

Builder Design Pattern

Director calls individual steps required to build the object

MealBuilder defines the name of those steps.

Veg/NonvegBuilders give the different implementations of these building steps

```
public class Meal {  
    String burger;  
    String drink;  
    String sweet;  
    String fries;  
}
```

standard Meal without a constructor

Director:-

```
public class Director {  
    private MealBuilder builder;  
  
    public void setMealBuilder(MealBuilder builder) {  
        this.builder = builder;  
    }  
  
    public Meal construct() {  
        builder.reset();  
        builder.addBurger();  
        builder.addDrink();  
        builder.addFries();  
        builder.addSweet();  
        return builder.getMeal();  
    }  
}
```

Meal Builder :-

↓ Different Implementations

```
public abstract class MealBuilder {  
    protected Meal meal = new Meal();  
  
    public void reset() {  
        meal = new Meal();  
    }  
  
    public abstract void addBurger();  
    public abstract void addSweet();  
    public abstract void addFries();  
    public abstract void addDrink();  
  
    public Meal getMeal() {  
        return meal;  
    }  
}
```

```
public class NonVegLunchBuilder extends MealBuilder {  
  
    @Override  
    public void addBurger() {  
        meal.burger = "Chicken Burger";  
    }  
  
    @Override  
    public void addDrink() {  
        meal.drink = "McFlurry";  
    }  
  
    @Override  
    public void addFries() {  
        meal.fries = "King";  
    }  
  
    @Override  
    public void addSweet() {  
        meal.sweet = "Gulab Jamun";  
    }  
}
```

```
public class VegLunchBuilder extends MealBuilder {  
  
    @Override  
    public void addBurger() {  
        meal.burger = "Cheese Veg Burger";  
    }  
  
    @Override  
    public void addDrink() {  
        meal.drink = "Diet Coke";  
    }  
  
    @Override  
    public void addFries() {  
        meal.fries = "small Fries";  
    }  
  
    @Override  
    public void addSweet() {  
        meal.sweet = "Vanilla Ice Cream";  
    }  
}
```

```

class Test {
    Run | Debug
    public static void main(String[] args) {
        Director dir = new Director();
        dir.setMealBuilder(new VegLunchBuilder());
        Meal meal = dir.construct();
        System.out.println(meal.burger);

        dir.setMealBuilder(new NonVegLunchBuilder());
        meal = dir.construct();
        System.out.println(meal.burger);
    }
}

```

```

Cheese Veg Burger
Chicken Burger

```

first director's meal Builder is Veg Lunch Builder
so it constructs veg meal and returns Veg Burger.

At next, its meal Builder is set to NonVeg builder
which sets or creates non-veg meal in construct with
following set algorithm and returns Chicken Burger.

Steps of creation:- Director class.

Individual steps defined:- In Builder (meal Builder)

Individual Steps different Behaviour in different Builder classes.

Problem Statement:- to streamline Construction's Process.

