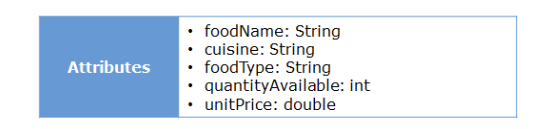
**EX.No 1**

**Create a new class Food in the Java project SwiftFood with the details given below.**

****

**ALGORITHM:**

STEP 1: Initialize the code

STEP 2: Enter the values

STEP 3: Run the code in jdk and output is displayed

STEP 4: End of execution

**PROGRAM:**

package Unit2;

class Food {

    String name;

    String cusine;

    String type;

    int quantity;

    double price;

}

public class FoodClassEx1 {

    public static void main(String[] args) {

        Food f  = new Food();

        f.name = "Idli";

        f.cusine = "South Indian";

        f.type = "Veg";

        f.quantity = 2;

        f.price = 15;

        System.out.println(f.name + " " + f.cusine + " " + f.type + " " + f.quantity + " " + f.price);

    }

}

**OUTPUT:**

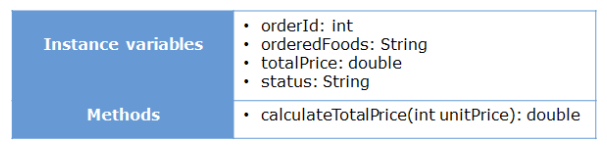


**Result:**

The program is Executed successfully

**EX.No 2A**

**Create a new class Order in the Java project SwiftFood with the instance variables and methods mentioned below.**

****

**Method Description**

**calculateTotalPrice(int unitPrice)**

* **Calculate the total price by applying a service charge of 5% on the food item ordered and store it in the instance variable totalPrice.**
* **Return the calculated total price.**

**Create an object of the Order class, initialize the instance variables, invoke the calculateTotalPrice() method and display the values of the instance variables in the main() method of the Tester class.**

**ALGORITHM:**

STEP 1: Initialize the code

STEP 2: Enter the values

STEP 3: Run the code in jdk and output is displayed

STEP 4: End of execution

**PROGRAM:**

package Unit2;

class Order {

    int id;

    String orderedFood;

    String status;

    double total;

    double calculateTotal(int price) {

        double tot = price + price \* 0.05;

        return tot;

    }

}

public class OrderClassEx2A {

    public static void main(String[] args) {

        Order o = new Order();

        o.id = 1;

        o.orderedFood = "Idli";

        o.status = "Ordered";

        o.total = o.calculateTotal(15);

        System.out.println(o.id + " " + o.orderedFood + " " + o.status + " " + o.total);

    }

}

**OUTPUT:**

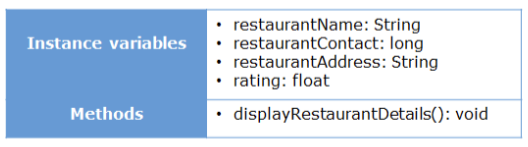
****

**Result:**

The program is Executed successfully

**EX.No 2B**

**Create a new class Restaurant in the Java project SwiftFood with the instance variables and methods mentioned below.**

****

**Method Description**

**displayRestaurantDetails()**

* **Display the details of the restaurant (the values of the member variables)**

**Create an object of the Restaurant class, initialize the instance variables, and invoke the isplayRestaurantDetails() method in the main() method of the Tester class.**

**ALGORITHM:**

STEP 1: Initialize the code

STEP 2: Enter the values

STEP 3: Run the code in jdk and output is displayed

STEP 4: End of execution

**PROGRAM:**

class Resturent {

    String name;

    String address;

    Long phone;

    float rating;

    void display() {

        System.out.println(name + " " + address + " " + phone + " " + rating);

    }

}

public class ResturentEx2B {

    public static void main(String[] args) {

        Resturent r = new Resturent();

        r.name = "KFC";

        r.address = "Pune";

        r.phone = 1234567890L;

        r.rating = 4.5f;

        r.display();

    }

}

**OUTPUT:**

****

**Result:**

The program is Executed successfully

**EX.No 3A**

**SwiftFood also provides the feature for non-registered customers to order food. Add one more constructor to the**

**Customer class to implement this functionality.**

****

**Method Description**

**Customer(String customerName, long contactNumber, String address)**

* **Initialize the customerName, contactNumber and address instance variables appropriately with the values passed to the constructor.**

**Create an object of the Customer class by using the parameterized constructor and display the customer details by**

**invoking the displayCustomerDetails() method in the main() method of the Tester class.**

**ALGORITHM:**

STEP 1: Initialize the code

STEP 2: Enter the values

STEP 3: Run the code in jdk and output is displayed

STEP 4: End of execution

**PROGRAM:**

package Unit2;

class Customer {

    String name;

    String address;

    Long phone;

    Customer(String name, String address, Long phone) {

        this.name = name;

        this.address = address;

        this.phone = phone;

    }

}

public class CustomerClassEx3A {

    public static void main(String[] args) {

        Customer c = new Customer("kani", "Pune", 1234567890L);

        System.out.println(c.name + " " + c.address + " " + c.phone);

    }

}

**OUTPUT:**

****

**Result:**

The program is Executed successfully

**EX.No 3B**

**Modify the Order class created before and add two constructors in the class.**

****

**Method Description**

**Order()**

* **Set the value of status to ” Ordered”**

**Order(int orderId, String orderedFoods)**

* **Initialize the instance variables appropriately with the values passed to the constructor.**
* **Set the value of status to ” Ordered”**

**Create an object of the Order class by using the parameterless constructor and display the value of the status instance variable in the main() method of the Tester class.**

**Create one more object of the Order class by using the parameterized constructor and display the value of orderId, orderFoods and status instance variables in the main() method of the Tester class.**

**ALGORITHM:**

STEP 1: Initialize the code

STEP 2: Enter the values

STEP 3: Run the code in jdk and output is displayed

STEP 4: End of execution

**PROGRAM:**

package Unit2;

class ModOrder {

    int id;

    String orderedFood;

    String status;

    double total;

    ModOrder(int id, String orderedFood){

        this.id = id;

        this.orderedFood = orderedFood;

        this.status = "Ordered";

    }

}

public class modOrderEx3B {

    public static void main(String[] args) {

        ModOrder o1 = new ModOrder(1, "Idli");

        System.out.println(o1.id + " " + o1.orderedFood + " " + o1.status);

    }

}