

[By OnlineInterviewQuestions.com](http://OnlineInterviewQuestions.com)

[Cyient Java Interview Questions](#)

Q1. [List types of storage classes in java?](#)

There are basically four types of storage classes in Java:

- **Automatic storage class:** When a variable that is used in the coding is defined within a function and that also with the auto specifier then it simply belongs to this storage class.
- **Register storage class:** Those variables in the coding which are declared by the register specifier then it belongs to this storage class.
- **Static storage class:** It has a function to declare the variable with the help of the static specifier and that is how it belongs to the static storage class.
- **External storage class:** The main objective with this is that the variable which is being declared consists of external linkage.

Q2. [Write a java program to generate Fibonacci series?](#)

A JAVA program to generate Fibonacci series is as follows:

```
import java.util.Scanner;
public class Fibonacci
{
    public static void main(String[] args)
    {
        int n, a = 0, b = 0, c = 1;
        Scanner s = new Scanner(System.in);
        System.out.print("Enter the value of n: ");
        n = s.nextInt();
        System.out.print("Fibonacci Series: ");
        for(int i = 1; i <= n; i++)
        {
            a = b;
            b = c;
            c = a + b;
            System.out.print(a+" ");
        }
    }
}
```

Compile:

```
$ javac Fibonacci.java
```

Run:

```
$ java Fibonacci
```

Enter the value of n: 5

Fibonacci Series: 0 1 1 2 3

Q3. How does the garbage collector work in Java?

Garbage collection is the process of automatically managing memory in java. It is helpful to find the unused objects that are no longer used by the program and delete or remove them to free up the memory. The internal structure of the garbage collection mechanism uses several GC algorithms. It works with the most popular algorithm such as Mark and Sweep. Garbage collection is controlled by a thread known as Garbage Collector.

In Java, `System.gc()` and `Runtime.gc()` are two methods that send requests to the JVM for garbage collection. It totally depends on the JVM. If the Heap Memory is full, the JVM will not allow creating a new object and shows an error `java.lang.OutOfMemoryError`. When the garbage collector removes the object from the memory, first, the garbage collector thread calls the `finalize()` method of that object and then remove it.

Q4. What is difference between throw and throws ?

Difference between throw and throws are:

Throw

It is basically used inside the function. It is used when it needs to insert the exception.

It is used to basically use the exception. It can throw only a single exception at ones.

The exception which is been thrown cannot be checked here.

Throws

It is basically present in the function signature. Here if the functions have the same statements then it can result in certain exceptions.

Here multiple exceptions can be thrown and then whichever matches are thrown automatically.

Here for the propagation purpose, the throws can use the throw keyword. Here it can use the specific exception class.

Q5. Write a program to check String is Palindrome without using loop?

C program to Check String is Palindrome without using a loop is as follows:

```
/*
 * C program to find the length of a string without using the
 * built-in function also check whether it is a palindrome
 */
#include <stdio.h>
#include <conio.h>

void main()
{
    char string[25], reverse_string[25] = {'\0'};
    int i, length = 0, flag = 0;

    printf("Enter a string: ");
    gets(string);
    /* keep going through each character of the string till its end */
    for (i = 0; string[i] != '\0'; i++)
```

```

{
    length++;
}
printf("The length of the string '%s' = %d\n", string, length);
for (i = length - 1; i >= 0 ; i--)
{
    reverse_string[length - i - 1] = string[i];
}
/* Check if the string is a Palindrome */

for (flag = 1, i = 0; i < length ; i++)
{
    if (reverse_string[i] != string[i])
        flag = 0;
}
if (flag == 1)
    printf ("%s is a palindrome \n", string);
else
    printf ("%s is not a palindrome \n", string);
}

```

Output:

Enter a string:

how are youThe length of the string 'how are you' =
12

how are you is not a palindrome

Enter a string:

madamThe length of the string 'madam' =
5

madam is a palindrome

Enter a string:

mamThe length of the string 'mam' =
3

mam is a palindrome

Q6. What is the use of static keyword in Java?

The **static** keyword indicates that the particular member belongs to a type itself, rather than to an instance of that type in the Java programming language.

Q7. What is difference between JAVA and C?

The differences between JAVA and C are as follows:

S. N.	C	Java
1	C is a Procedural Programming Language.	Java is an Object-Oriented language.
2	C was developed by Dennis M. Ritchie in 1972.	Java language was developed by James Gosling in 1995.

S. N.	C	Java
3	It is a middle-level language as it is binding the gaps between machine level and high-level languages.	It is a high-level language because the translation of code is taken place into machine language, which uses a compiler or interpreter.
4	The C declaration variable is declared at the beginning of the block.	In Java, you can declare a variable anywhere.
5	Free is a variable used for freeing the memory in C.	A compiler will free up the memory by calling the garbage collector.
6	C generally breaks down to functions.	Java breaks down to Objects.
7	C support pointers.	Java does not support pointers.
8	Memory allocation can be done by malloc.	Memory allocation can be done by a new keyword.
9	The garbage collector needs to manage manually.	In Java, it is automatically managed by a garbage collector
10	C does not have a feature of overloading functionality.	Java supports method overloading.
11	C support for call by value and call by reference.	Java only supports a call by value.
12	C is a platform-dependent language.	Java is a platform-independent language.
13	C does not support Multithreading.	Java supports the concept of Multithreading.
14	It is not portable.	It is portable.
15	Exception handling cannot be directly achieved in C.	Exception Handling is supported in Java.

