What is lambda expression? why used?

* **have no** name this **function**
* no return types
* Increase readability, concise, reusability

What is functional interface?

* interface with one abstract method is called functional interface.
* annotated by @functional Interface
* Consumer, function, Bifunction, supplier, predicate
* The major benefit of java 8 functional interfaces is that we can use lambda expressions to instantiate them and avoid using bulky anonymous class implementation

What is foreach implementation?

default void forEach(Consumer<? super T> action) {

    Objects.requireNonNull(action);

    for (T t : this) {

        action.accept(t);

    }

}

What is Predicate, Binary and function interface in java 8?

* Predicate: The Predicate interface has an abstract method test which gives a Boolean value as a result for the specified argument. Its prototype is
* public Predicate
* {
* public Boolean test(T t);
* }
* Binary Operator: The Binary Operator interface has an abstract method apply which takes two argument and returns a result of same type. Its prototype is
* public interface Binary Operator
* {
* public T apply(T x, T y);
* }
* Function: The Function interface has an abstract method apply which takes argument of type T and returns a result of type R. Its prototype is
* public interface Function
* {
* public R apply(T t);
* }

What is Stream and different way to create stream?

* Stream is a fixed data structure, where elements are computed on demand.
* Method to create Stream: Stream.Of, collection. stream (); Stream.generte().

Stream to collection conversion?

* Stream.collect(collector. toList)
* Stream.toArray(EntryType: new)

Stream Operations

**Intermediate Operations**

Intermediate operations return another Stream which allows you to call multiple operations in a form of a query. Intermediate operations do not get executed until a [terminal operation](https://www.leveluplunch.com/java/examples/stream-terminal-operations-example/) is invoked as there is a possibility they could be processed together when a terminal operation is executed.

https://www.leveluplunch.com/java/examples/stream-intermediate-operations-example/

* [filter()](https://howtodoinjava.com/java8/java-stream-filter-example/)
* [map()](https://howtodoinjava.com/java8/stream-map-example/)
* [flatMap()](https://howtodoinjava.com/java8/stream-flatmap-example/)
* [distinct ()](https://howtodoinjava.com/java8/java-stream-distinct-examples/)
* [sorted()](https://howtodoinjava.com/java8/stream-sorted-method/)
* [peek()](https://howtodoinjava.com/java8/java-stream-peek-example/)
* [limit()](https://howtodoinjava.com/java8/java-stream-limit-method-example/)
* [skip()](https://howtodoinjava.com/java8/stream-skip-example/)

Terminal Operations

* [forEach()](https://howtodoinjava.com/java8/java-stream-foreach/)
* [forEachOrdered()](https://howtodoinjava.com/java8/java-stream-foreachordered/)
* [toArray()](https://howtodoinjava.com/java8/convert-stream-to-array/)
* reduce()
* collect()
* [min()](https://howtodoinjava.com/java8/java-stream-min/)
* [max()](https://howtodoinjava.com/java8/java-stream-max/)
* [count()](https://howtodoinjava.com/java8/stream-count-elements-example/)
* [anyMatch()](https://howtodoinjava.com/java8/stream-anymatch-example/)
* [allMatch()](https://howtodoinjava.com/java8/stream-allmatch-example/)
* [noneMatch()](https://howtodoinjava.com/java8/stream-nonematch-example/)
* [findFirst()](https://howtodoinjava.com/java8/stream-findfirst-findany/)
* [findAny()](https://howtodoinjava.com/java8/stream-findfirst-findany/)

What is box stream?

* Primitive to wrapper conversion used box stream.

How to access method reference in java8?

* Reference to **static method: CLASSNAME: METHOD ()**
* Instance method from instance: instance: method ();
* Reference to **constructor: Class:: new**

What is optional and method to create optional object?

* Optional used to avoid null pointer exception.
* Method: Optional.Empty();
* Optional.of(5);
* Optional.ofNullable(null);

What is directory Stream?

* To access files

What is watch Service ?

What is difference b/w Stream.map() vs Stream.flatMap() in Java 8 ?

* The function you pass to map () operation returns a single value.
* The function you pass to flat Map() operation returns a Stream of value.
* flat Map() is combination of map and flat operation.
* map() is used for transformation only, but flat Map() is used for both transformation and flattening.