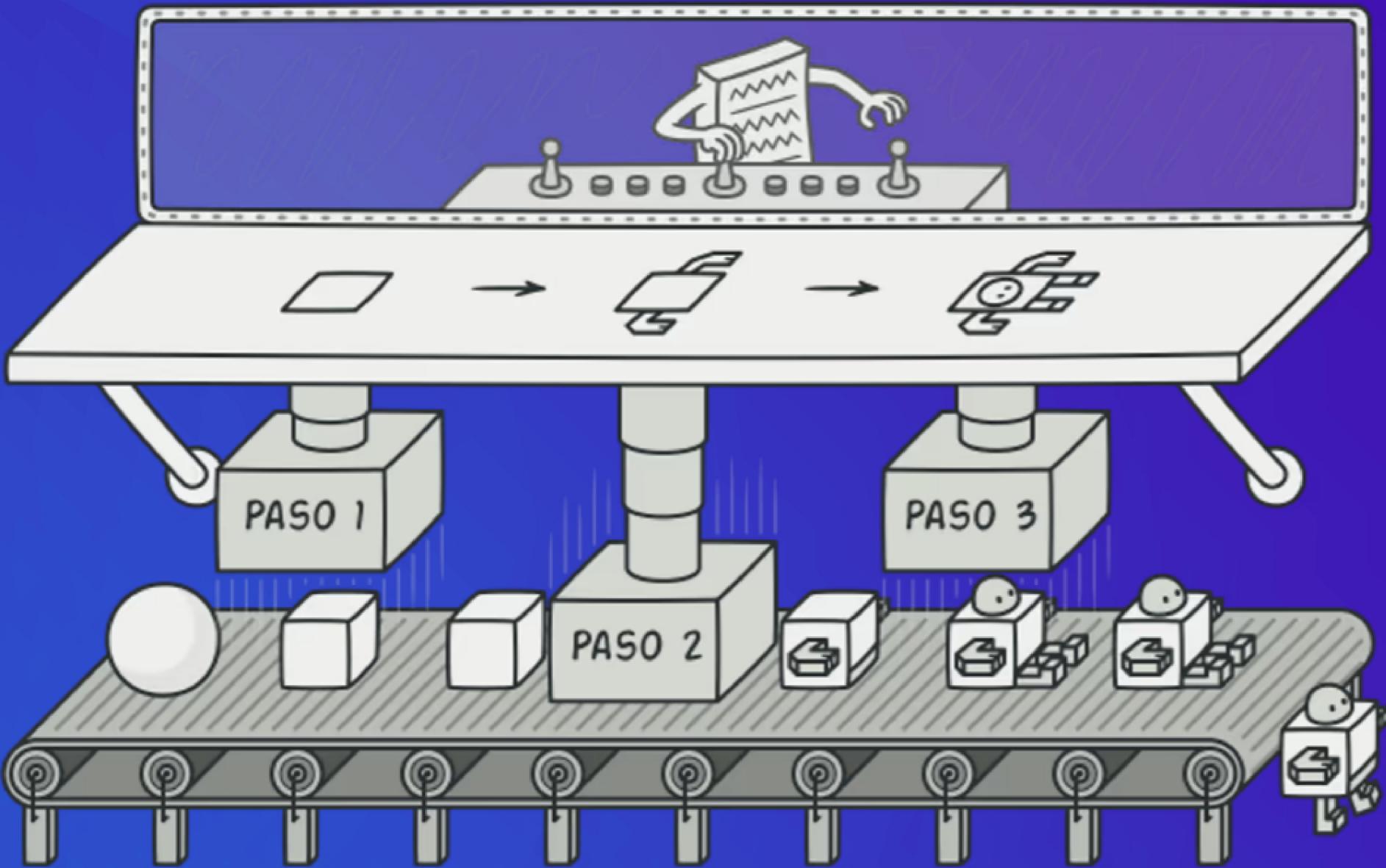


Patrón de Diseño **Builder**



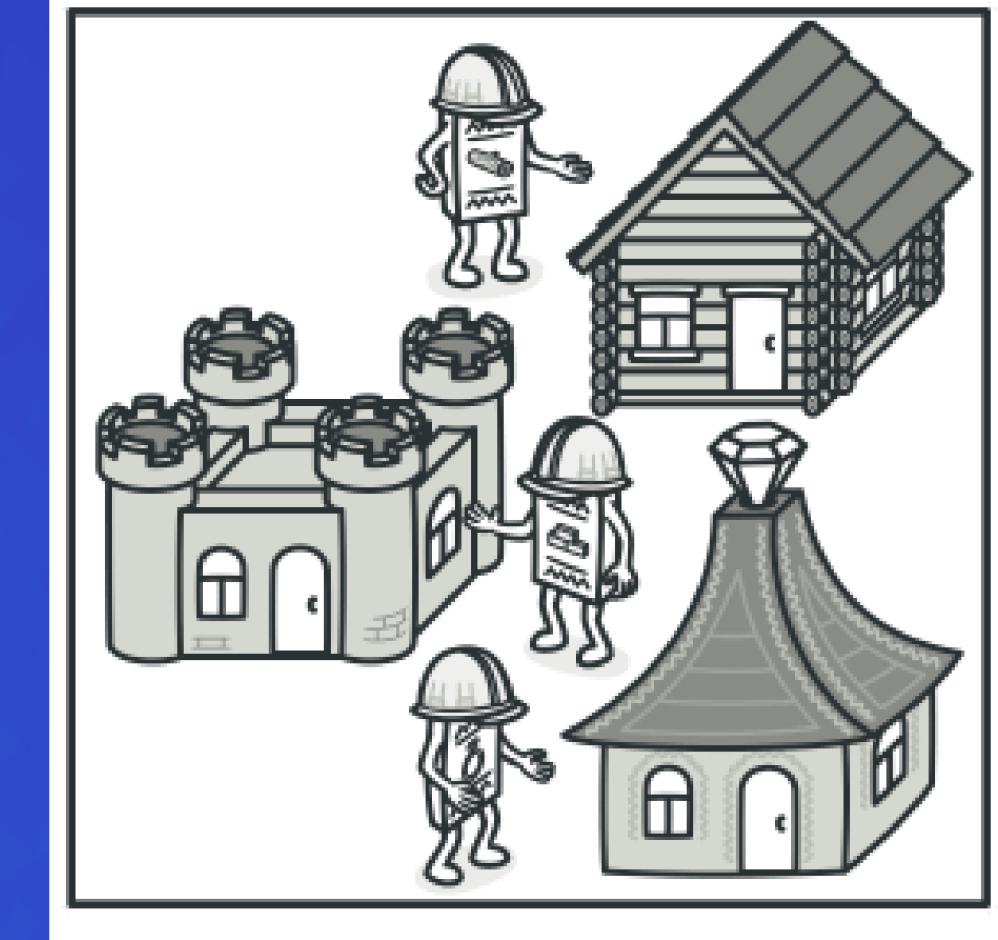
¿Qué es?

Es un patrón de diseño que nos permite construir objetos de diferentes tipos y representaciones usando el mismo código de construcción



