1. What are the four access modifiers available in Java and what is their significance in terms of class, method, and variable accessibility?

In Java, access specifiers or access modifiers are used to control the visibility and accessibility of classes, methods, and variables. There are four access modifiers in Java, namely: public, protected, private, and default

1. What is the difference between exception and Error?

Exception:

The term [**exception**](https://www.javatpoint.com/exception-handling-in-java) is shorthand for the phrase **exception event**. It is an event that occurs during the execution of the program and interrupts the normal flow of program instructions. These are the errors that occur at compile time and run time. It occurs in the code written by the developers. It can be recovered by using the try-catch block and throws keyword. There are two types of exceptions i.e. **checked** and **unchecked**

Error:

[Errors](https://www.javatpoint.com/java-error) are problems that mainly occur due to the lack of system resources. It cannot be caught or handled. It indicates a serious problem. It occurs at run time. These are always unchecked. An example of errors is **OutOfMemoryError, LinkageError, AssertionError**, etc. are the subclasses of the Error class.

|  |  |  |
| --- | --- | --- |
| Basis of Comparison | Exception | Error |
| Recoverable/ Irrecoverable | Exception can be recovered by using the try-catch block. An error cannot be recovered. |  |
| Type | It can be classified into two categories i.e. checked and unchecked. | All errors in Java are unchecked. |
| Occurrence | It occurs at compile time or run time. | It occurs at run time. |
| Package | It belongs to java.lang.Exception package. | It belongs to java.lang.Error package. |
| Known or unknown | Only checked exceptions are known to the compiler. | Errors will not be known to the compiler. |
| Causes | It is mainly caused by the application itself. | It is mostly caused by the environment in which the application is running. |
| Example | Checked Exceptions: SQLException, IOException Unchecked Exceptions: ArrayIndexOutOfBoundException, NullPointerException, ArithmaticException | Java.lang.StackOverFlow, java.lang.OutOfMemoryError |

1. What is the difference between checked exception and unchecked Exception?

) **Checked Exception**

The classes that directly inherit the Throwable class except RuntimeException and Error are known as checked exceptions. For example, IOException, SQLException, etc. Checked exceptions are checked at compile-time.

) **Unchecked Exception**

The classes that inherit the RuntimeException are known as unchecked exceptions. For example, ArithmeticException, NullPointerException, ArrayIndexOutOfBoundsException, etc. Unchecked exceptions are not checked at compile-time, but they are checked at runtime.

1. Write a Java Program that reads user input for two integers and performs division. Handle the exception that is thrown when the second number is zero, and display an error message to the user.
2. Write the code of ArrayindexoutofboundsException & StringIndexoutofBoundsException?

When an array exceeds to it's size, the ArrayIndexOutOfBoundsException occurs. there may be other reasons to occur ArrayIndexOutOfBoundsException. Consider the following statements.

Int a[] = new int[5];

A[10] = 50; // ArrayIndexoutOfBoundsException

**StringIndexoutofBoundsException:**

Attempting to access a character at an index that is either negative or outside the range of the String's length causes this exception to be thrown

public class StringIndexOutOfBoundsExceptionExample {

public static void main(String args[]) {

String s = "example";

String substring = s.substring(2, 8);

}

}

1. You are building a login system for a website using Java. If the user enters an incorrect password, you want to display a message informing them of the error. How would you use exception handling to handle this situation?
2. Create a custom exception in Java called “InvalidAgeException” that is thrown when the user enters an age less than 18. Implement exception handling in a Java program to catch the “InvalidAgeException” and display an error message.
3. Implement exception handling in a Java program that reads data from a file. If the file does not exist, throw a “FileNotfoundException” and display an error message to the user.