Week 7 - S7 - Core OOP - Polymorphism - Assignment Problem(HW)

Name:Ramesh Harisabapathi Chettiar

Date of Submission:24/09/25

**PROBLEM 1: Hotel Booking System**

**Concept: Method Overloading**

**You're building a hotel reservation system that calculates room prices in various ways:**

**● Standard booking (just room type and nights)**

**● Seasonal booking (room type, nights + seasonal multiplier)**

**● Corporate booking (room type, nights + corporate discount + meal package)**

**● Wedding package (room type, nights + guest count + decoration fee + catering**

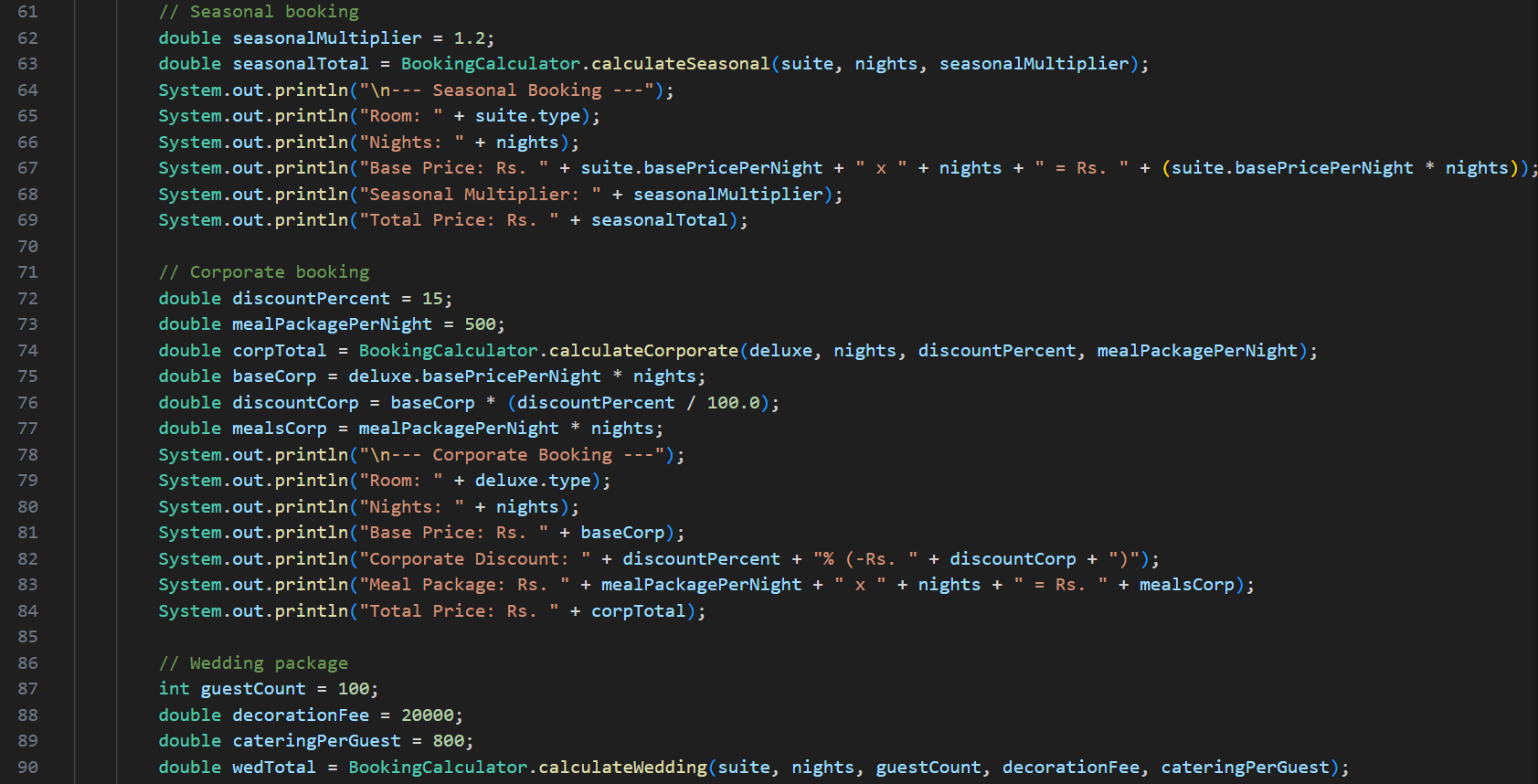
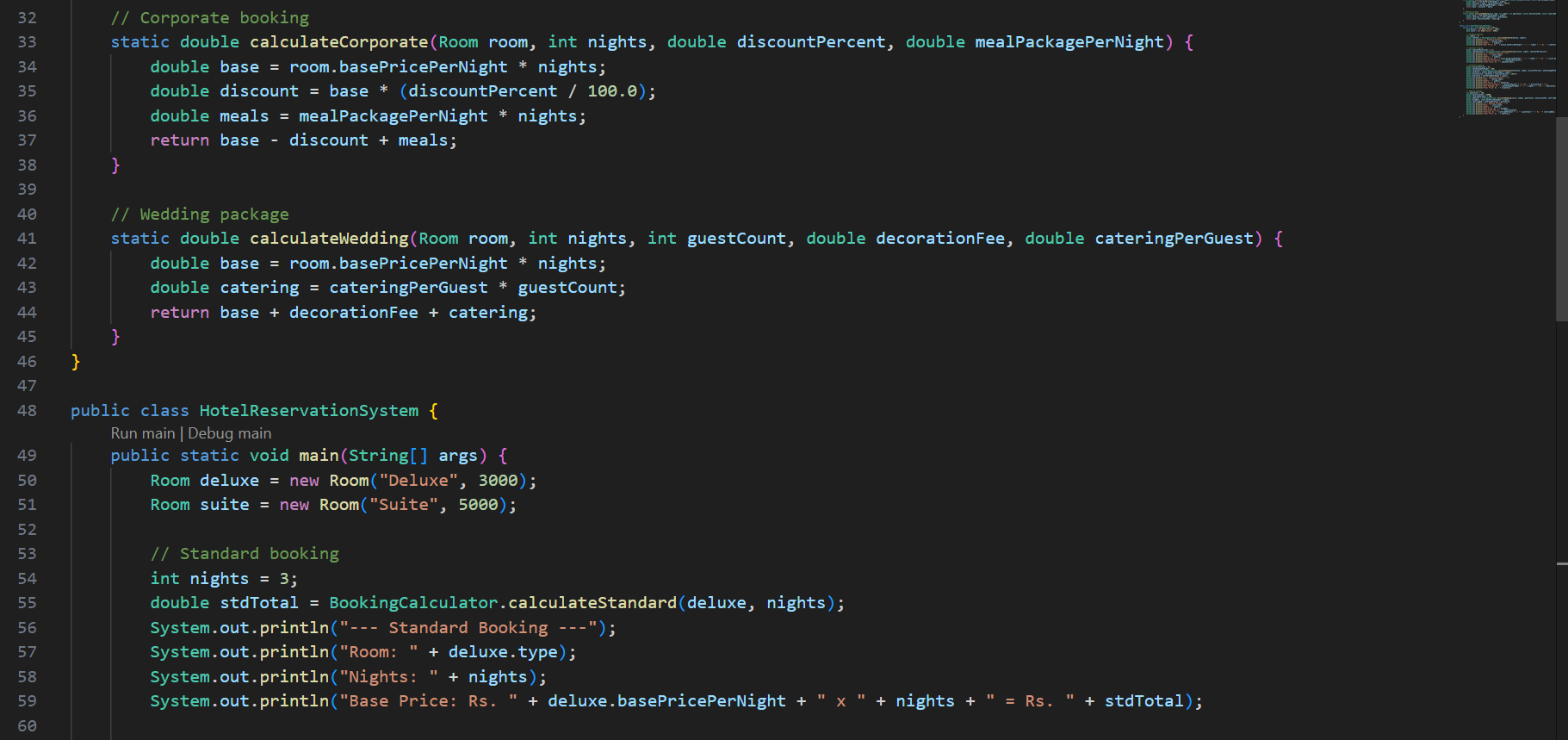
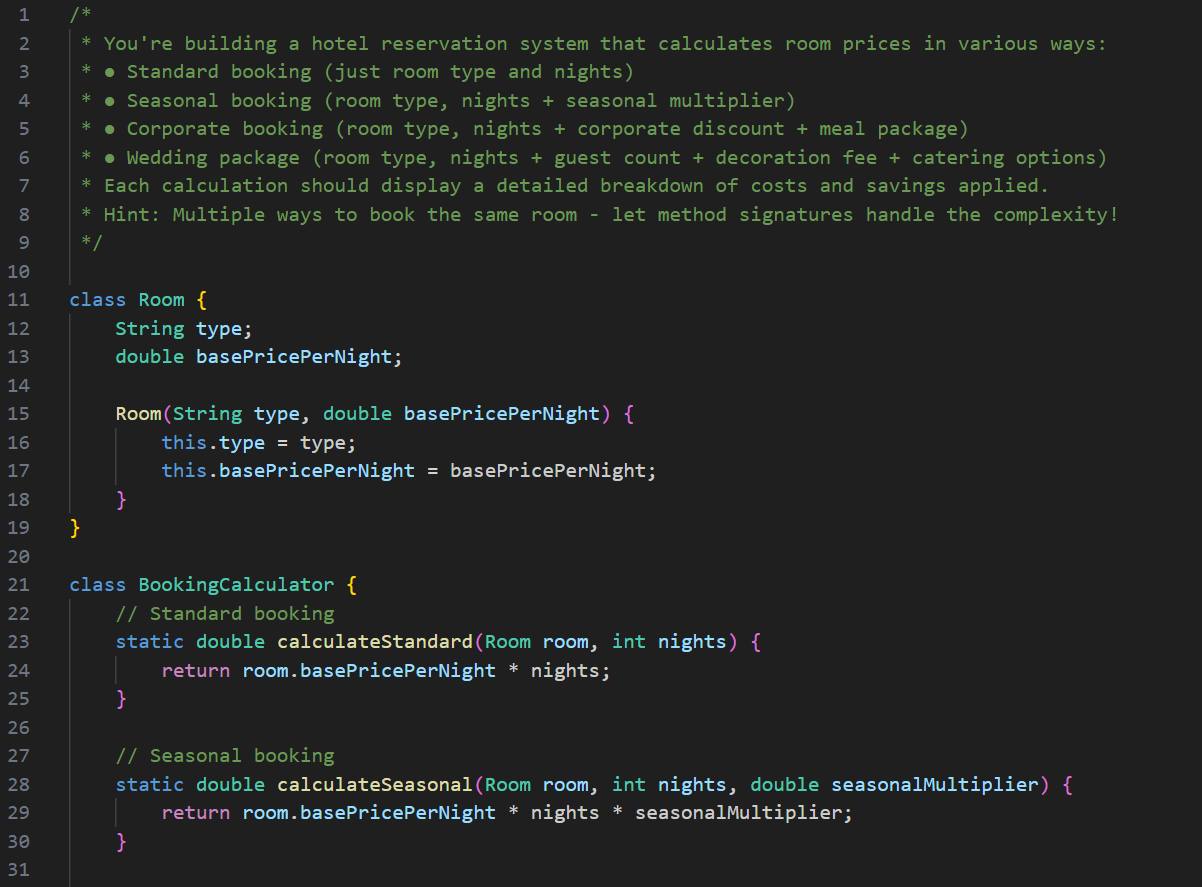
**options)**

**Each calculation should display a detailed breakdown of costs and savings applied.**

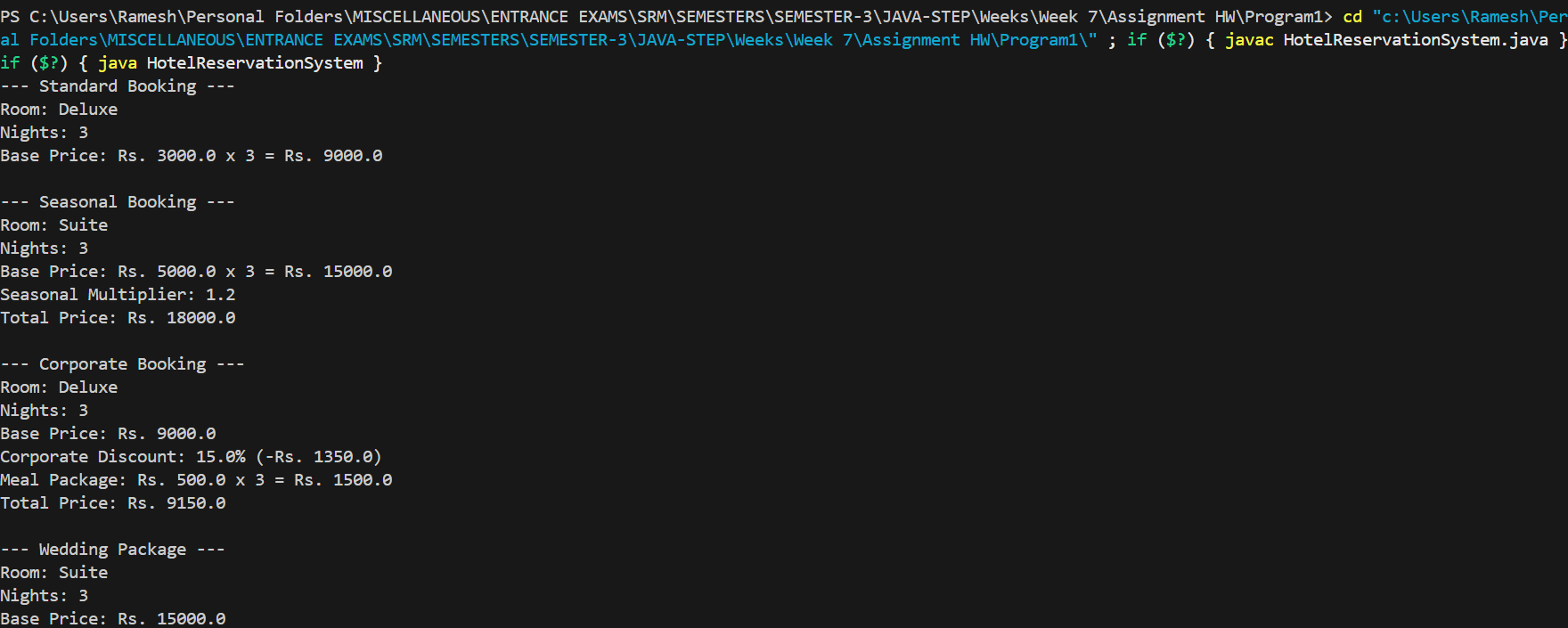
**Hint: Multiple ways to book the same room - let method signatures handle the**

**complexity!**

**PROGRAM🡪**

****

**OUTPUT🡪**

****

**PROBLEM 2: Online Learning Platform**

**Concept: Method Overriding**

**Create an educational content system where different course types display progress**

**differently:**

**● Video courses show completion percentage and watch time**

**● Interactive courses show quiz scores and hands-on projects completed**

**● Reading courses show pages read and note-taking progress**

**● Certification courses show exam attempts and certification status**

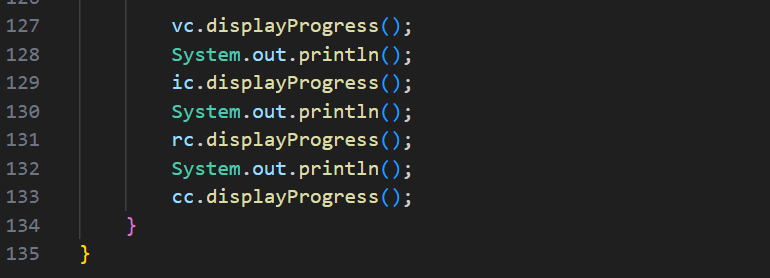
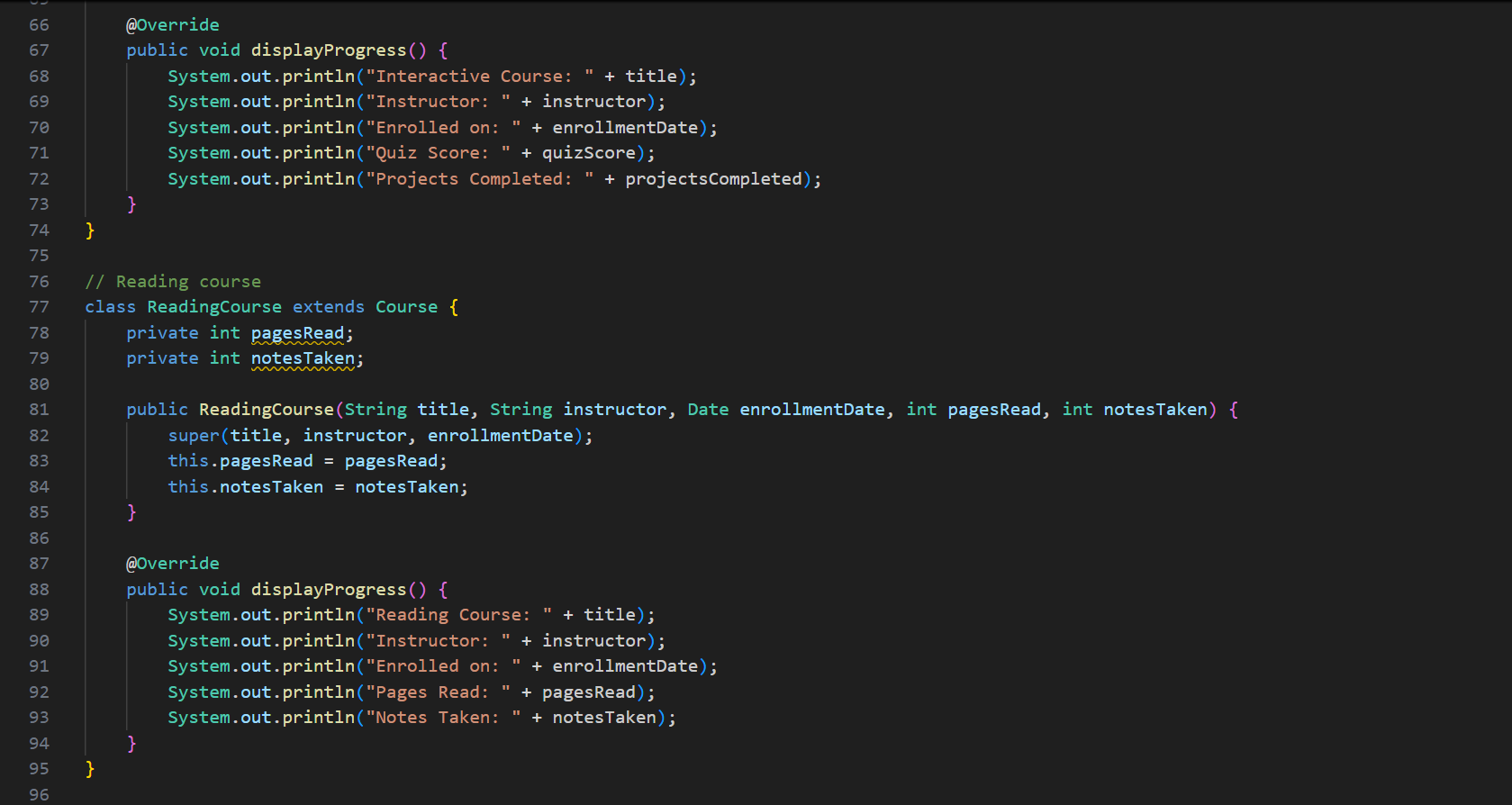
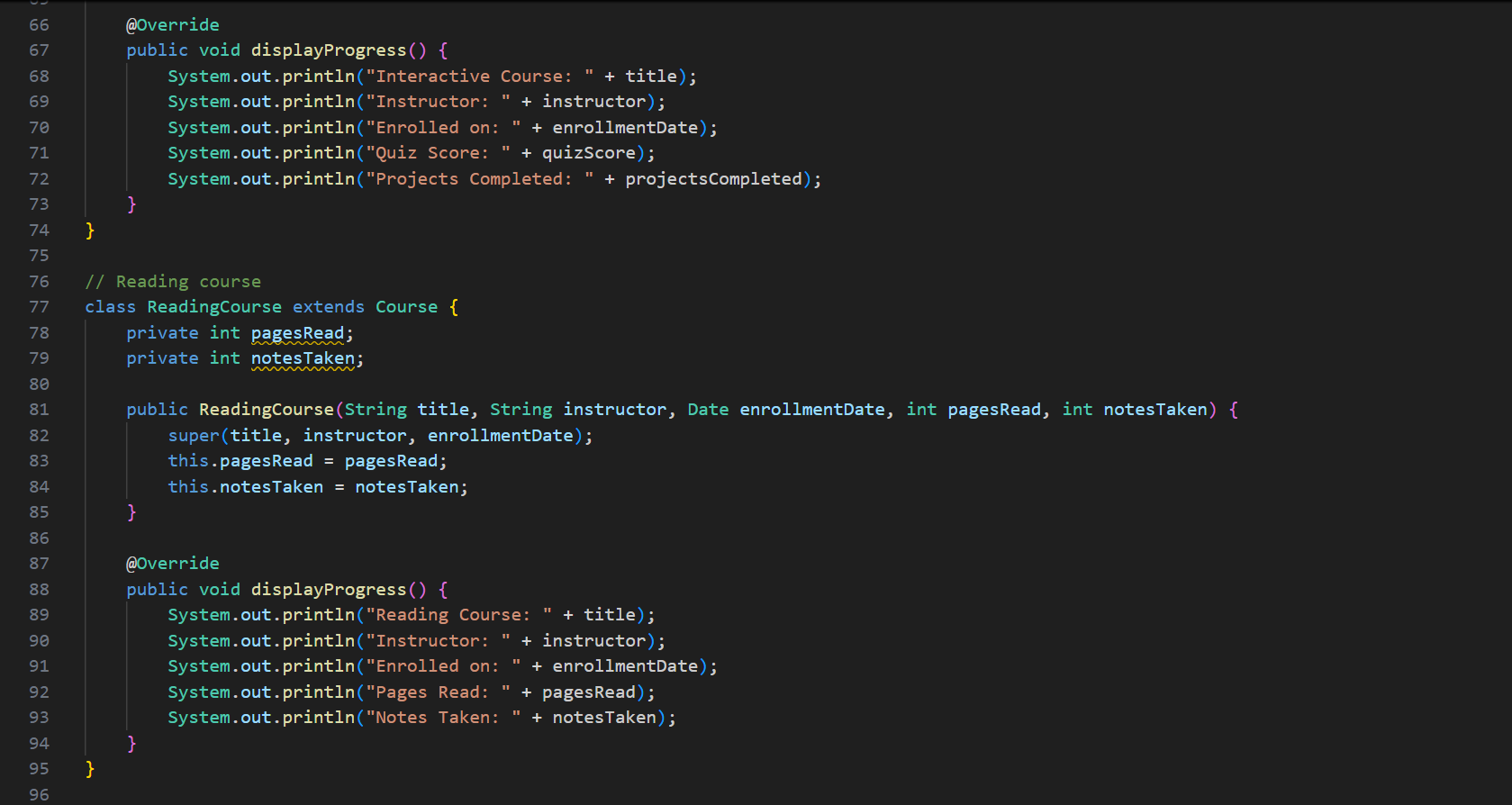
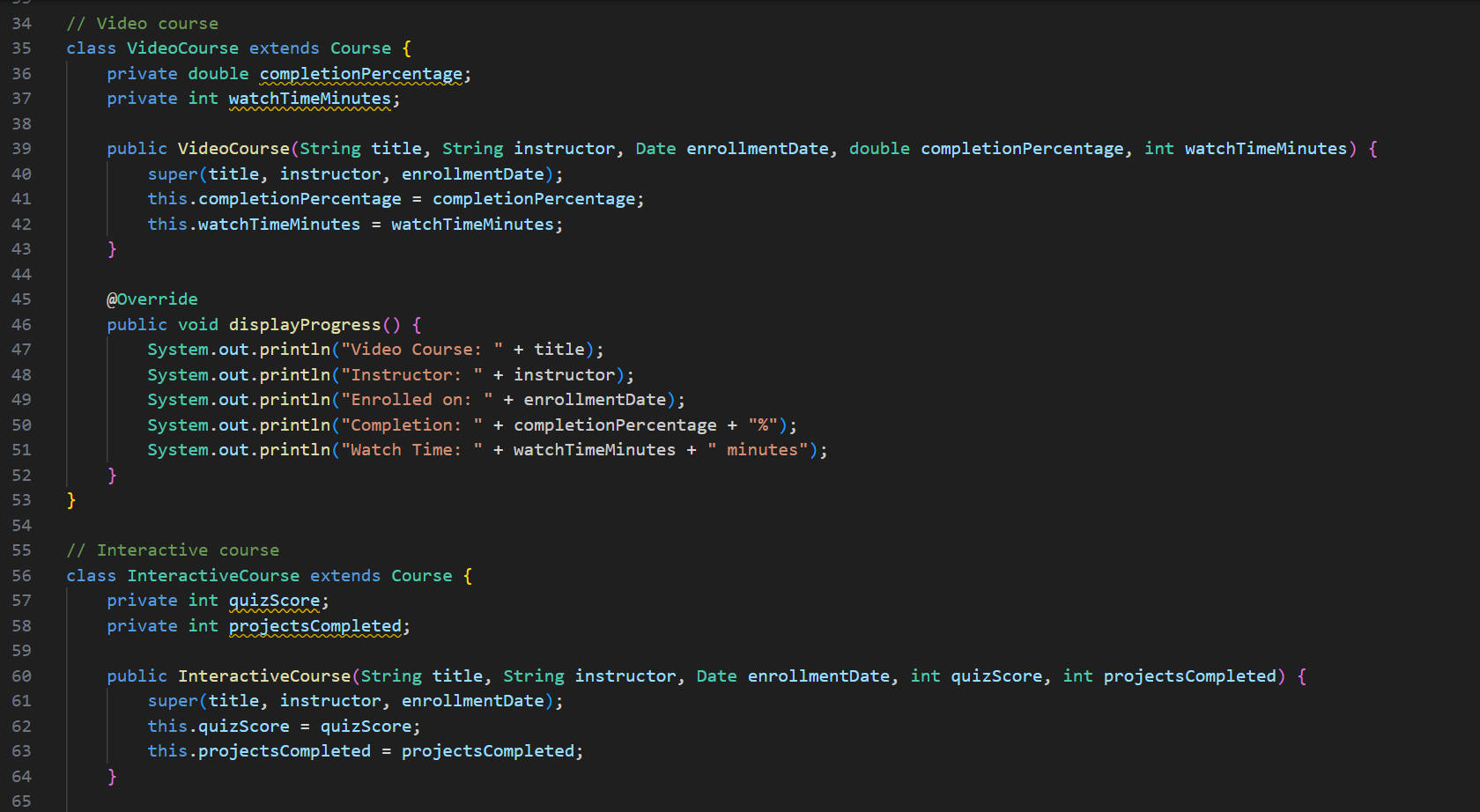
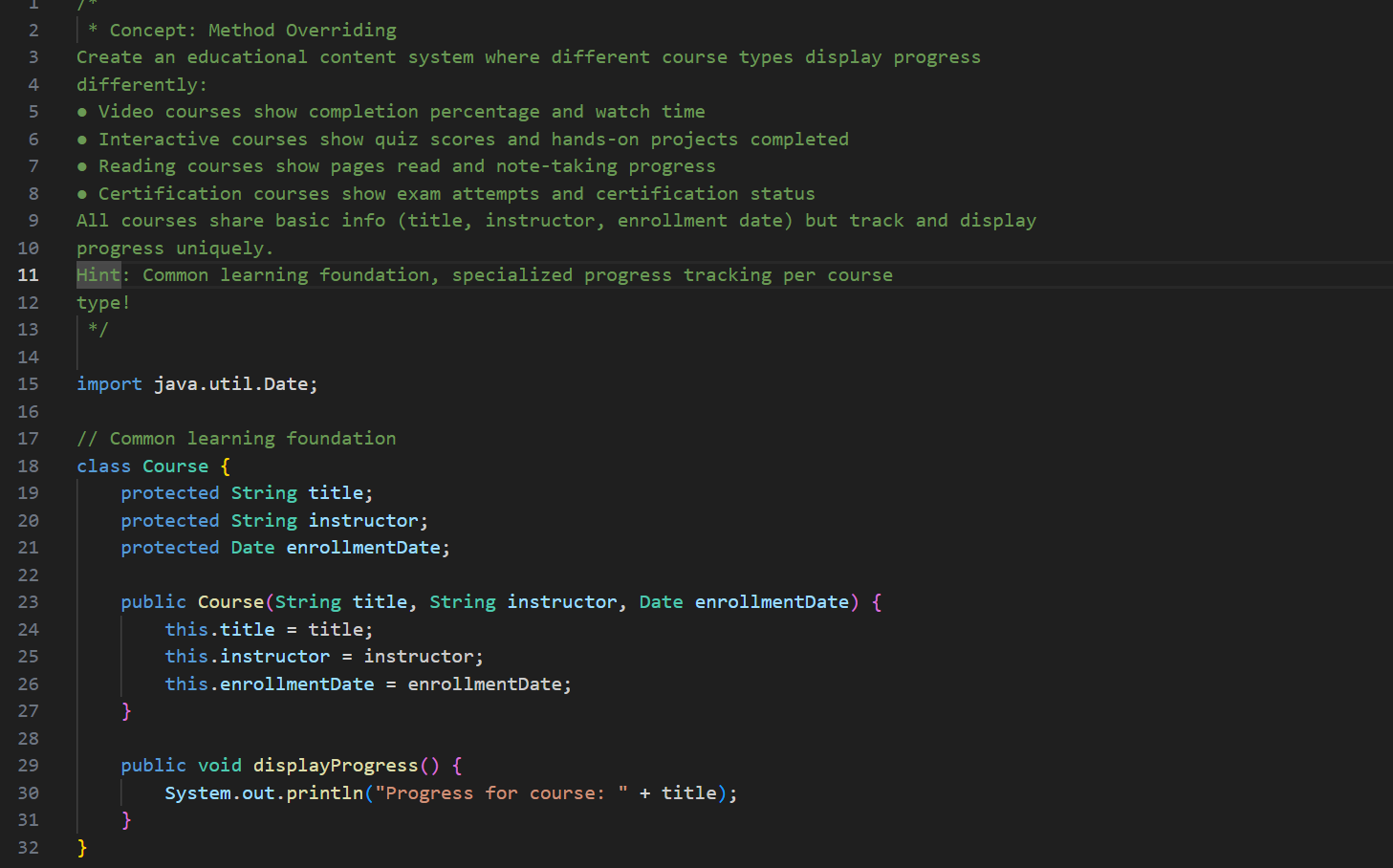
**All courses share basic info (title, instructor, enrollment date) but track and display**

**progress uniquely.**

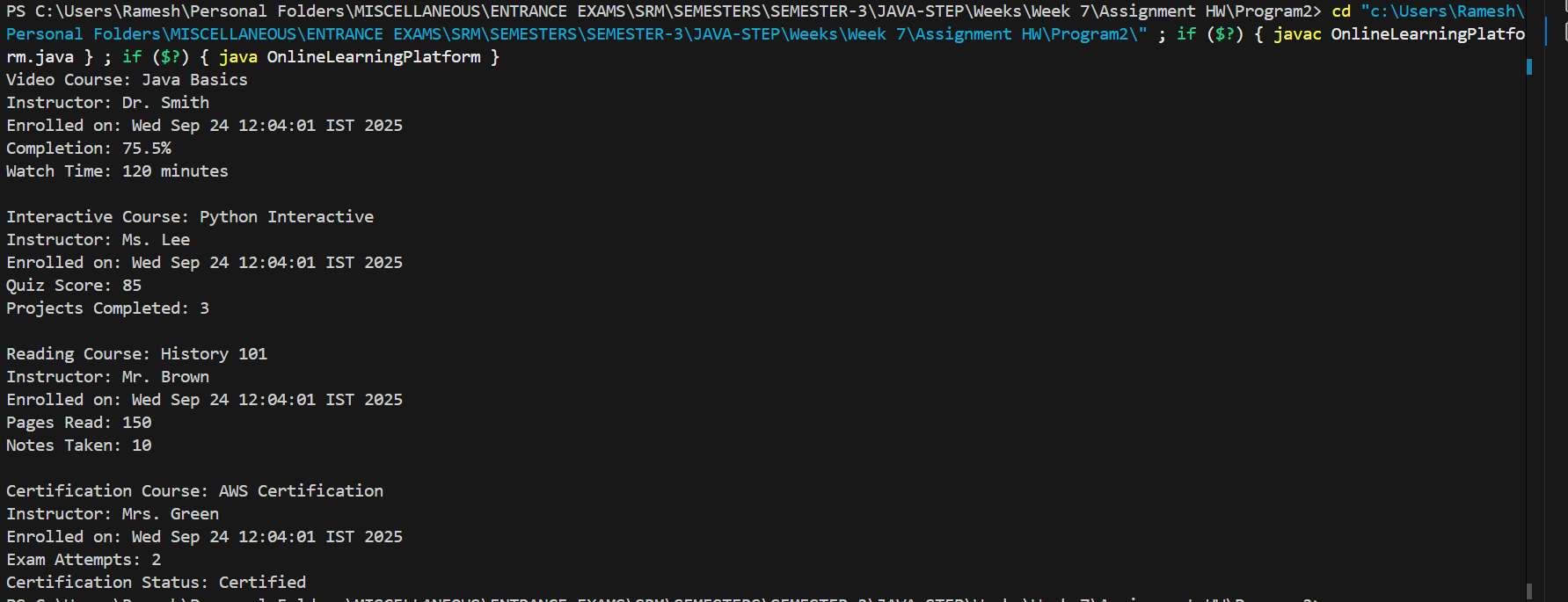
**Hint: Common learning foundation, specialized progress tracking per course**

**type!**

**PROGRAM🡪**

****

**OUTPUT🡪**

****

**PROBLEM 3: Transportation Fleet Management**

**Concept: Dynamic Method Dispatch**

**Design a city transport system with different vehicle types:**

**● Buses follow fixed routes and track passenger capacity**

**● Taxis provide door-to-door service and calculate fare by distance**

**● Trains operate on schedules and manage multiple car capacity**

**● Bikes are available for short-distance eco-friendly trips**

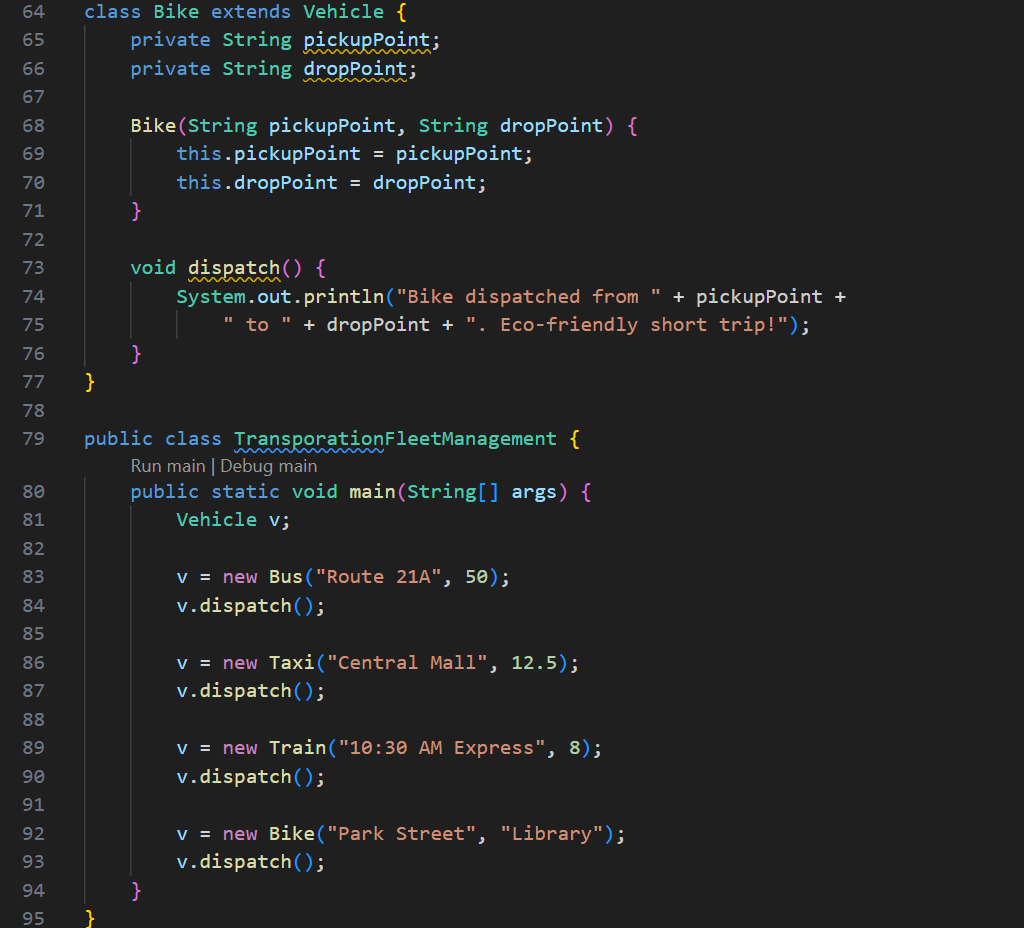
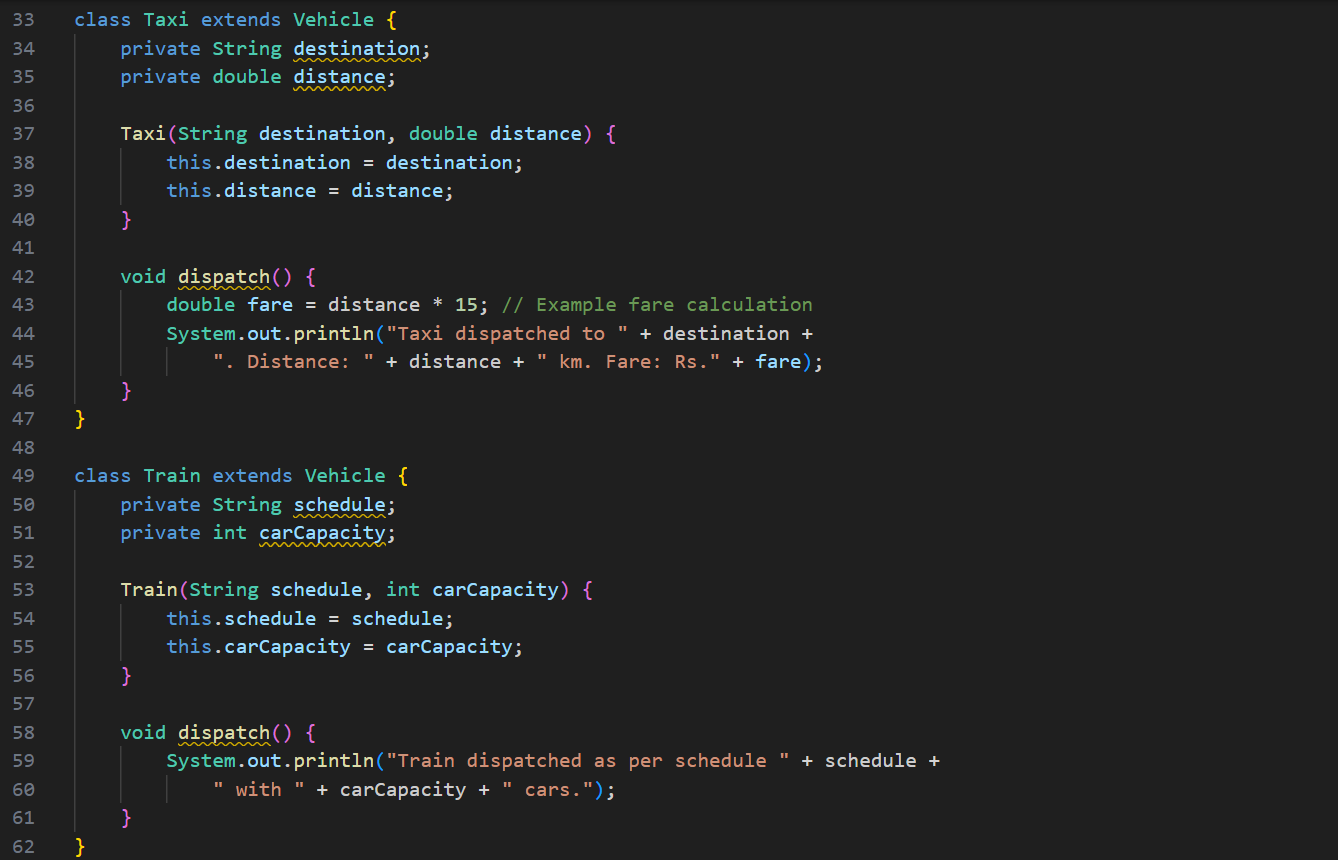
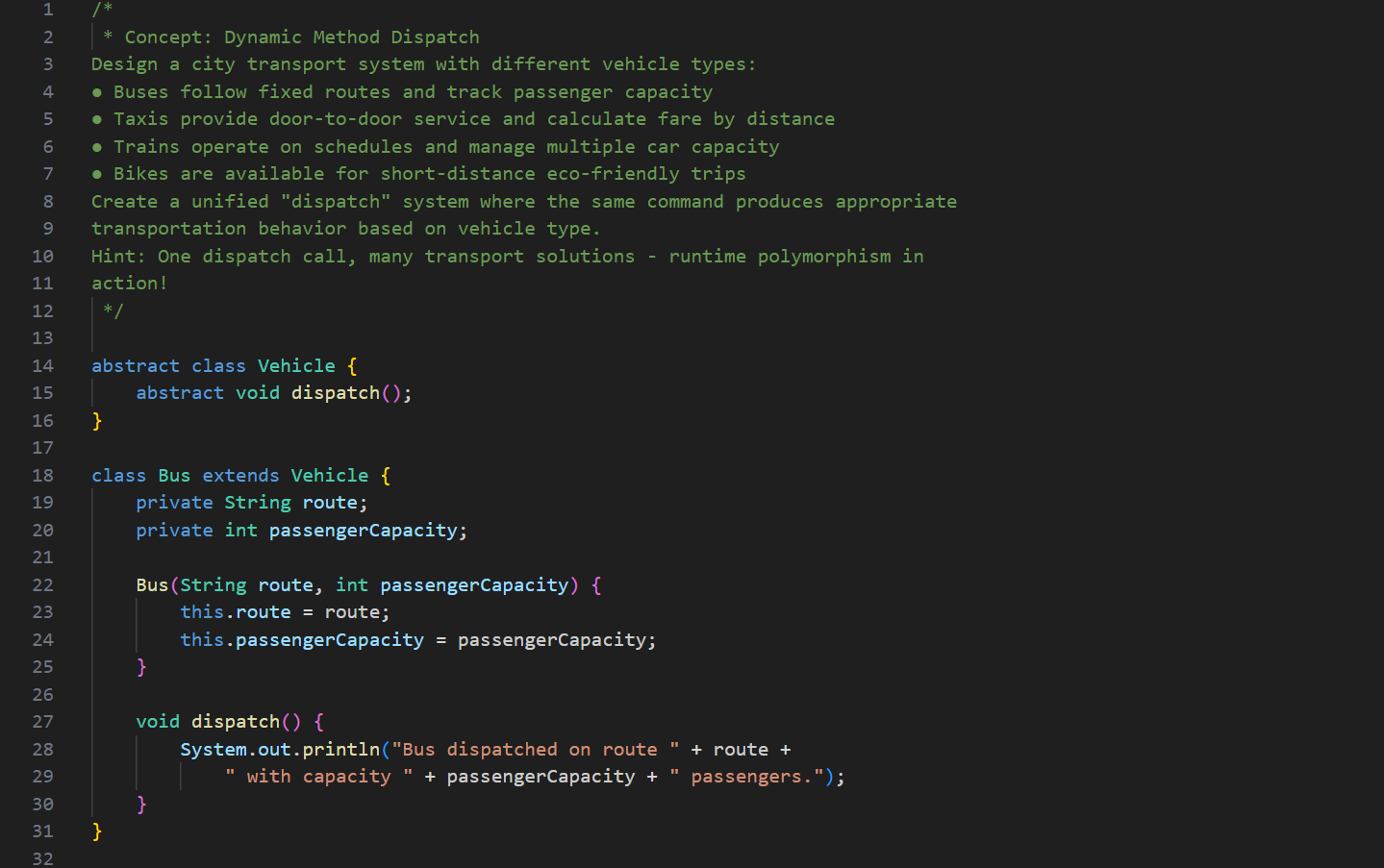
**Create a unified "dispatch" system where the same command produces appropriate**

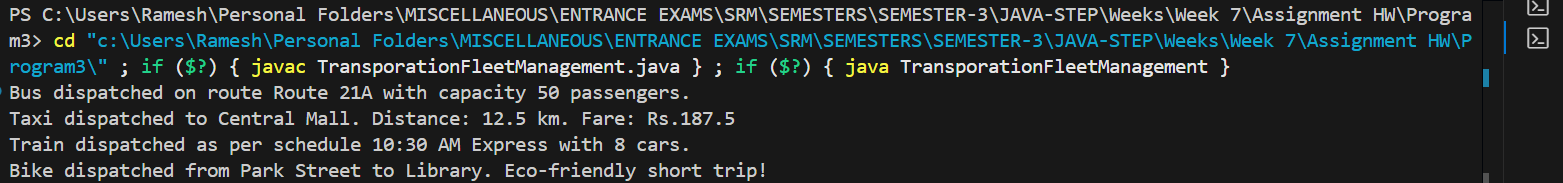
**transportation behavior based on vehicle type.**

**Hint: One dispatch call, many transport solutions - runtime polymorphism in**

**action!**

**PROGRAM🡪**

****

**OUTPUT🡪**

**PROBLEM 4: Hospital Management System**

**Concept: Upcasting**

**Build a hospital system managing different types of medical staff:**

**● Doctors can diagnose patients, prescribe medicine, and perform surgeries**

**● Nurses can administer medicine, monitor patients, and assist procedures**

**● Technicians can operate equipment, run tests, and maintain instruments**

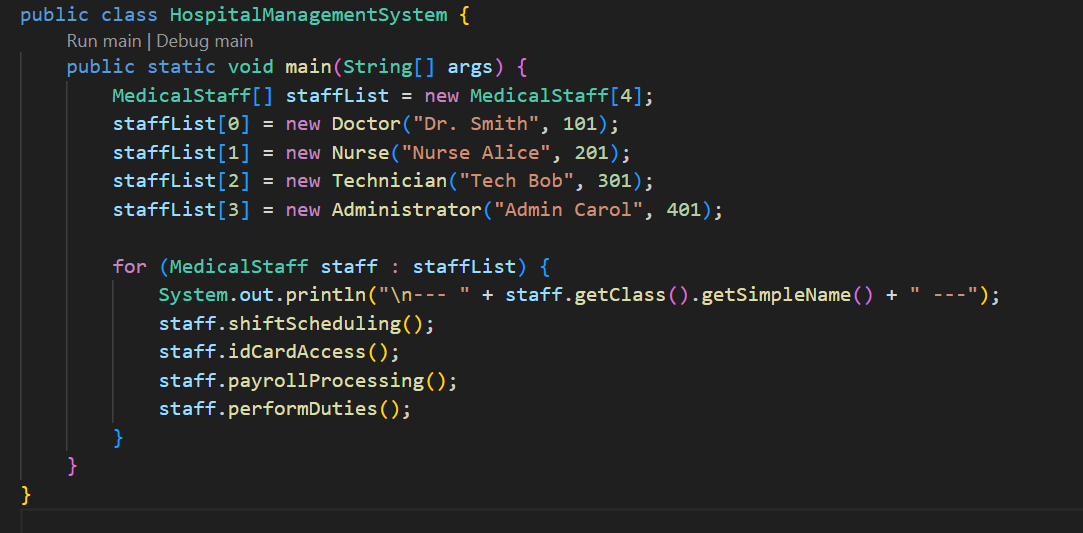
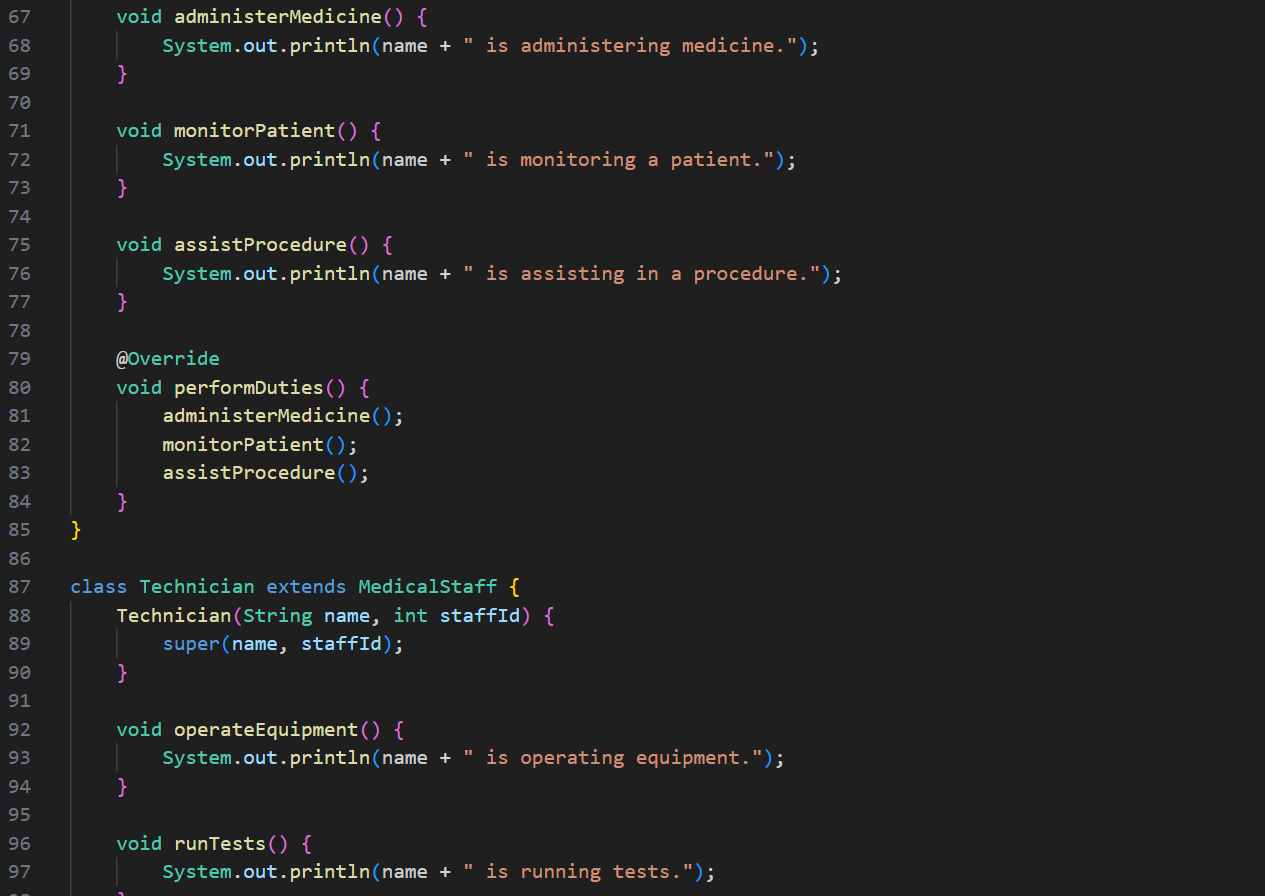
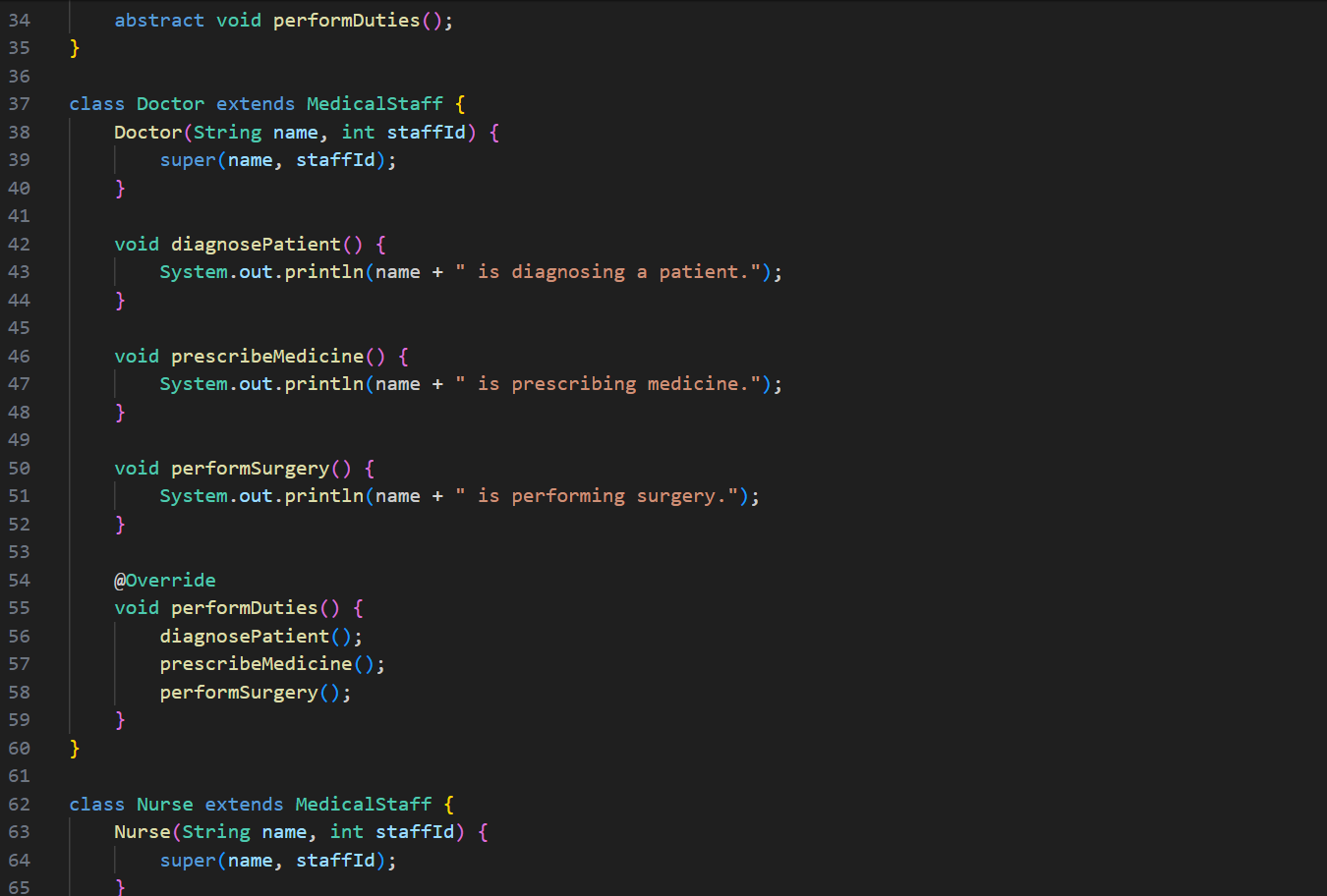
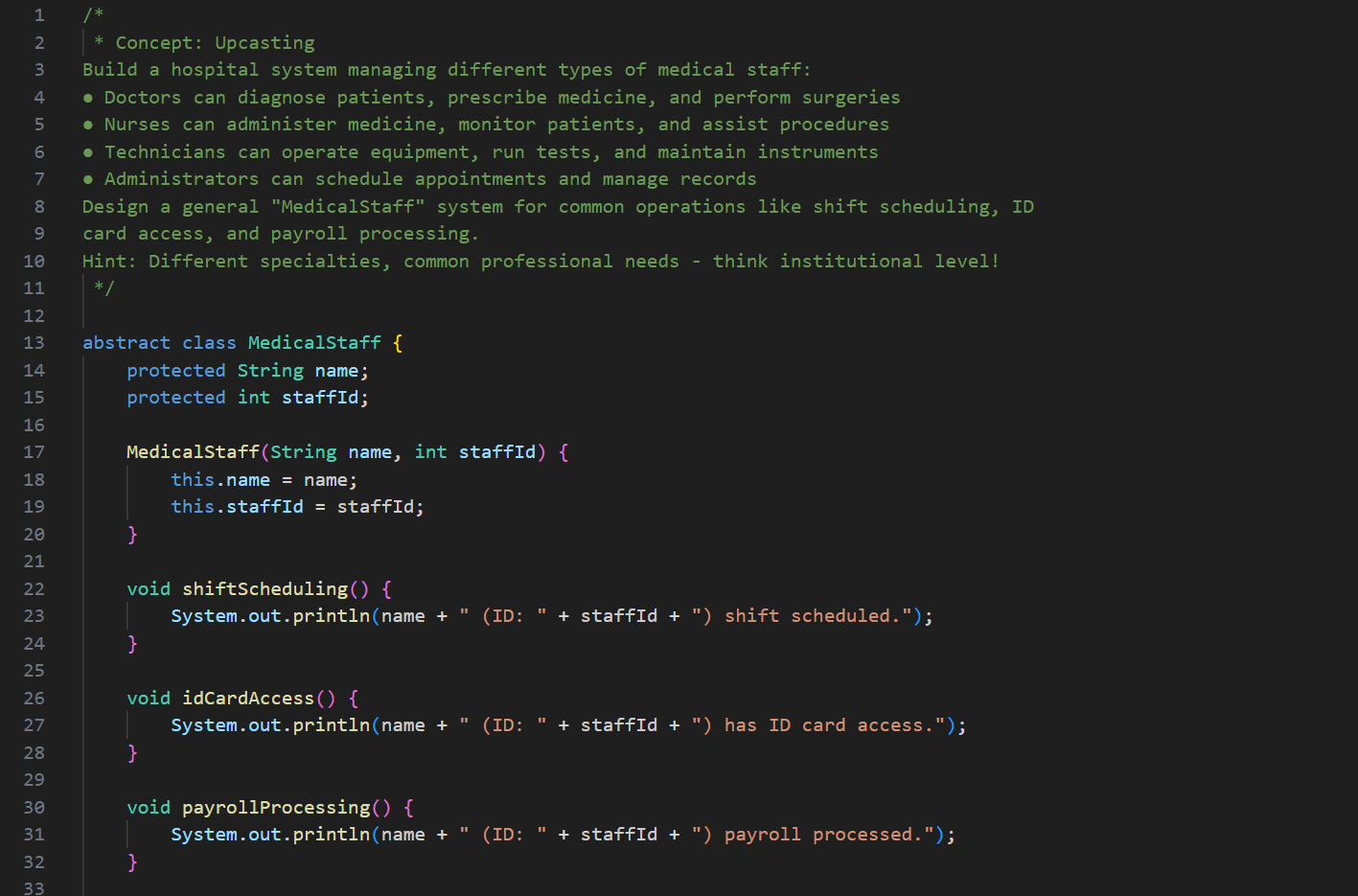
**● Administrators can schedule appointments and manage records**

**Design a general "MedicalStaff" system for common operations like shift scheduling, ID**

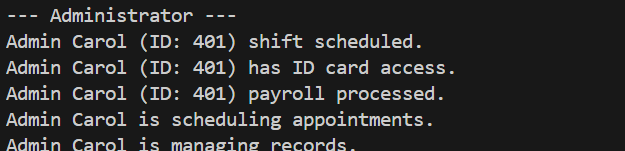
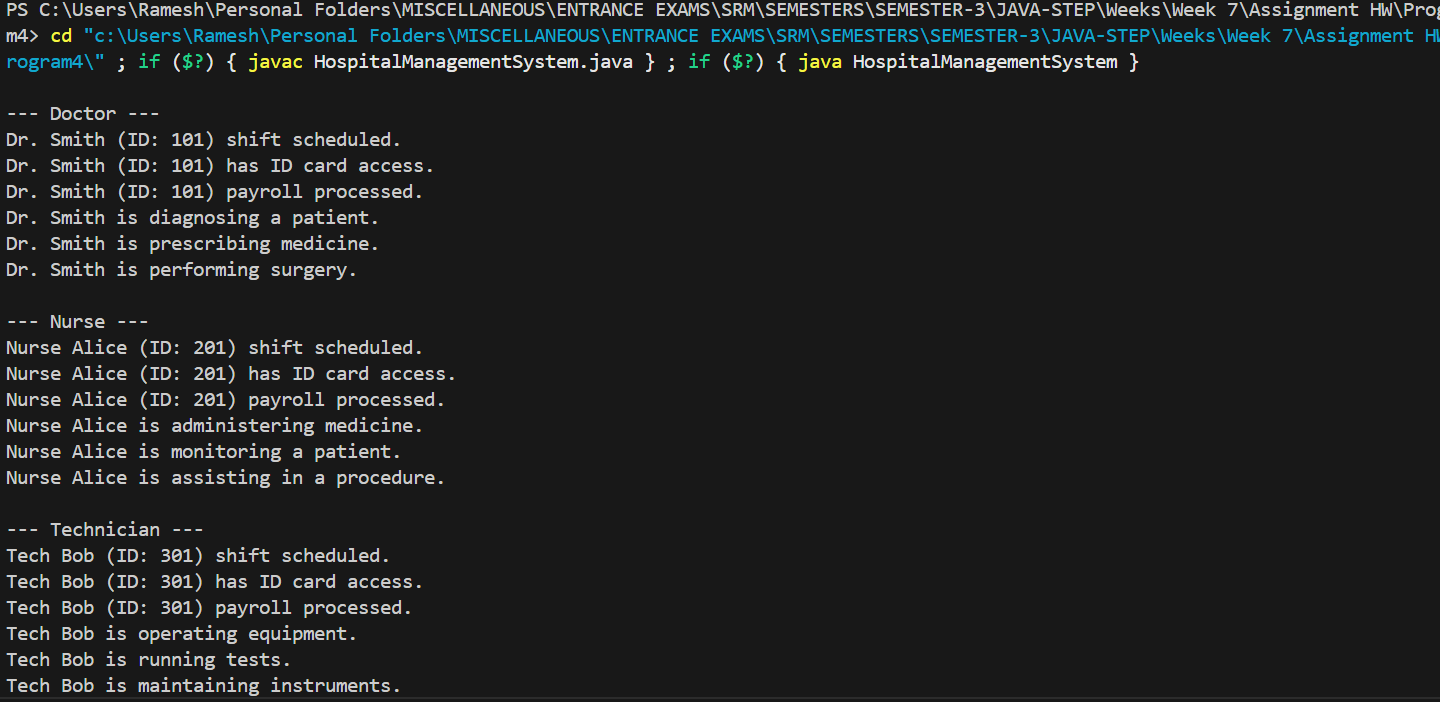
**card access, and payroll processing.**

**Hint: Different specialties, common professional needs - think institutional level!**

**PROGRAM🡪**

****

**OUTPUT🡪**

****

**PROBLEM 5: Digital Art Gallery**

**Concept: Downcasting**

**Create an art gallery system handling different artwork types:**

**● Paintings have brush techniques, color palettes, and frame specifications**

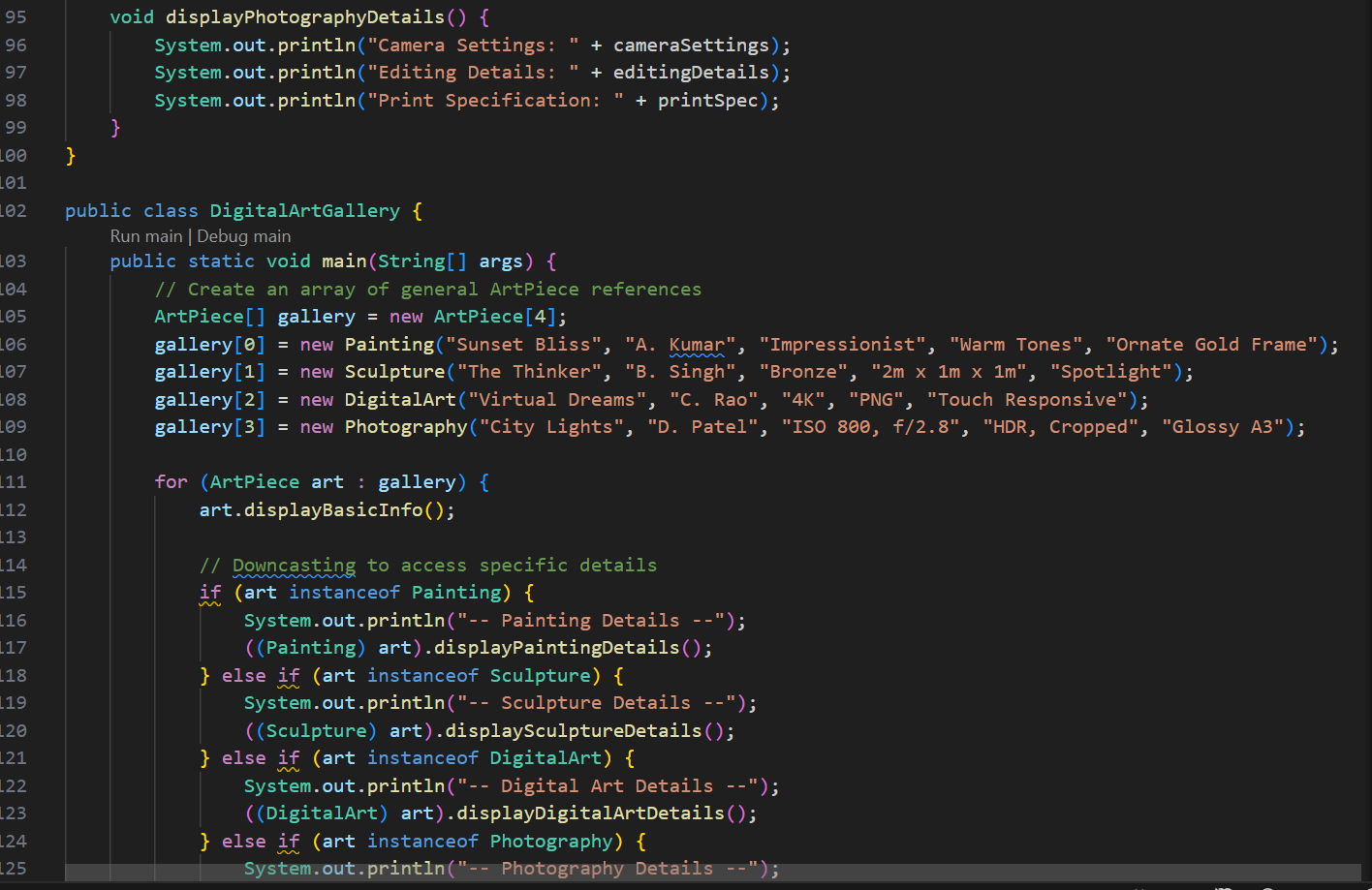
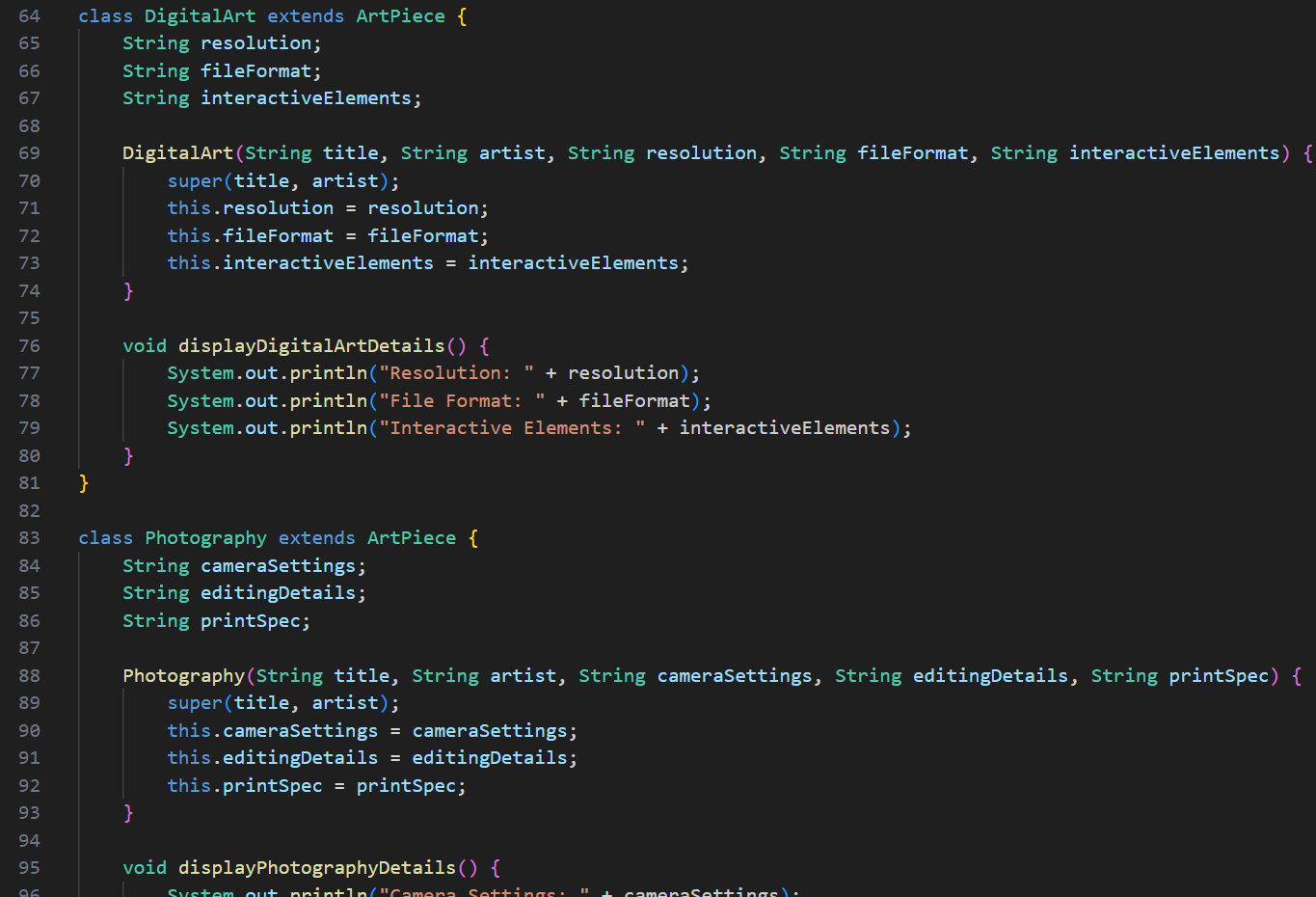
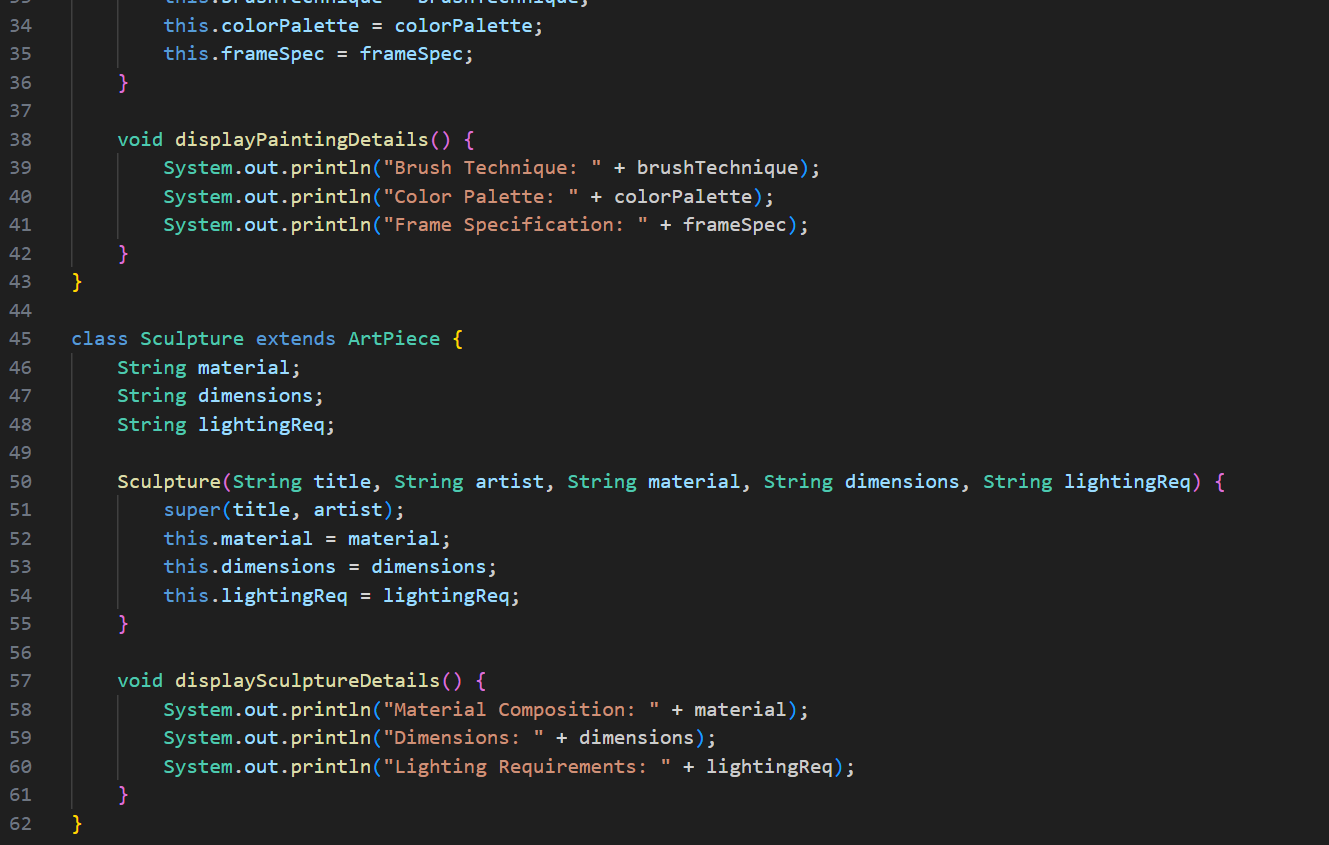
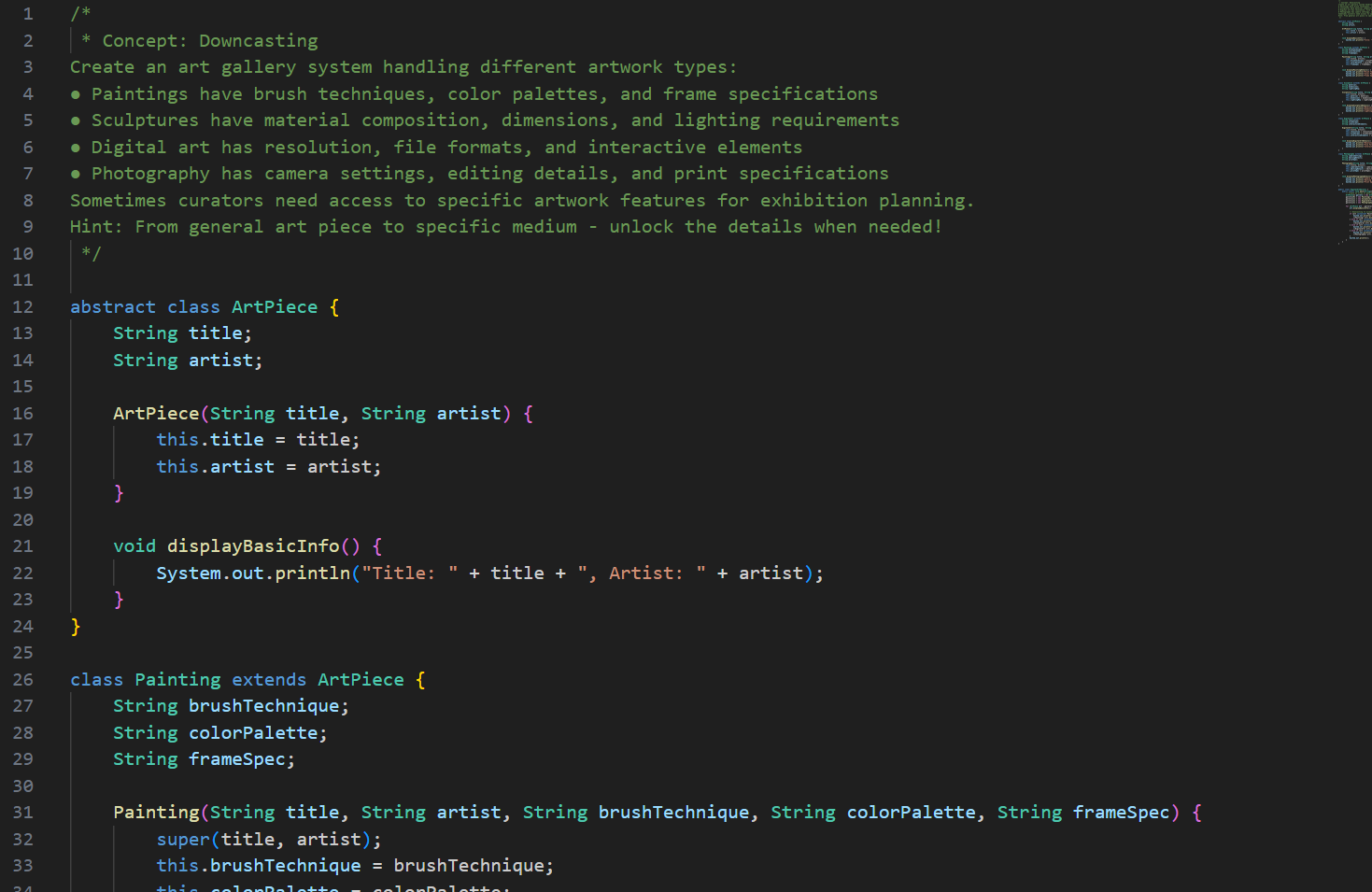
**● Sculptures have material composition, dimensions, and lighting requirements**

