**Lambda Expressions**

* Lambda expression is a way to implement a functional interface.
* A functional interface is an interface that has only one abstract method. It can have multiple default or static methods, but only 1 abstract method. A black screen with white text

  AI-generated content may be incorrect. A computer screen shot of a program

  AI-generated content may be incorrect.A screen shot of a computer program

  AI-generated content may be incorrect.

Wrapper Class: Wrapper classes are a set of classes that provide a way to use primitive data types as objects.

A screen shot of a computer program

AI-generated content may be incorrect.

**Generics:**

Generics are powerful feature that allows to write type-safe and reusable code. They enable classes, interfaces and methods to operate on objects of various types while providing compile time checking.

A screenshot of a computer code

AI-generated content may be incorrect.

Bounded Generics: Restricts the type that can be used as type parameters.

A computer code with text

AI-generated content may be incorrect.

Wildcards: Wildcards allow flexibility when working with generic types, especially in method parameters.

A screenshot of a computer program

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.

Comparator: Comparator is a functional interface in Java which has only 1 abstract method: **int compare(T o1, T o2);**