Design patterns and principles

SuperSet ID:6412063

Exercise 3: Implementing the Builder Pattern

Code:

Computer.java

public class Computer {

    private String cpu;

    private String ram;

    private String storage;

    private String graphicsCard;

    private String keyboard;

    private String monitor;

    private Computer(Builder builder) {

        this.cpu = builder.cpu;

        this.ram = builder.ram;

        this.storage = builder.storage;

        this.graphicsCard = builder.graphicsCard;

        this.keyboard = builder.keyboard;

        this.monitor = builder.monitor;

    }

    public String getCpu() { return cpu; }

    public String getRam() { return ram; }

    public String getStorage() { return storage; }

    public String getGraphicsCard() { return graphicsCard; }

    public String getKeyboard() { return keyboard; }

    public String getMonitor() { return monitor; }

    public String toString() {

        return "Computer Configuration:\n" +

               "CPU: " + cpu + "\n" +

               "RAM: " + ram + "\n" +

               "Storage: " + storage + "\n" +

               "Graphics Card: " + graphicsCard + "\n" +

               "Keyboard: " + keyboard + "\n" +

               "Monitor: " + monitor + "\n";

    }

    public static class Builder {

        private String cpu;

        private String ram;

        private String storage;

        private String graphicsCard;

        private String keyboard;

        private String monitor;

        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 setGraphicsCard(String graphicsCard) {

            this.graphicsCard = graphicsCard;

            return this;

        }

        public Builder setKeyboard(String keyboard) {

            this.keyboard = keyboard;

            return this;

        }

        public Builder setMonitor(String monitor) {

            this.monitor = monitor;

            return this;

        }

        public Computer build() {

            return new Computer(this);

        }

    }

}

ComputerTest.java

public class ComputerTest {

    public static void main(String[] args) {

        // Basic computer

        Computer basicComputer = new Computer.Builder()

                .setCpu("Intel i3")

                .setRam("4GB")

                .setStorage("256GB SSD")

                .build();

        System.out.println("Basic Computer:");

        System.out.println(basicComputer);

        // High-end gaming computer

        Computer gamingComputer = new Computer.Builder()

                .setCpu("Intel i9")

                .setRam("32GB")

                .setStorage("1TB SSD")

                .setGraphicsCard("NVIDIA RTX 4090")

                .setKeyboard("Mechanical RGB")

                .setMonitor("4K 27 inch")

                .build();

        System.out.println("Gaming Computer:");

        System.out.println(gamingComputer);

    }

}

Output:

A white background with black text

Description automatically generated