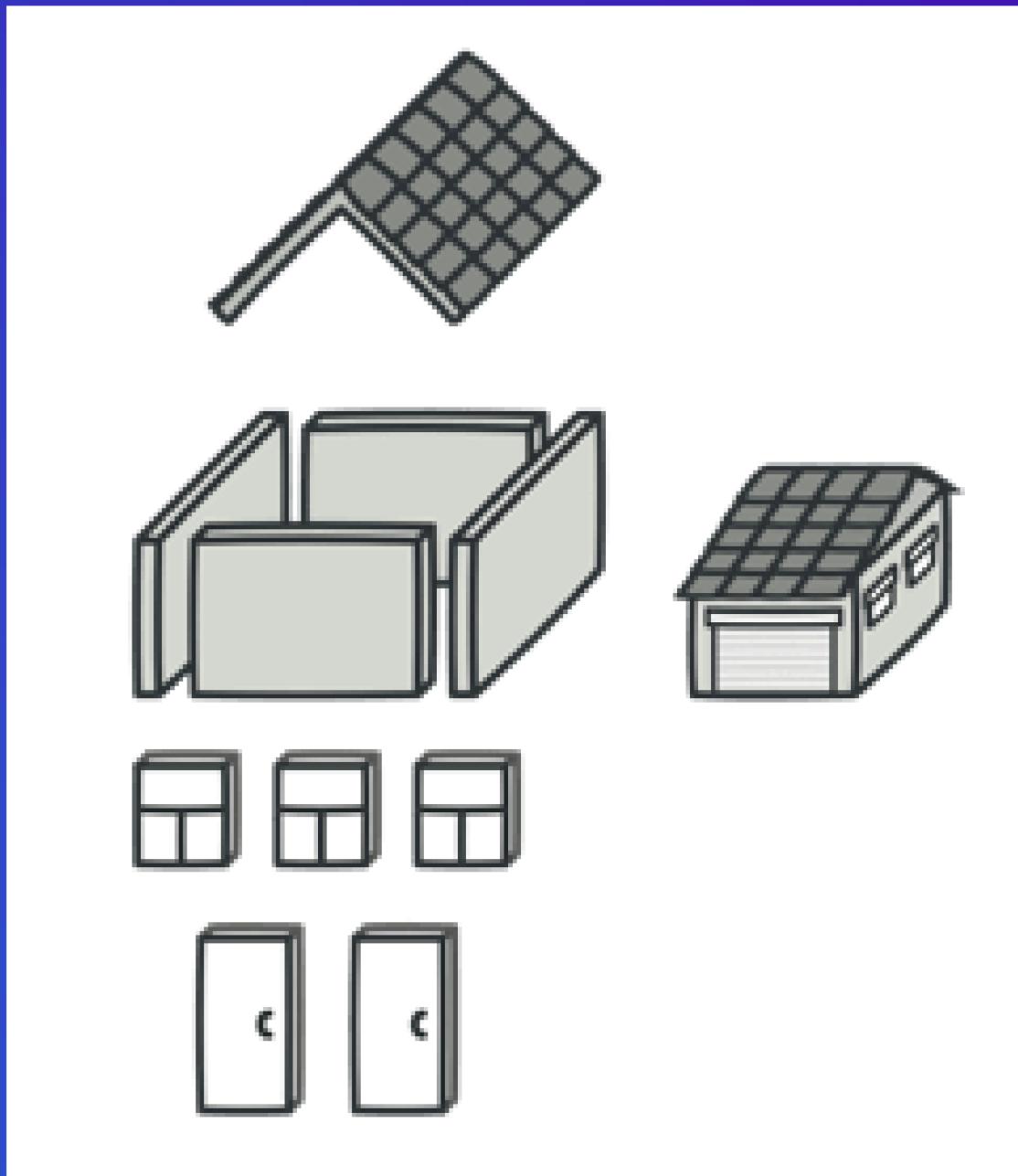
¿El problema?

En el caso que se tenga un objeto complejo con varios campos y objetos anidados, todo queda debajo de un enorme constructor con muchos parámetros disperso por todo el código



La solución

En vez de tener el código de construcción en una sola clase, se crean objetos independientes llamados constructores



