Java Lambda Expressions

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.

```
import java.util.ArrayList;

public class Main {
  public static void main(String[] args) {
    ArrayList<Integer> numbers = new ArrayList<Integer>();
    numbers.add(5);
    numbers.add(9);
    numbers.add(8);
    numbers.add(1);
    numbers.forEach( (n) -> { System.out.println(n); } );
  }
}
```

```
import java.util.ArrayList;
import java.util.function.Consumer;

public class Main {
  public static void main(String[] args) {
    ArrayList<Integer> numbers = new ArrayList<Integer>();
    numbers.add(5);
    numbers.add(9);
    numbers.add(8);
    numbers.add(1);
    Consumer<Integer> method = (n) -> { System.out.println(n); };
    numbers.forEach( method );
  }
}
```

```
// A Java program to demonstrate simple lambda expressions
import java.util.ArrayList;
class Test {
  public static void main(String args[])
  {
    // Creating an ArrayList with elements
    // {1, 2, 3, 4}
    ArrayList<Integer> arrL = new ArrayList<Integer>();
```

```
    arrL.add(1);
    arrL.add(2);
    arrL.add(3);
    arrL.add(4);

    // Using lambda expression to print all elements
    // of arrL
    arrL.forEach(n -> System.out.println(n));

    // Using lambda expression to print even elements
    // of arrL
    arrL.forEach(n -> {
      if (n % 2 == 0)
        System.out.println(n);
    });
  }
}
```