block in Java?
Ans: A try is a block of code or statements that might throw an exception. That's why a try block is also known as exception generated block. An exception generated code (risky code) must be placed within a try block.

28. What is catch block in Java?
Ans: A catch is a block of code that handles the exception thrown by the try block. That's why it is also known as exception handler block.

29. Do we have to always put a catch block after a try block?

Ans: No, we do not always need to put a catch block after a try block.

30. What are the three possible forms of try block?

Ans: There are three possible forms of try block that are:

- try-catch
- try-finally
- try-catch-finally

31. Can we write statements between try block and catch block?

Ans: A catch block must be followed by try block. We cannot write a statement between the end of try block and the beginning of catch block.

32. What is a nested try block in Java?

Ans: A try block within another try block is called nested try block.

33. Assume that statement 2 arises an exception in the following statements. The exception object created is matched with argument of the catch block.

```
try

{

  statement 1;

  statement 2;

  statement 3;

}

catch(exception_class var)

{

  statement 4;

}

statement 5;
```

Which of the following statements is true?

a. statement 1 will be executed normally.

b. statement 4 inside catch block will be executed.

c. After executing statement 4, statement 3 in try block will be executed.
d. After executing statement 3, statement 5 will be executed.

Ans: a, b.

34. In question number 31, suppose the exception object created is not matched with argument of the catch block.
Which of the following statements is true?
a. statement 1 will be executed normally.
b. Program will be terminated abnormally after executing statement 5.
c. Program will be terminated abnormally and the rest of code will not execute.
d. Program will be terminated normally.

Ans: a, c.

35. In question number 31, suppose the exception arises in statement 4.
Which of the following statements is true?
a. statements 1, 2, 3 will be executed normally.
b. statement 5 will be executed.
c. Program will be terminated abnormally after execution of statement 5.
d. Program will be terminated normally.

Ans: a, b, d.

36. Which of the following statements are true?
a. Java try-catch block must be within a method.
b. A try block can also be used without a catch or finally block.
c. A finally block cannot come before catch block.
d. A catch block cannot be empty.
Ans: a, c.

37. Which of the following are checked exceptions?
a. ClassNotFoundException, InstantiationException, IllegalAccessException
b. ClassNotFoundException, InstantiationException, ClassCastException
c. ArrayIndexOutOfBoundsException, NegativeArraySizeException, ArrayStoreException
d. NegativeArraySizeException, SQLException
Ans: a.

38. Which of the following is RuntimeException?
a. IOException

b. IllegalAccessException
c. ArrayStoreException
d. None of these.
Ans: c.

39. Which exception will be thrown by the following code?

```
public class Test

{

public static void main(String[] args)

{

 try {

   int[ ] list = new int[10];

   System.out.println("list[10] is " + list[10]);

 }

 catch (ArithmeticException ex) {

    System.out.println("ArithmeticException");

 }

 catch (ArrayIndexOutOfBoundsException ai) {

    System.out.println("ArrayIndexOutOfBoundsException");

  }

 catch (RuntimeException ex) {

    System.out.println("RuntimeException");

   }

  }
```

```
}
```

Ans: ArrayIndexOutOfBoundsException

40. What is unreachable catch block error in Java?

Ans: Refer to this tutorial: Multiple catch block in Java

41. Assume that statement 2 arises an exception in the following try-catch block:

```
try {
  statement 1;
  statement 2;
  statement 3;
}
catch (Exception1 e1) {
}
catch (Exception2 e2) {
}
statement 4;
```

Answer the following questions:
a) Will statement3 be executed?
b) If the exception is not handled in catch block, will statement 4 be executed?
c) If the exception is handled in the catch block, will statement 4 be executed?

Ans: a) No

b) No, program will be terminated abnormally if the exception is not caught in catch block.

c) Yes.

42. What is a finally block in Java?
Ans: A finally block is a block of code that is always executed whether an exception occurs within a try block or not.

43. What is the use of finally block in Java?
Ans: finally block or clause is used for freeing up resources, cleaning up code, db closing connection, io stream, for terminating threads, etc.

44: Can we create a finally block without creating a catch block?
Ans: Yes

45. In what scenarios or conditions, a finally block will not be executed?
Ans: a) When System.exit() method is invoked before executing finally block.
b) When an exception happens in the finally block.

46. What is the importance of finally block in exception handling?
Ans: Refer to answer of question 43.

47. What is the difference between finally block and finalize() method?
Ans: Finally block will be always executed whether an exception is thrown or not. So, it is used to free up resources.

finalize() is a protected method in the Object class which is called by the JVM just before an object is a garbage collected.

48. Does a finally block override the value returned by try or catch block?
Ans: Yes

49. What is throw in Java?
Ans: Throw in Java is a keyword that is used to throw a built-in exception or a custom exception explicitly.

