**Exercise 3: Implementing the Builder Pattern**

public class Main {

    static class Computer {

        private String CPU;

        private String RAM;

        private String storage;

        private String graphicsCard;

        private String motherboard;

        private String powerSupply;

        private Computer(Builder builder) {

            this.CPU = builder.CPU;

            this.RAM = builder.RAM;

            this.storage = builder.storage;

            this.graphicsCard = builder.graphicsCard;

            this.motherboard = builder.motherboard;

            this.powerSupply = builder.powerSupply;

        }

        public static class Builder {

            private String CPU;

            private String RAM;

            private String storage;

            private String graphicsCard;

            private String motherboard;

            private String powerSupply;

            public Builder(String CPU, String RAM) {

                this.CPU = CPU;

                this.RAM = RAM;

            }

            public Builder setStorage(String storage) {

                this.storage = storage;

                return this;

            }

            public Builder setGraphicsCard(String graphicsCard) {

                this.graphicsCard = graphicsCard;

                return this;

            }

            public Builder setMotherboard(String motherboard) {

                this.motherboard = motherboard;

                return this;

            }

            public Builder setPowerSupply(String powerSupply) {

                this.powerSupply = powerSupply;

                return this;

            }

            public Computer build() {

                return new Computer(this);

            }

        }

        @Override

        public String toString() {

            return "Computer [CPU=" + CPU + ", RAM=" + RAM + ", Storage=" + storage +

                   ", GraphicsCard=" + graphicsCard + ", Motherboard=" + motherboard +

                   ", PowerSupply=" + powerSupply + "]";

        }

    }

    public static void main(String[] args) {

        Computer basicComputer = new Computer.Builder("Intel i3", "8GB").build();

        System.out.println("Basic Computer:\n" + basicComputer);

        Computer gamingComputer = new Computer.Builder("Intel i7", "16GB")

                .setGraphicsCard("NVIDIA RTX 3060")

                .setStorage("1TB SSD")

                .setMotherboard("ASUS ROG")

                .setPowerSupply("650W")

                .build();

        System.out.println("\nGaming Computer:\n" + gamingComputer);

        Computer workstation = new Computer.Builder("AMD Ryzen 9", "32GB")

                .setGraphicsCard("NVIDIA Quadro")

                .setStorage("2TB NVMe SSD")

                .setMotherboard("MSI Creator")

                .setPowerSupply("850W")

                .build();

        System.out.println("\nWorkstation:\n" + workstation);

    }

}

Output: