

JAVA 8

1. What are the new features introduced in Java 8?

- Functional Interface
- Lambda Expression
- Method Reference
- Default methods in Interface
- Optional
- Date API
- Stream API

2. What is functional interface?

Functional interface is an interface with only one abstract method. It can have static method, default methods or can override Object's class methods.

You can create function interface by having only one abstract method in it. You can have @FunctionalInterface annotation optionally.

3. Built-In Functional Interfaces

1. Predicate: The Predicate interface has an abstract method test which gives a Boolean value as a result for the specified argument.

```
public interface Predicate  
{  
    public boolean test(T t);  
}
```

2. Function: The Function interface has an abstract method apply which takes argument of type T and returns a result of type R.

```
public interface Function
{
    public R apply(T t);
}
```

3. Consumer: The Consumer interface has an abstract method accept which takes argument of type T and performs operation on it. This method does not return any value.

```
public interface Consumer
{
    public void accept(T t);
}
```

4. Supplier: The Supplier interface has an abstract method get which does not take any argument as input but produces a value of Type T.

```
public interface Supplier
{
    public T get()
    ;
}
```

4. What is Lambda expression?

Lambda expression is anonymous function which have set of parameters and a lambda (->) and a function body . You can call it function without name.

Lambda expression provides implementation of functional interface.

Parameters:

- ()-> without input parameter
- a->single input parameter
- (int a)->single input parameter with type
- (int a, int b)-> multiple parameters with type of each*
- (a, b)-> multiple parameters with type of no one

Array Token: ->

Body:

If only single statement no curly braces required, return type same as body expression.

5. Method Reference in Java?

Method Reference is introduced in Java 8. It is used to refer method of functional interface

- Reference to Static Method

ClassName::staticMethodName

- Reference to Instance Method

containingObject::instanceMethodName

- Reference to Constructor

ClassName::new

6. Default Method in Interface?

A default method is a method declared with keyword default with an implementation, which can be found in an interface.

We can use a default method to add a new functionality to an interface, while maintaining backward compatibility with classes that are already implementing the interface.

7. What is Optional Class ?

Java 8 has introduced new class Called Optional. This class is basically introduced to avoid NullPointerException in java.

Optional class encapsulates optional value which is either present or not.

It is a wrapper around object and can be used to avoid NullPointerExceptions.

8. DATE API?

Old API Drawbacks – Thread Safety, Day Start at 0 which led to popularity of third-party API.

To address the drawbacks and provide better support Java 8 provides new Date & Time API.

- LocalDate
- LocalTime
- LocalDateTime

9. Stream API

A *Stream* in Java can be defined as a sequence of elements from a source. Source can be Collection, Array etc.

- Stream.of()
- Stream.of(array)
- list.stream()
- Stream.empty()

10. Intermediate Operations

operations that produce another stream as a result.

- Filter() -> that match given predicate
- Limit(n) -> limit upto only n records
- Skip(n) -> skip first n records
- Distinct()-> return distinct records
- Sorted()-> sort
- Map()-> apply given function and return stream

- peek() allows execution of system logic as part of a stream flow.
- flatMap - Transform each element to a stream of its constituent elements and flatten all the streams into a single stream

11. Terminal operations

terminal operations return non-stream values like primitive or object or collection or may not return anything.

- forEach -> perform operation on each element
- toArray -> converting stream element to array
- reduce ->
- collect -> store the result of the stream operation into a collection
- min
- max
- count
- anyMatch
- allMatch
- noneMatch
- findFirst
- findAny

12. What is MetaSpace in Java 8?

In Java 8, a new feature was introduced to store classes. The area where all the classes that are stored in Java 8 are called MetaSpace. MetaSpace has replaced the PermGen.

Till Java 7, PermGen was used by Java Virtual Machine to store the classes. Since MetaSpace is dynamic as it can grow dynamically and it does not have any size limitation, Java 8 replaced PermGen with MetaSpace.

13. What is the difference between a normal and functional interface in Java?

The normal interface in Java can contain any number of abstract methods, while the functional interface can only contain just one abstract method.

Well, they are called functional interfaces because they wrap a function as an interface. The function is represented by the single abstract method on the interface.

14. What is a Predicate interface? (answer)

A Predicate is a functional interface that represents a function, which takes an Object and returns a boolean. It is used in several Stream methods like filter(), which uses Predicate to filter unwanted elements.

here is how a Predicate function looks like:

```
public boolean test(T object){  
    return boolean;  
}
```

You can see it just has one test() method which takes an object and returns a boolean. The method is used to test a condition if it passes; it returns true otherwise false.

15. What is Parallel Stream in Java Stream API?

You can execute streams in serial or in parallel. When a stream executes in parallel, the Java runtime partitions the stream into multiple sub-streams

16. What are Reduction Operations in Java Stream API?

Stream API contains many terminal operations (such as average, sum, min, max, and count) that return one value by combining the contents of a stream. These operations are called reduction operations because these operations reduce the stream to a single non-stream value.

<https://www.java67.com/2018/10/java-8-stream-and-functional-programming-interview-questions-answers.html>

<https://www.netjstech.com/2017/09/java-stream-api-interview-questions.html>