**Java 8 features**

1. Lambda Expressions
2. Marker Interfaces
3. Functional Interfaces

Functional Programming

1. Default (Concrete) methods in interfaces
2. Static (Concrete) methods in interfaces
3. Predefined functional interfaces --- JAVA API
4. Java.util.function.Pridicate
5. Java.util.function.Function
6. Java.util.function.Consumer
7. Java.util.function.Supplier
8. Stream API
   1. used to work with collection
9. Method and constructor reference operator (::)
10. Date and Time API
    1. Joda API, simplified

**Lambda Expressions**

Usages

To simplify the programming in java

To use functional programming in java

To increase the performance of application… during parallel processing

To reduce the size of the code

Example 1

Public void great(){

System.out.println(“hello”);

}

()->{System.out.println(“hello”);}

Example 2

Public void add(int a, int b){

System.out.println(a+b)

}

(int a, int b) - >System.out.println(a+b);

Or

(a, b) - >System.out.println(a+b);

Example 3:

Public intgetLength(String s){

Return s.length();

}

(String s) - > {return s.length();};

OR

(S)- >s.length();

Marker Interfaces

No method at all

**Functional Interfaces**

To use lambda expressions we need FI. FI is an interface with SAM – Single Abstract Method

Example 1

Interface inter

{

Public String getLength( String s);

}

Class StringDemo implements (String s){

Return s.length();

}

Class Test {

Ps vm (Srtingargs[]){

StringDemosd = new StringDemo();

Intlen = sd.getLenght(“hello”);

System.out.println(len);

}

}

@FunctionalInterface

Interface inter{

Public String getLength( String s);

}

Class Test {

P s v m (String args []){

Inter I = s- >s.length();

Intlen= I.getLength(“hello”);

System.out.println(len);

}

}

One and only abstract meths

One or more default method

One or more static method

Java.util.function.Pridicate

Has a SAM, used to do some condition checking. It takes one argument and returns a

{

"header":{"requestId":"102","userId":"GUI","timeStamp":1520322142893, "corelationId":"1000001"},

"body":{"departmentId":"100"}

}





















