□ Core Java 8 Features

1. What are the main features introduced in Java 8?
Java 8 introduced Lambda expressions, Functional Interfaces, Stream API, Default and Static methods in interfaces, Optional, new Date/Time API, and Nashorn JavaScript engine.

2. What is a lambda expression in Java 8?
A lambda is an anonymous function that can be passed as a parameter.
Example:

java

CopyEdit

(int a, int b) -> a + b

3. What is a functional interface?
An interface with exactly **one abstract method**.
Example: Runnable, Callable, Comparator.

4. How can you define a custom functional interface?

java

CopyEdit

@FunctionalInterface

interface MyFunc {

    void execute();

}

5. What is the purpose of the @FunctionalInterface annotation?
It tells the compiler the interface is intended to be a functional interface. It helps catch errors if multiple abstract methods are added.

🔁 Lambda Expressions

6. Can you pass a lambda to a method?
Yes.

java

CopyEdit

public void process(MyFunc f) { f.execute(); }

process(() -> System.out.println("Hello"));

7. What are the restrictions of lambda expressions?

- Cannot access non-final local variables unless they're "effectively final".
- Cannot throw checked exceptions without handling.

8. How does a lambda expression differ from an anonymous class?

- Lambdas are more concise.
- Do not have a scope for this.
- Cannot define state or constructors.

🔁 Streams API

9. What is the Stream API?
A feature to process collections in a **functional style** using operations like map, filter, reduce.

10. Difference between Collection and Stream?

- Collection is in-memory, can be modified.
- Stream is a **pipeline**, cannot be reused, and processes data lazily.

11. What is the difference between intermediate and terminal operations in Stream?

- **Intermediate**: return a Stream (filter, map)
- **Terminal**: return a result (collect, forEach, count)

12. What does .map() do in Stream?
Transforms each element.

java

CopyEdit

list.stream().map(String::toUpperCase)

13. How is .filter() used?
Filters based on a condition.

java

CopyEdit

list.stream().filter(s -> s.startsWith("A"))

14. What is .reduce() in Stream API?
Combines elements into a single value.

java

CopyEdit

list.stream().reduce(0, Integer::sum)

15. What is the use of .collect()?
Terminal operation to convert Stream to Collection, String, or another type.

java

CopyEdit

list.stream().collect(Collectors.toList())

Date and Time API (java.time)

16. What is LocalDate, LocalTime, LocalDateTime?
Immutable classes to represent date, time, and both without timezone.

17. How is Java 8 date/time better than old Date/Calendar?

- Immutable
- Thread-safe
- Clear API design

18. How to get current date/time?

java

CopyEdit

LocalDate.now();

LocalTime.now();

LocalDateTime.now();

19. How to parse and format a date in Java 8?

java

CopyEdit

LocalDate.parse("2024-01-01");

DateTimeFormatter.ofPattern("dd-MM-yyyy");

Optional

20. What is Optional in Java 8?
A container object which may or may not contain a value. Prevents NullPointerException.

21. How to create an Optional?

java

CopyEdit

Optional.of(value);

Optional.ofNullable(value);

Optional.empty();

22. How to retrieve value from Optional?

java

CopyEdit

optional.get(); // risky

optional.orElse("default");

optional.ifPresent(val -> System.out.println(val));

23. Difference between orElse and orElseGet?

- orElse: always evaluates the default even if Optional is present.
- orElseGet: only evaluates when needed (lazy).

🧑‍💻 Interface Enhancements

24. What are default methods in interfaces?
Methods with a default implementation in interfaces.

java

CopyEdit

default void log() {

  System.out.println("Log");

}

25. What are static methods in interfaces?
Static utility methods.

java

CopyEdit

static void print() {

  System.out.println("Print");

}

26. Can an interface have a constructor?
No, interfaces can't have constructors.

🔧 Miscellaneous

27. What is method reference in Java 8?
Shorthand for a lambda that calls an existing method.

java

CopyEdit

list.forEach(System.out::println);

28. What is the use of Predicate, Function, Consumer interfaces?

- **Predicate**: takes T, returns boolean
- **Function**: takes T, returns R

- **Consumer**: takes T, returns nothing

29. What is the difference between findFirst() and findAny() in Stream?

- findFirst(): always returns the first element
- findAny(): may return any element (useful in parallel streams)

30. Is Java 8 backward compatible?
Yes. Java 8 code can use older Java features, and most older code runs on Java 8 JVM.