



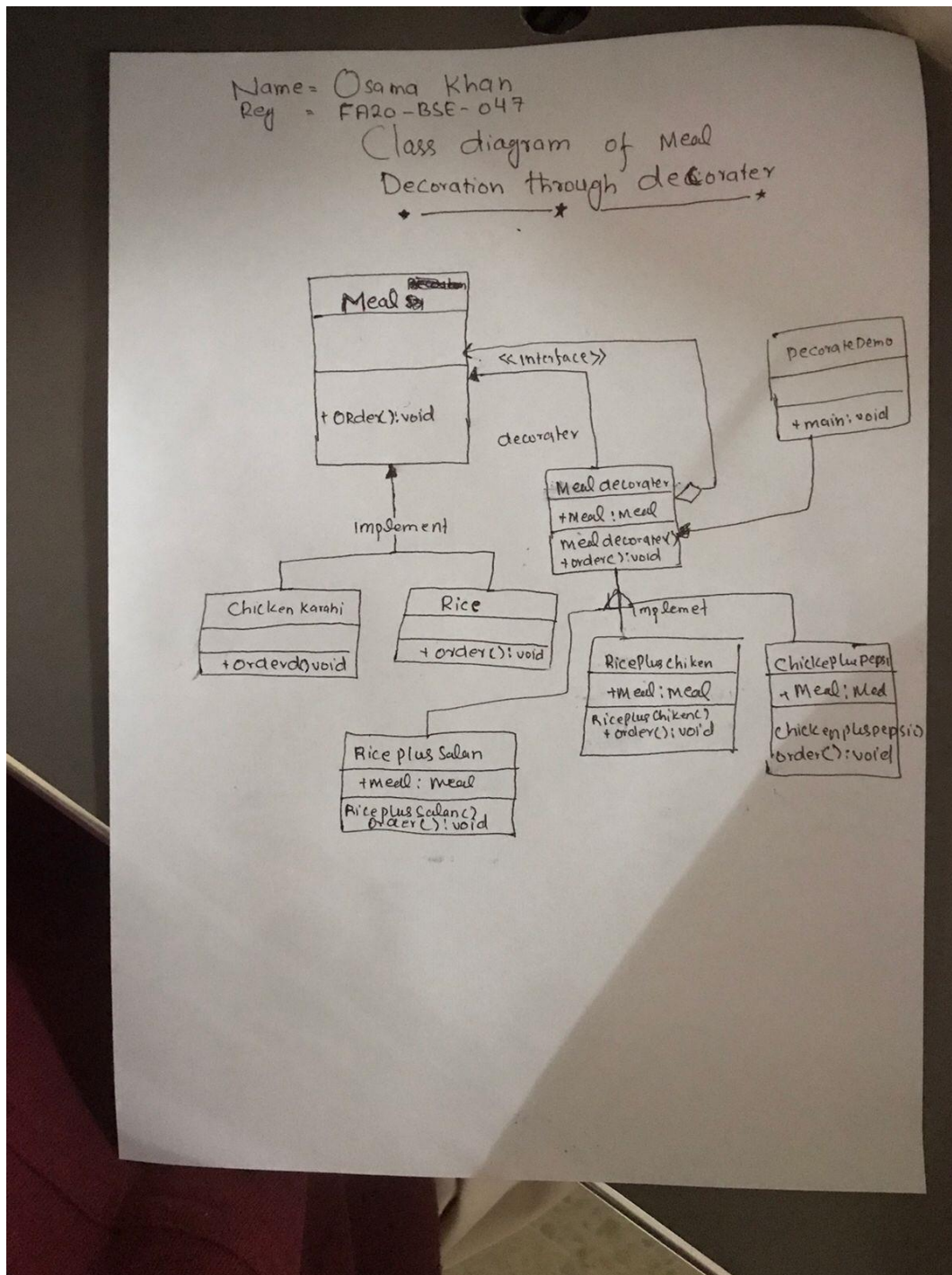
COMSATS University Islamabad

Abbottabad campus

SUBJECT:	Design Pattern
SUBMITTED BY:	Osama Khan
SUBMITTED TO:	Sir Mukhtiar Zamin
REGISTRATION NO:	FA20-BSE-047
DATE:	Thursday, 02 November 2023

Q1: Draw a Class diagram for meal decoration through decorator Pattern and also implement.