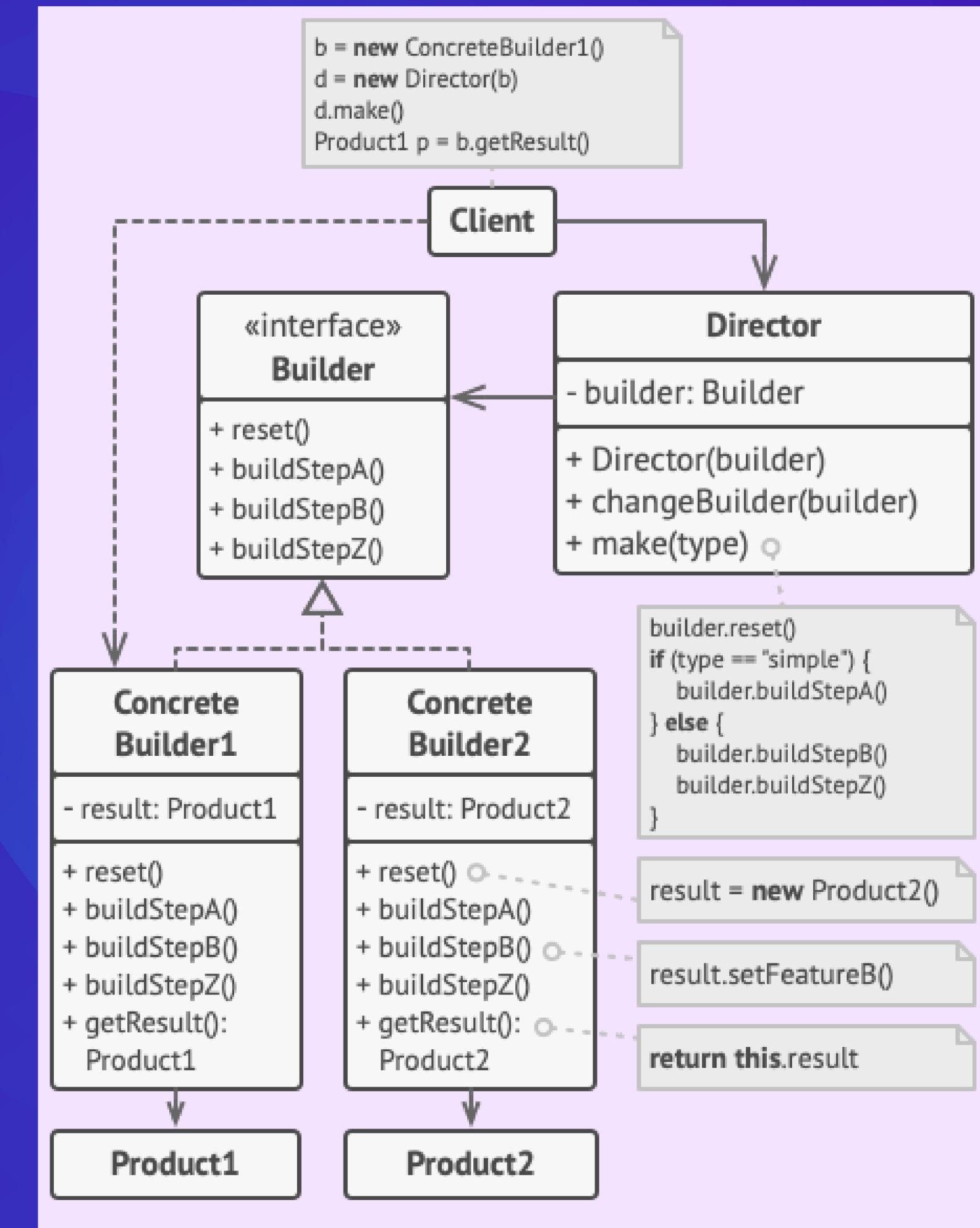
Pros

- Se puede reutilizar el mismo código de construcción
- Construir objetos paso a paso
- Principio de responsabilidad única

Contras

- Aumenta la complejidad
- Más lento

Estructura



Ejemplo

```
1  class Car:
2      def __init__(self):
3          self.brand = None
4          self.model = None
5          self.year = None
6          self.color = None
7
8      def __str__(self):
9          return f"Car: {self.year} {self.brand} {self.model}, Color: {self.color}"
10
11 class CarBuilder:
12     def set_brand(self, brand):
13         pass
14
15     def set_model(self, model):
16         pass
17
18     def set_year(self, year):
19         pass
20
21     def set_color(self, color):
22         pass
23
24     def build(self):
```

Ejemplo

```
27 class ConcreteCarBuilder(CarBuilder):
28     def __init__(self):
29         self.car = Car()
30
31     def set_brand(self, brand):
32         self.carbrand = brand
33
34     def set_model(self, model):
35         self.car.model = model
36
37     def set_year(self, year):
38         self.car.year = year
39
40     def set_color(self, color):
41         self.car.color = color
42
43     def build(self):
44         return self.car
45
46 if __name__ == "__main__":
47     builder = ConcreteCarBuilder()
48     builder.set_brand("Toyota")
49     builder.set_model("Corola")
50     builder.set_year(2023)
51     builder.set_color("Blanco")
52     car = builder.build()
53     print(car)
```

¿Alguna duda, consulta o pregunta?

GRACIAS!!



Cecilia Lingua