**Java-8 – How to write lambda expression part\_02**

* **Rule\_01:**

If the body of lambda expression contains only one statement, then the curly braces are optional.

Example:

public void m1(){

System.out.println(“Hello”);

}

() -> {System.out.println(“Hello”);}

As the body contains only one statement we can remove the curly braces.

() -> System.out.println(“Hello”);

* **Rule\_02:**

Example:

public void add(int a, int b){

System.out.println(a+b);

}

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

Note:

Based on the context sometimes the compiler can guess the data types automatically. This feature is called “Type Inference”

In this case we can remove the type as well.

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

* **Rule\_03:**

Example:

public int getLength(String s){

return s.length();

}

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

(s) -> return s.length();

Based on the context compiler can guess, it has to return something, then in that case, “return” is optional.

(s) -> s.length();

If lambda expression has only one parameter, then open and close parenthesis also optional.

s -> s.length();