PROGRAMS BASED ON LAMBDA EXPRESSION

Practical No.:05

Program No.:01

Aim:-Write a Java Program using Lambda Expression to print “Hello World”.

Code:

```
interface Hello
{
	void sayHello();

}
public class HelloWorld {
	public static void main(String[] args)
	{
		Hello hello=()->System.out.println("Hello World");
		hello.sayHello();
	}

}
```

Output:-

```
<terminated> HelloWorld [Java A
Hello World
```

PROGRAMS BASED ON LAMBDA EXPRESSION

Program No.:02

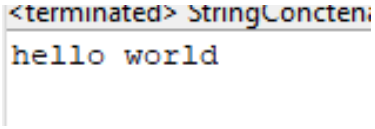
Aim:-Write a Java Program using Lambda Expression to concatenate two strings.

Code:

```
interface Concatenate
{
	String join(String s1,String s2);
}

public class StringConctenate
{
	public static void main(String[] args)
	{
		Concatenate concat=(s1,s2)->s1+s2;
		System.out.println(concat.join("hello"," world"));
	}
}
```

Output:-



Program No.:03

Roll no:39 **Arif Shaikh**

PROGRAMS BASED ON LAMBDA EXPRESSION

Aim:-Write a Java Program using Lambda Expression with Single Parameter.

Code:

```
interface SingleParameter
{
	void display(String message);
}
public class SingleParameterExample
{
	public static void main (String[] args)
	{
		SingleParameter displayMessage=message->System.out.println("Message:
"+message);
		displayMessage.display("Lambda with single parameter");
	}
}
```

Output:-

```
Message: Lambda with single parameter
```

Program No.:04

Roll no:39 **Arif Shaikh**

PROGRAMS BASED ON LAMBDA EXPRESSION

Aim:-Write a Java Program using Lambda Expression with Multiple Parameter to add two numbers.

Code:

```
interface Add
{
	int sum(int a, int b);
}
public class AddTwoNumbers {
	public static void main(String[] args)
	{
		Add addition=(a,b)->a+b;
		System.out.println("Sum: "+addition.sum(5, 10));
	}

}
```

Output:

```
<terminated> AddTwoNumbers [Java Applic
Sum: 15
```

Program No.:05

Roll no:39 **Arif Shaikh**

PROGRAMS BASED ON LAMBDA EXPRESSION

Aim:-Write a Java Program using Lambda Expression to calculate following:-

a. **Convert Fahrenheit to Celcius.**
b. **Convert Kilometers to Miles.**

a.Convert Fahrenheit to Celcius:

Code:

```
interface FahrenheitToCelcuis
{
	double convert(double fahrenheit);
}
public class TemperatureConversion
{
	public static void main (String[] args)
	{
		FahrenheitToCelcuis convert=f->(5.0/9)*(f-32);
		System.out.println("Celcuis: "+convert.convert(98.6));
	}
}
```

Output:-

```
<terminated> TemperatureConver
Celcuis: 37.0
```

b.Convert Kilometers to Miles:

Code:

```
interface KilometersToMiles{
	double convert(double km);
}
public class DistanceConversion
{
	public static void main(String[] args)
	{
		KilometersToMiles convert=km -> km*0.621371;
		System.out.println("Miles: "+convert.convert(10));
	}
}
```

Output:

```
<terminated> DistanceC
Miles: 6.21371
```

Program No.:06

Roll no:39 **Arif Shaikh**

PROGRAMS BASED ON LAMBDA EXPRESSION

Aim:-Write a Java Program using Lambda Expression with and without return keyword.

6.1:-With return keyword

Code:

```
interface Multiply{
	int product(int a, int b);
}
public class WithReturn {
	public static void main(String[] args)
	{
		Multiply multiply=(a,b)->
		{
			return a*b;
		};
		System.out.println("Product: "+multiply.product(5, 4));
	}
}
```

Output:

```
<terminated> WithR
Product: 20
```

6.1:-Without return keyword

Code:

PROGRAMS BASED ON LAMBDA EXPRESSION

```
interface Subtract
{
	int difference(int a,int b);
}
public class WithoutRetrun
{
	public static void main(String[] args)
	{
		Subtract subtract=(a,b)->a-b;
		System.out.println("Difference: "+subtract.difference(10, 4));
	}
}
```

Output:

```
<terminated> WithoutRet
Difference: 6
```