Ans: Class Diagram



Implementation:

Classes:

Meal

```
public interface Meal {  
    void order();  
}
```

ChickenKaramahi

```
public class ChickenKaramahi implements Meal {  
    @Override  
    public void order() {  
        System.out.println("Ordered Chicken Karamahi.");  
    }  
}
```

Rice

```
public class Rice implements Meal {  
    @Override  
    public void order() {  
        System.out.println("Ordered Rice.");  
    }  
}
```

MealDecorator

```
public abstract class MealDecorator implements Meal {  
    protected Meal meal;  
  
    public MealDecorator(Meal meal) {  
        this.meal = meal;  
    }  
}
```

@Override

```
public void order() {  
    meal.order();  
}  
}
```

RicePlusChicken

```
public class RicePlusChicken extends MealDecorator {
```

```
    public RicePlusChicken(Meal meal) {  
        super(meal);  
    }
```

@Override

```
public void order() {  
    super.order();  
    System.out.println("Added chicken to the rice meal.");  
}  
}
```

RicePlusSalan

```
public class RicePlusSalan extends MealDecorator {
```

```
    public RicePlusSalan(Meal meal) {  
        super(meal);  
    }
```

@Override

```
public void order() {
```

```
        super.order();  
        System.out.println("Added salan to the rice meal.");  
    }  
}
```

ChickenKarahiPlusPepsi

```
public class ChickenKarahiPlusPepsi extends MealDecorator {
```

```
    public ChickenKarahiPlusPepsi(Meal meal) {  
        super(meal);  
    }
```

```
    @Override
```

```
    public void order() {  
        super.order();  
        System.out.println("Added Pepsi to the chicken karamahi meal.");  
    }  
}
```

MealDemo

```
public class MealDemo {
```

```
    public static void main(String[] args) {
```

```
        Meal chickenKaramahi = new ChickenKaramahi();
```

```
        Meal meal = new ChickenKarahiPlusPepsi(new RicePlusChicken(new Rice()));
```

```
meal.order();
```

```
Meal rice = new Rice();
```

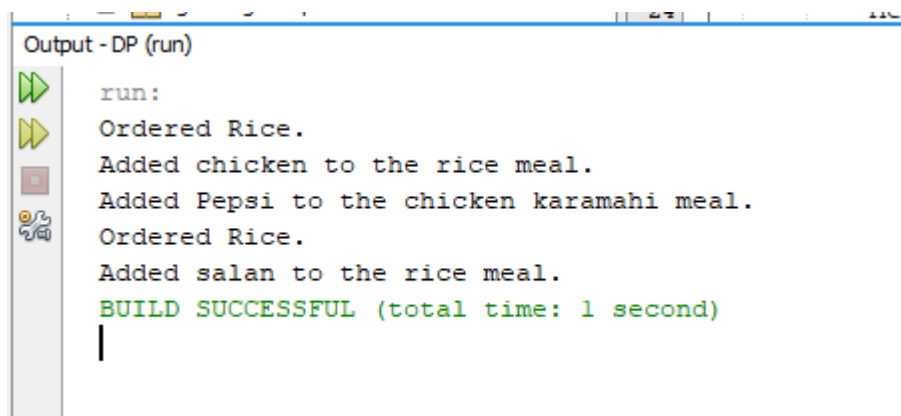
```
Meal ricePlusSalan = new RicePlusSalan(rice);
```

```
ricePlusSalan.order();
```

```
}
```

```
}
```

Output:



```
Output - DP (run)
run:
Ordered Rice.
Added chicken to the rice meal.
Added Pepsi to the chicken karamahi meal.
Ordered Rice.
Added salan to the rice meal.
BUILD SUCCESSFUL (total time: 1 second)
|
```