Name of the Course : Java 9 Core principles

Level : Easy

Tool Stack : List interface, Map and Collectors.

Problem Statement :

In Laptop store sales different company laptop and need to maintain list contain id, company name and prize of laptop. Show list of prices of all laptop. Insert different id, laptop company name and price to list using ArrayList.

Description : You are provided with a public class Product with private attributes :

**int** id;

String name;

**float** price;

Appropriate public getters and setters are already written.

Add contrutor in Product class :

**public** Product(**int** id, String name, **float** price)

Write a **public** **static** **boolean** addLaptopDetails() in LaptopCollectors class as, **static** **boolean** addLaptopDetails() this method will return true if laptop details are added through contrutor and also create ArrayList object to add productlist shown below.

List<Product>productsList = **new** ArrayList<Product>();

productsList.add(**new** Product(1,"HP Laptop",25000f)); productsList.add(**newP**roduct(2,"Dell Laptop",30000f));

In this method also print only all laptop prices by using Collectors and Map .

You are provided with a public class LaptopCollectors which has the main method.

Check the correctness of the methods written in these classes.

Note : All class, methods needs to be declared as public

**Code:**

**package** main.java.yaksha;

**public** **class** Product{

**private** **int** id;

**private** String name;

**private** **float** price;

**public** **int** getId() {

**return** id;

}

**public** **void** setId(**int** id) {

**this**.id = id;

}

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

**public** **float** getPrice() {

**return** price;

}

**public** **void** setPrice(**float** price) {

**this**.price = price;

}

**public** Product(**int** id, String name, **float** price) {

**this**.id = id;

**this**.name = name;

**this**.price = price;

}

}

package main.java.yaksha;

import main.java.yaksha.Product;

import java.util.stream.Collectors;

import java.util.List;

import java.util.ArrayList;

public class LaptopCollectors {

public static void main(String[] args) {

boolean status=addLaptopDetails();

}

public static boolean addLaptopDetails() {

List<Product> productsList = new ArrayList<Product>();

productsList.add(new Product(1,"HP Laptop",25000f));

productsList.add(new Product(2,"Dell Laptop",30000f));

productsList.add(new Product(3,"Lenevo Laptop",28000f));

productsList.add(new Product(4,"Sony Laptop",28000f));

productsList.add(new Product(5,"Apple Laptop",90000f));

List<Float> productPriceList =

productsList.stream().map(x->x.getPrice()).collect(Collectors.toList());

System.out.println(productPriceList);

return true;

}

}

Junit Testing

**package** test.java.yaksha;

**import** **static** org.junit.jupiter.api.Assertions.\*;

**import** java.io.IOException;

**import** org.junit.jupiter.api.Test;

**import** main.java.yaksha.LaptopCollectors;

**import** main.java.yaksha.TestUtils;

**class** LaptopCollectorsTest {

@Test

**void** testAddLaptopDetails() **throws** IOException {

*assertEquals*(**true**, LaptopCollectors.*addLaptopDetails*());

}

}

Test Data1

Output:

[25000.0, 30000.0, 28000.0, 28000.0, 90000.0]

Learning outcome: Participant could able to learn how to use List interface, Map and Collectors.