Class, Object and Sequence Diagrams

Sample Problem 1: School Results Application

Class Diagram

The class diagram represents the structure of a school results application where students have subjects, and their scores are calculated for grades.

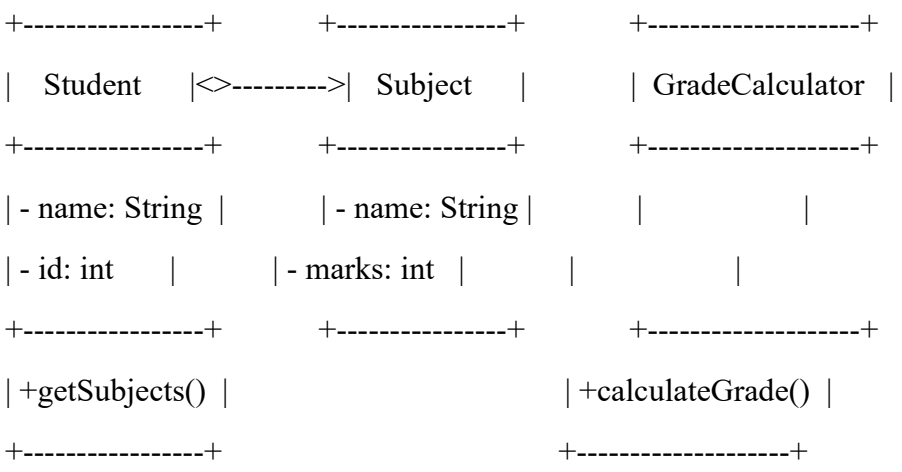
Classes:

Student, Subject, GradeCalculator

Relationships:

- A Student has multiple Subject entries (Aggregation).

- GradeCalculator computes the results for a Student.



Object Diagram

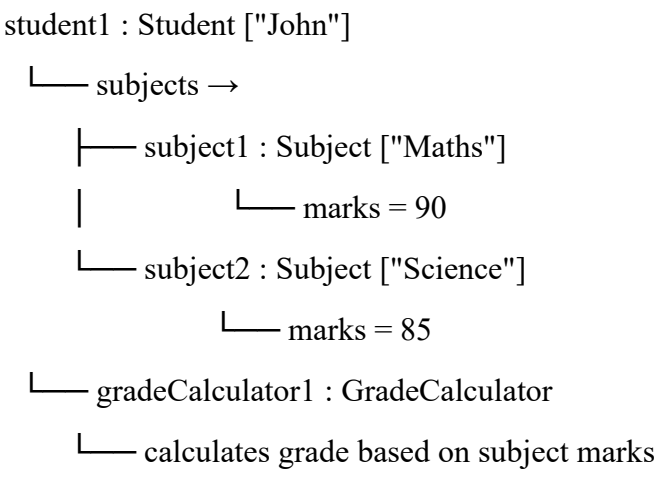
An object diagram provides a snapshot of the Student and their Subject objects at a particular point.

Example:

Student: John

Subjects: Maths, Science

Marks: 90, 85



Sequence Diagram

The sequence diagram shows how objects interact to calculate grades.

Scenario:

A student requests their grade based on marks in subjects.

Actors:

Student, GradeCalculator

Student → GradeCalculator : requestGrade(subjects)

GradeCalculator → Subject[] : get marks

GradeCalculator → GradeCalculator : calculateGrade()

GradeCalculator → Student : return grade/result

Sample Problem 2: Grocery Store Bill Generation Application

Class Diagram

The class diagram models the system where a customer buys products, and the bill is generated.

