

1.What is the lambda expression in Java?

Ans: Lambda Expressions were added in Java 8. A lambda expression is a short block of code which takes in parameters and returns a value. Lambda expressions are similar to methods, but they do not need a name and they can be implemented right in the body of a method.

```
car ref = () -> {  
    System.out.println("Driving....");  
};  
ref.drive();
```

It is also called an anonymous inner class because we write the method without name.

2. Can you pass lambda expression to a method and when?

Ans: A lambda expression passed in a method that has an argument of type of functional interface. If we need to pass a lambda expression as an argument, the type of parameter receiving the lambda expression argument must be of a functional interface type.

3. What is the functional interface in Java 8?

Ans: Functional interface is nothing but an interface with a single abstract method. They are also annotated with `@FunctionalInterface`.

4. What is benefit of lambda expressions in Java?

Ans: Lambda expressions improve code readability and do not require interpretation. Lambdas allow you to write concise code. It encourages the use of functional programming. It simplifies variable scope and encourages code reusability.

5. Is it mandatory for a lambda expression to have parameter?

Ans: No, it is not mandatory.