Q8. Why is string buffer better than string ?

StringBuffer is better than string because StringBuffer is faster than String when performing simple concatenations.

Q9. Explain exception chaining in Java?

Exception chaining is also known as **exception wrapping**. It is an object-oriented programming technique for handling exceptions by re-throwing a caught exception after wrapping it inside a new exception. The original exception is saved as property such as the cause of the new exception. In an application, Chained Exception helps to identify a situation in which one exception causes another Exception. For instance, Consider the I/O error: A method that throws an ArithmeticException because of an attempt to divide by zero but that caused the divisor to be zero.

Q10. What is OutOfMemoryError in Java?

OutOfMemoryError is an **exception** that is a runtime error thrown in the JAVA programming application when there is insufficient space to allocate an object in the Java heap.

Q11. What is the differences between final and finally in Java?

The differences between the final and finally are as follows:

S. N.	final	finally
1	A final method can't be overridden.	It ensures that the code inside that block is executed no matter what happens in try and catch blocks.
2	The final variable value can't be changed.	Only " <i>System.exit()</i> " can stop finally from being executed.
3	The final class can't be inherited,	finally is an optional block that is used together with try-catch blocks.
4	The final keyword is used to apply restrictions on class, method, and variable.	Finally is used to place important code, it will be executed whether an exception is handled or not.

Q12. What is Exception in Java?

Exception in Java is basically an event that takes place all of a sudden. This is really unexpected and also unwanted. Mostly it happens when the program is ready to execute and during the runtime, it can break the normal and easy flow of the program. These exceptions can be also called errors. At a time when the error occurs the method which is present there sets an object which is then sent to the runtime process.

Q13. How to write custom exception in Java?

A custom exception is also known as a user-defined exception are derived classes of Java Exception classes. In order to create your Custom exception in Java following point must be taken care of.

- All exceptions must be a child of Throwable.
- If you want to write a checked exception that is automatically enforced by the Handle or Declare Rule, you need to extend the Exception class.
- If you want to write a runtime exception, you need to extend the RuntimeException class.

Custom exception example in Java

```
class InvalidAgeException extends Exception{  
    InvalidAgeException(String s){  
        super(s);  
    }  
}
```

```

    }
}
class CustomExceptionTest{

    static void validate(int age)throws InvalidAgeException{
        if(age<18)
            throw new InvalidAgeException("not valid");
        else
            System.out.println("welcome to vote");
    }

    public static void main(String args[]){
        try{
            validate(13);
        }catch(Exception m){System.out.println("Exception occurred: "+m);}

        System.out.println("rest of the code...");
    }
}

```

Above is an example of a custom exception in Java that checks the age of voter. If age is less than 18 years than it throws an InvalidAgeException.

Q14. [How can you catch multiple exceptions in java?](#)

With the help of **Java catch multiple exceptions** you can catch multiple exceptions. In this method a try block can be followed by one or more catch blocks such that each catch block must contain a different exception handler. Now you can use java multi-catch block.

Q15. [What is the difference between a checked and an unchecked exception?](#)

Whenever an exception is generated in Java it is either type of a checked or unchecked exception. Below are the few differences between checked and unchecked exceptions in Java.

Checked Exception

Checked exceptions are checked at compile-time.

Interrupted Exception, file not found, Class Not Found are few Checked exceptions.

Unchecked Exception

Unchecked exceptions are not checked at compile time

Empty Stack Exception, Arithmetic Exception, Null Pointer Exception, Array Index Out of Bounds Exception are few Unchecked exceptions

Difference Between Checked And Unchecked Exceptions

Exceptions that are checked and handled at compile time are checked exceptions.	Exceptions that are not checked and handled at compile time are unchecked exceptions.
They are direct subclasses of Exception but do not inherit from RuntimeException.	They are a direct subclass of RuntimeException class.
The program gives a compilation error if a method throws a checked exception and the compiler is not able to handle the exception on its own.	The program compiles fine because the exceptions escape the notice of compiler. Exceptions occur due to errors in programming logic.
A Checked Exception occurs when the chances of failure are too high.	Unchecked Exception occurs mostly due to programming mistakes.
Common checked exceptions include IOException, DataAccessException, IllegalAccessException, InterruptedException, etc.	Common unchecked exceptions include ArithmeticException, InvalidClassException, NullPointerException, etc.

Q16. What are meta-annotations?

Meta annotations are annotations that are used to annotate other annotation types. In Java `java.lang.annotation` package contains four annotation types they are `@Documented`, `@Inherited`, `@Repeatable`, `@Target`.

Please Visit [OnlineInterviewQuestions.com](https://www.onlineinterviewquestions.com) to download more pdfs