50. How do we throw an exception in Java?
Ans: We can throw an exception (either checked or unchecked exceptions) using throw statement in a try block. When an exception occurs in the try block, throw keyword transfers the control of execution to the caller by throwing an object of exception.

51. Can we throw multiple exceptions in one throw statement?

Ans: No, we cannot throw multiple exceptions in one throw statement. Only one object of exception type can be thrown by using throw statement at a time.

52. Can we use throw statement inside static block?

Ans: Yes, we can use throw statement inside the static block provided that exception handling is present.

53. What is rethrowing an exception in Java?

Ans: When an exception occurs in a try block, it is caught by a catch block inside the same method. The same exception object out from catch block can be rethrown explicitly using throw keyword. This mechanism is called rethrowing of exception in Java.

54. Explain the throws clause in Java.

Ans: Throws clause in Java is used with a method declaration. It provides information to the caller method about exceptions being thrown and the caller method has to take the responsibility of handling the exception.

55. What is the difference between throw and throws in Java?

Ans: There are several differences between throw and throws keywords. A list of important differences between them are given below:

Throw Clause	Throws Clause
1. throw clause is used when we want to throw an exception explicitly and want to handle it using catch block.	1. throws clause is used when we do not want to handle the exception and throw it out of the method.
2. Throw keyword is used inside the method body.	2. Throws keyword is used in method declaration (signature).
3. Throw is always followed by an object of Exception. e.g. throw new FileNotFoundException	3. Throws is always followed by name of exception class. e.g. throws FileNotFoundException
4. Throw is used to throw only one exception at a time.	4. Throws is used to throw multiple exceptions at a time. e.g. public void method()throws IOException, SQLException

56. Can we re-throw an Exception in Java?

Or Can a catch block throw an exception caught by itself?

Ans: Yes, we can re-throw an exception from catch block to another class where it can be caught.

57. What is custom or user-defined exception in Java?
Ans: Custom exceptions in Java are those exceptions that are created by a programmer to meet their specific requirements of the application. With the help of user-defined exception, we can create our own exception and message.

58. How to create your own user-defined exception in Java?
Ans: Refer to this tutorial: User-defined Exception in Java

59. What is a chained exception in java?
Ans: Throwing an exception along with another exception is called chained exception in java.

60. What is Throwable in Java?
Ans: Throwable in Java is a class that is the superclass of all exceptions and errors which may occur in Java program. It extends object class.

61. What are the methods provided by Throwable class in Java?
Ans: An important list of methods provided by Throwable class is as follows:

- getMessage()
- toString()
- printStackTrace()
- fillInStackTrace()
- getStackTrace()
- getClause()

62. What is error in Java? What are the types of errors in Java programming?
Ans: Error in Java occurs when a programmer violates the rules of Java programming language. It might be due to programmer's typing mistakes while developing a program.

There are three types of errors in Java. They are:

a. Compile-time errors (Syntax errors)
b. Runtime errors
c. Logical errors

63. What is the difference between error and exception in Java?
Ans: The difference between error and exception is as follows:

Exception	Error
1. Exception represents the problem that can be solved.	1. Error represents the problem that cannot be solved.
2. Exceptions are related to the application.	2. Errors are related to the environment in which the application is running.
3. Exceptions can be handled and can continue the execution of program.	3. Errors cannot be handled.
4. Exceptions are further classified into checked and unchecked exceptions.	4. Errors are not further classified into such classification.

64. State the difference between runtime error and syntax error (compile-time error).

Ans: The main difference between syntax error and runtime errors are as follows:

Syntax Error	Runtime Error
1. A syntax error occurs when we violate any grammatical rule in a programming language.	1. A runtime error occurs when a program terminates abnormally.
2. Syntax error can be found at compile time.	2. Runtime error can be found only when we execute the program.
3. Syntax errors are those errors that are caused by incorrect usage of programming language.	3. Runtime errors are those errors that are caused by incorrect usage of programming logic.

65. What is the difference between catch block and finally block?

Ans: The difference between catch block and finally block is as follows:

Catch block	Finally block
a. Catch block is used to handle an exception thrown by try block.	a. Finally block is used to execute important code such as closing connection, io stream, etc.

b. Catch block is executed only when an exception occurs in the try block otherwise it is skipped.	b. Finally block is always executed whether an exception occurs or not.
c. We can use multiple catch blocks for a single try block.	c. Only one finally block can be used for a single try block.

66. Can a catch or finally block throw an exception?

Ans: Yes, catch or finally block can throw checked or unchecked exception but it should be handled accordingly.

67. Can we throw an exception explicitly or manually?

Ans: Yes, using throw clause.

Hope that this tutorial has covered almost all the important exception handling interview questions in Java with the best possible answers. I hope that this collection of exception handling interview questions will help you to understand what kind of interview questions can be asked in the interview.
