**EXERCISE 3:**

**BUILDER PATTERN EXAMPLE**

// Computer.java  
class Computer {  
 // Attributes of the Computer  
 private String CPU;  
 private String RAM;  
 private String storage;  
 private String GPU;  
 private String motherboard;  
 private String powerSupply;  
 private String coolingSystem;  
  
 // Private constructor to be called by the Builder  
 private Computer(Builder builder) {  
 this.CPU = builder.CPU;  
 this.RAM = builder.RAM;  
 this.storage = builder.storage;  
 this.GPU = builder.GPU;  
 this.motherboard = builder.motherboard;  
 this.powerSupply = builder.powerSupply;  
 this.coolingSystem = builder.coolingSystem;  
 }  
  
 // Static nested Builder class  
 public static class Builder {  
 private String CPU;  
 private String RAM;  
 private String storage;  
 private String GPU;  
 private String motherboard;  
 private String powerSupply;  
 private String coolingSystem;  
  
 // Methods to set each attribute  
 public Builder setCPU(String CPU) {  
 this.CPU = CPU;  
 return this;  
 }  
  
 public Builder setRAM(String RAM) {  
 this.RAM = RAM;  
 return this;  
 }  
  
 public Builder setStorage(String storage) {  
 this.storage = storage;  
 return this;  
 }  
  
 public Builder setGPU(String GPU) {  
 this.GPU = GPU;  
 return this;  
 }  
  
 public Builder setMotherboard(String motherboard) {  
 this.motherboard = motherboard;  
 return this;  
 }  
  
 public Builder setPowerSupply(String powerSupply) {  
 this.powerSupply = powerSupply;  
 return this;  
 }  
  
 public Builder setCoolingSystem(String coolingSystem) {  
 this.coolingSystem = coolingSystem;  
 return this;  
 }  
  
 // Build method to return an instance of Computer  
 public Computer build() {  
 return new Computer(this);  
 }  
 }  
  
 // Method to display the configuration of the computer  
 public void showConfiguration() {  
 System.*out*.println("CPU: " + CPU);  
 System.*out*.println("RAM: " + RAM);  
 System.*out*.println("Storage: " + storage);  
 System.*out*.println("GPU: " + GPU);  
 System.*out*.println("Motherboard: " + motherboard);  
 System.*out*.println("Power Supply: " + powerSupply);  
 System.*out*.println("Cooling System: " + coolingSystem);  
 }  
}  
  
public class BuilderPatternExample {  
 public static void main(String[] args) {  
 Computer gamingComputer = new Computer.Builder()  
 .setCPU("Intel Core i9")  
 .setRAM("32GB")  
 .setStorage("1TB SSD")  
 .setGPU("NVIDIA GeForce RTX 3080")  
 .setMotherboard("ASUS ROG Maximus XII")  
 .setPowerSupply("750W")  
 .setCoolingSystem("Liquid Cooling")  
 .build();  
  
 System.*out*.println("Gaming Computer Configuration:");  
 gamingComputer.showConfiguration();  
 Computer officeComputer = new Computer.Builder()  
 .setCPU("Intel Core i3")  
 .setRAM("8GB")  
 .setStorage("256GB SSD")  
 .build();  
 System.*out*.println("\nOffice Computer Configuration:");  
 officeComputer.showConfiguration();  
 }  
}