Exercise 3: Builder Pattern- Computer Setup System

Approach and Understanding:

In this exercise, I created a **simple system to build computers** by allowing users to **choose parts like CPU, RAM, and Storage.**

Sometimes we don’t want to build everything at once — so the **Builder Pattern** helps us to make a step-by-step setup in a clean way.

**Code**

import java.util.Scanner;

class PC { String c; int r;

int s;

private PC(Make b) { this.c = b.c;

this.r = b.r; this.s = b.s;

}

void show() {

System.out.println("\nHere’s your PC setup:"); System.out.println("CPU: " + c); System.out.println("RAM: " + r + " GB"); System.out.println("Storage: " + s + " GB");

static class Make { String c;

int r; int s;

Make setC(String c) { this.c = c;

return this;

}

Make setR(int r) { this.r = r;

return this;

}

Make setS(int s) { this.s = s; return this;

}

PC done() {

return new PC(this);

}

public class PCApp {

public static void main(String[] args) { Scanner sc = new Scanner(System.in);

System.out.print("What CPU do you want? "); String cpu = sc.nextLine();

System.out.print("How much RAM? (GB) "); int ram = sc.nextInt();

System.out.print("How much Storage? (GB) "); int storage = sc.nextInt();

PC myPC = new PC.Make()

.setC(cpu)

.setR(ram)

.setS(storage)

.done();

myPC.show();

sc.close();

**OUTPUT:**

