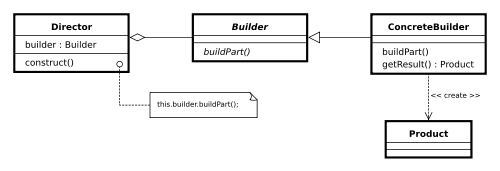
**Builder Design (Creational) Pattern - 2022**

**GOF** : **Separate the construction of a complex object from its representation so that the same construction process can create different representations.**

# **Class Diagram**



**ConcreteBuilder.java**

public class ConcreteBuilder implements IBuilder {  
 private Product product = new Product();  
  
 @Override  
 public void buildBody() {  
 System.*out*.println("... body building ...");  
 product.setName("Prod Name 1");  
 }  
  
 @Override  
 public Product getProduct() {  
 return product;  
 }  
}

**IBuilder.java**

public interface IBuilder {  
 void buildBody();  
 Product getProduct();  
}

**Director.java**

**Product.java**

**@Data**

public class Product {  
 private String name;  
 private String type;  
}

public class Director {  
 private **IBuilder builder = null**;  
 public Director(IBuilder builder ) {  
 this.builder = builder;  
 }  
  
 public void build() {  
 builder.buildBody();  
 }  
  
 public Product getProduct() {  
 return builder.getProduct();  
 }  
}

**Test.java**

public class Test {  
  
 public static void main(String[] args) {  
 Director director = new Director( new ConcreteBuilder());  
 director.build();  
 Product product = director.getProduct();  
 System.*out*.println("Product Name :::"+product.getName());  
  
 }  
}

**Using Fluent Interface**

**public** **class** Car {

**private** **int** quantity;

// Optional Parameters

**private** **boolean** isAVS;

**private** **boolean** isAutomaticGear;

**private** **boolean** isBulletProof;

**public** **static** **class** CarBuilder {

**private** **int** quantity;

**private** **boolean** isAVS;

**private** **boolean** isAutomaticGear;

**private** **boolean** isBulletProof;

**public** CarBuilder(**int** quantity) {

**this**.quantity = quantity;

}

**public** CarBuilder avsSystem(**boolean** flag) {

isAVS = flag;

**return** **this**;

}

**public** CarBuilder automaticGear(**boolean** flag) {

isAutomaticGear = flag;

**return** **this**;

}

**public** CarBuilder bulletProof(**boolean** flag) {

isBulletProof = flag;

**return** **this**;

}

**public** Car build() {

**return** **new** Car(**this**);

}

}

**private** Car(CarBuilder builder) {

isAVS = builder.isAVS;

isAutomaticGear = builder.isAutomaticGear;

isBulletProof = builder.isBulletProof;

}

}

**public** **class** TestCar {

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

Car car = **new** Car.CarBuilder(1).automaticGear(**true**).bulletProof(**true**).build();

System.***out***.println(car);

}

}