**Java Method References**

Java provides a new feature called method reference in Java 8. Method reference is used to refer method of functional interface. It is compact and easy form of lambda expression. Each time when you are using lambda expression to just referring a method, you can replace your lambda expression with method reference.

## Types of Method References

There are following types of method references in java:

1. Reference to a static method.
2. Reference to an instance method.
3. Reference to a constructor.

## 1) Reference to a Static Method

You can refer to static method defined in the class. Following is the syntax

Syntax : ContainingClass::staticMethodName

Example 1:

public class Demo2

{

public static void ThreadStatus()

{

System.out.println("Thread is running...");

}

public static void main(String[] args) {

Thread t2=new Thread(Demo2::ThreadStatus);

t2.start();

} }

Example 2:

interface Sayable{

void say();

}

public class Demo {

public static void saySomething(){

System.out.println("Hello, this is static method.");

}

public static void main(String[] args) {

// Referring static method

Sayable sayable = Demo::saySomething;

// Calling interface method

sayable.say();

}

}

## 2) Reference to an Instance Method

like static methods, you can refer instance methods also. In the following example, we are describing the process of referring the instance method.

Syntax: containingObject::instanceMethodName

Example 1: public class InstanceMethodReference2 {

public void printnMsg(){

System.out.println("Hello, this is instance method");

}

public static void main(String[] args) {

Thread t2=new Thread(new InstanceMethodReference2()::printnMsg);

t2.start();

}

}

Example 2:

interface Sayable{

void say();

}

public class InstanceMethodReference {

public void saySomething(){

System.out.println("Hello, this is non-static method.");

}

public static void main(String[] args) {

// Creating object

InstanceMethodReference methodReference = new InstanceMethodReference();

// Referring non-static method using reference

Sayable sayable = methodReference::saySomething;

// Calling interface method

sayable.say();

// Referring non-static method using anonymous object

Sayable sayable2 = new InstanceMethodReference()::saySomething; // You can use anonymous object also

// Calling interface method

sayable2.say();

}

}

## 3) Reference to a Constructor

You can refer a constructor by using the new keyword. Here, we are referring constructor with the help of functional interface.

Syntax: ClassName::new

Example 1:

interface Messageable{

Message getMessage(String msg);

}

class Message{

Message(String msg){

System.out.print(msg);

}

}

public class ConstructorReference {

public static void main(String[] args) {

Messageable hello = Message::new;

hello.getMessage("Hello");

}

}