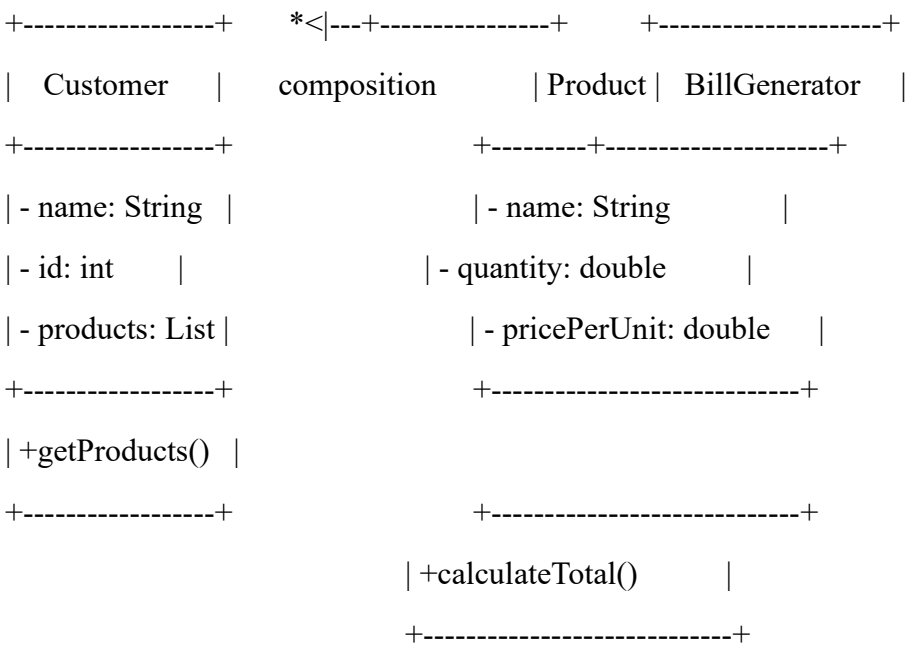
Classes:

Customer, Product, BillGenerator

Relationships:

- A Customer can purchase multiple Product items (Composition).

- BillGenerator computes the total for the Customer.



Object Diagram

An object diagram shows the details of a Customer and the Product objects they have purchased.

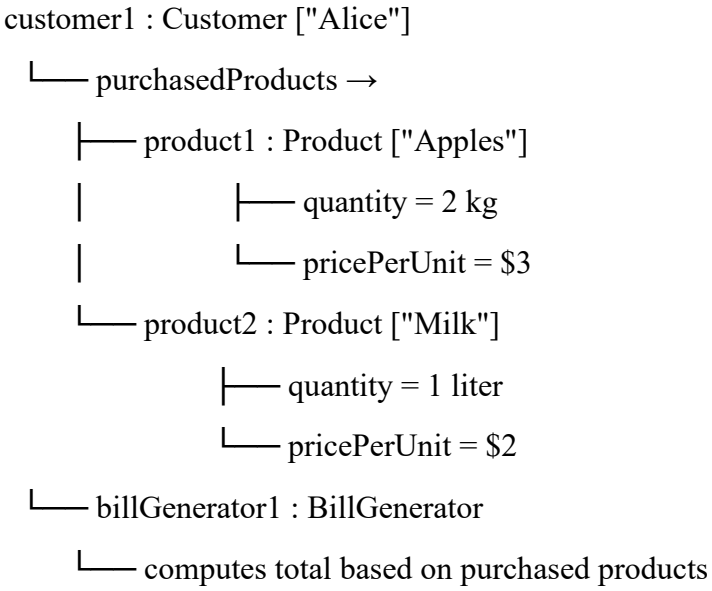
Example:

Customer: Alice

Products:

Apples (2 kg at $3 per kg)

Milk (1 liter at $2 per liter)



Sequence Diagram

The sequence diagram shows the process of bill generation for a customer.

Scenario:

A customer checks out at the grocery store, and the total bill is generated.

Actors:

Customer, BillGenerator

Customer → BillGenerator : checkout(products)

BillGenerator → Product[] : retrieve price and quantity

BillGenerator → BillGenerator : calculateTotal()

BillGenerator → Customer : return total amount