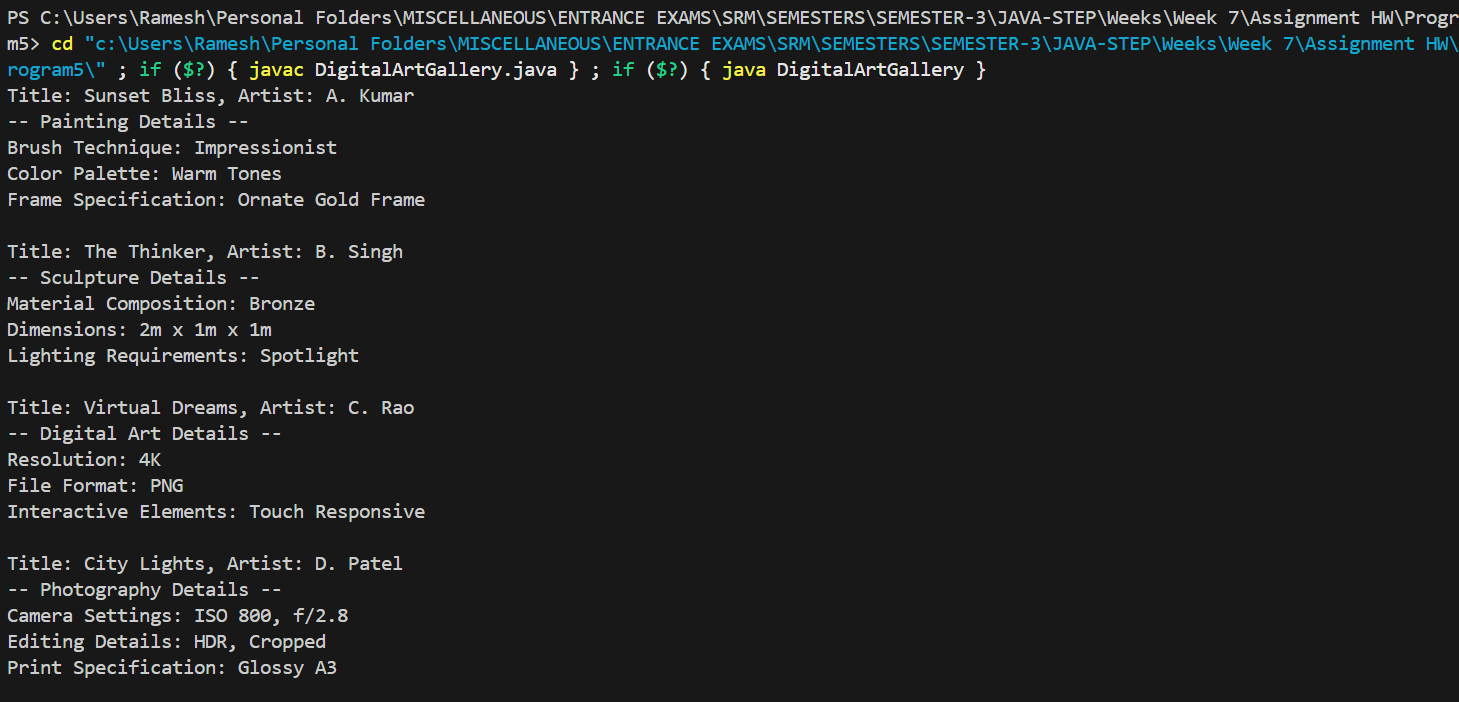
**● Digital art has resolution, file formats, and interactive elements**

**● Photography has camera settings, editing details, and print specifications**

**Sometimes curators need access to specific artwork features for exhibition planning.**

**Hint: From general art piece to specific medium - unlock the details when needed!**

**PROGRAM🡪**

**OUTPUT🡪**

**PROBLEM 6: Smart Home Automation**

**Concept: Safe Downcasting with instanceof**

**Design a home automation system controlling various smart devices:**

**● Smart TVs manage channels, volume, and streaming apps**

**● Smart thermostats control temperature, humidity, and energy saving modes**

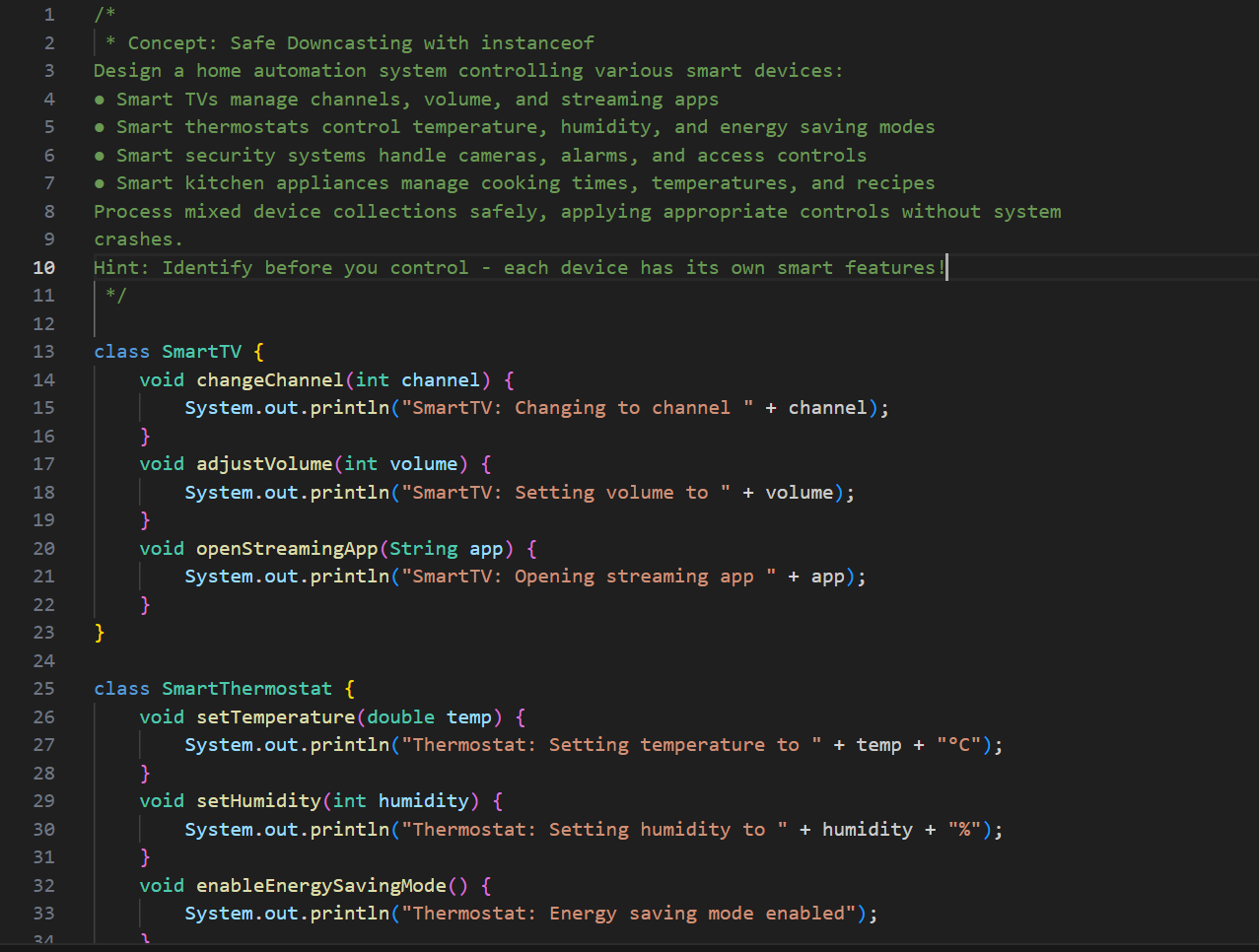
**● Smart security systems handle cameras, alarms, and access controls**

**● Smart kitchen appliances manage cooking times, temperatures, and recipes**

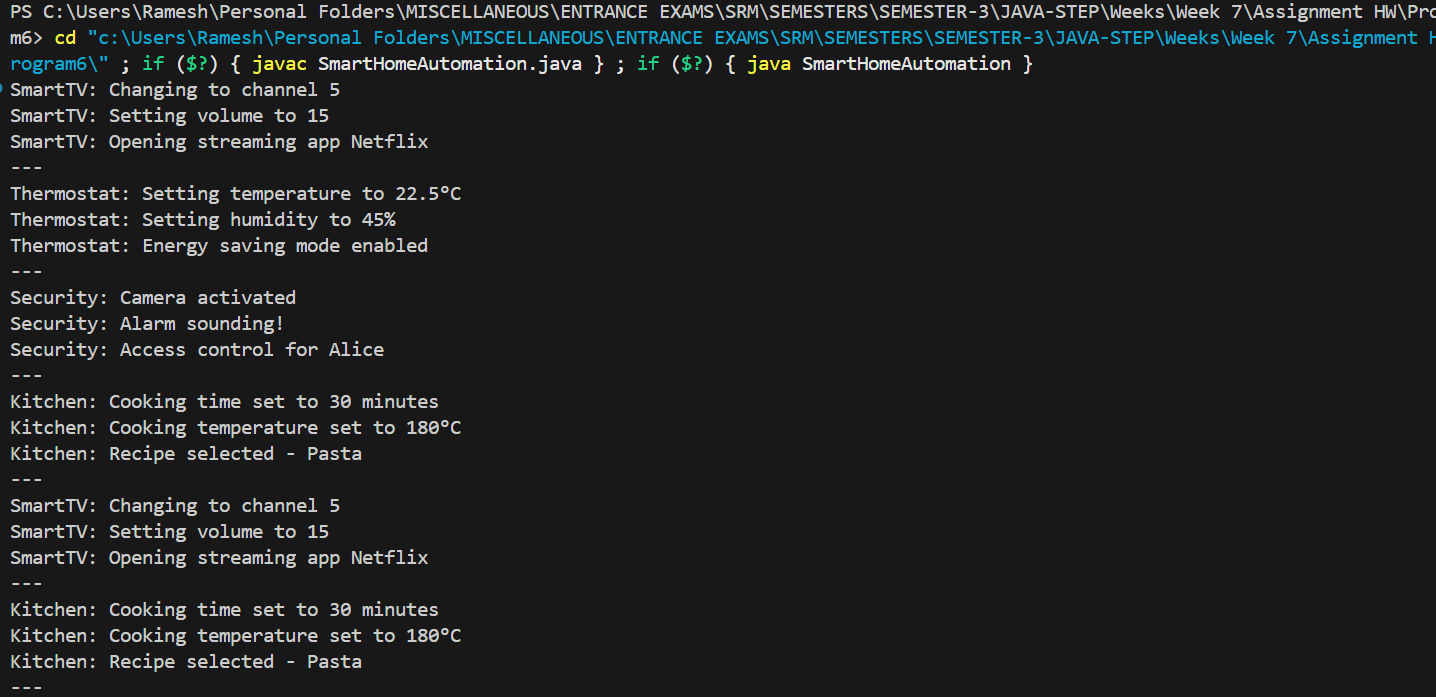
**Process mixed device collections safely, applying appropriate controls without system**

**crashes.**

**Hint: Identify before you control - each device has its own smart features!**

**PROGRAM🡪**

**OUTPUT🡪**

****