

## **Assignment 1: Student Encapsulation**

Create a Student class:

- Private variables: id, name
- Public getters & setters
- Display details using getter methods.

**Concepts:** Data hiding, getter/setter

## **Assignment 2: Bank Account**

Create BankAccount class:

- Private: accountNumber, balance
- Setter validates balance (no negative value)
- Getter returns balance.

**Concepts:** Validation logic

## **Assignment 3: Employee Salary**

Create Employee class:

- Private: empId, salary
- Setter restricts salary > 0
- Method displaySalary()

**Concepts:** Controlled access

## **Assignment 4: Product Price**

Create Product class:

- Private: price
- Setter allows price only between 100 – 100000

**Concepts:** Business rule encapsulation

## **Assignment 5: Login Credentials**

Create User class:

- Private: email, password
- Password setter validates length  $\geq 8$

**Concepts:** Security with encapsulation

## **Assignment 6: Customer Profile**

Create Customer class:

- Private: name, age
- Setter restricts age  $\geq 18$
- Getter returns formatted data

**Concepts:** Validation inside setters

## **Assignment 7: Mobile Phone**

Create Mobile class:

- Private: brand, price
- Getter returns price with GST

**Concepts:** Logic in getter

## **Assignment 8: ATM System**

Create ATM Account class:

- Private: pin, balance
- setPin() validates 4 digits
- withdraw(amount) checks balance

**Concepts:** Secure operations

### **Assignment 9: College Admission**

Create Admission class:

- Private: marks
- Setter assigns grade internally

**Concepts:** Internal processing

### **Assignment 10: Insurance Policy**

Create Policy class:

- Private: policyId, premium
- Premium calculated internally based on age

**Concepts:** Business rule hiding