

10 **scenario-based coding problems** focused on **if-else** statements in Python, at a **Medium level**. Each problem includes a problem statement, sample input, and expected output.

---

### 1. Movie Ticket Booking System

**Problem:** You need to implement a ticket pricing system for a cinema. The pricing scheme is as follows:

- Children ( $\text{age} < 12$ ) get a 50% discount on the ticket price.
- Senior citizens ( $\text{age} \geq 60$ ) get a 30% discount.
- Adults ( $12 \leq \text{age} < 60$ ) pay the full ticket price. Also, if the movie is a 3D movie, an additional charge of \$5 is added to the ticket price.

**Input:** age, ticket\_price, is\_3d\_movie

**Example:**

Input: age = 65, ticket\_price = 12, is\_3d\_movie = True

Output: Total price = \$13.4

Input: age = 10, ticket\_price = 10, is\_3d\_movie = False

Output: Total price = \$5.0

---

### 2. Credit Card Fraud Detection

**Problem:** Develop a fraud detection system for a bank. The system should flag a transaction as "suspicious" if:

- The transaction amount exceeds 70% of the account's total balance.
- The transaction happens after 10 PM (22:00) or before 6 AM (06:00).

**Input:** account\_balance, transaction\_amount, transaction\_time (24-hour format)

**Example:**

Input: account\_balance = 5000, transaction\_amount = 4000, transaction\_time = "23:30"

Output: Transaction flagged as suspicious

Input: account\_balance = 10000, transaction\_amount = 500, transaction\_time = "14:45"

Output: Transaction is safe

---

### 3. Online Shopping Discount System

**Problem:** Design a discount system for an online shopping platform. The discount rules are:

- If the purchase is above \$200, the customer gets a 15% discount.
- For purchases between \$100 and \$200, they get a 10% discount.
- If the customer is a premium member, they get an additional 5% discount on top of the initial discount.

**Input:** purchase\_amount, is\_premium\_member

**Example:**

Input: purchase\_amount = 250, is\_premium\_member = True

Output: Total amount after discount = \$195.75

Input: purchase\_amount = 180, is\_premium\_member = False

Output: Total amount after discount = \$162.0

---

### 4. Car Insurance Premium Calculation

**Problem:** Write a program to calculate car insurance premiums. The premium depends on:

- The car's age:
  - Less than 5 years: base premium is \$500.
  - 5 to 10 years: base premium is \$800.
  - More than 10 years: base premium is \$1200.
- If the car owner is under 25 years old, there's an additional 20% premium.
- If the car owner has had no accidents in the last year, they get a 10% discount.

**Input:** car\_age, owner\_age, no\_accidents

**Example:**

Input: car\_age = 6, owner\_age = 24, no\_accidents = True

Output: Premium = \$720.0

Input: car\_age = 12, owner\_age = 30, no\_accidents = False

Output: Premium = \$1200.0

## 5. Electricity Bill Calculation

**Problem:** Create a program to calculate electricity bills. The billing rules are:

- The first 100 units cost \$0.50 per unit.
- The next 100 units (101 to 200) cost \$0.75 per unit.
- After 200 units, the cost is \$1.20 per unit. If the total bill exceeds \$300, a surcharge of 15% is added.

**Input:** units\_consumed

**Example:**

Input: units\_consumed = 250

Output: Total bill = \$240.0

Input: units\_consumed = 350

Output: Total bill = \$414.0

## 6. Shipping Fee Calculation

**Problem:** Write a program to calculate shipping fees. The rules are:

- If the destination is international, a flat fee of \$30 is added.
- If the package weight exceeds 10 kg, an additional charge of \$5 per kg over 10 kg is added.
- Domestic shipments have no additional charges if the weight is under 10 kg.

**Input:** destination, weight

**Example:**

Input: destination = "international", weight = 12

Output: Shipping fee = \$40.0

Input: destination = "domestic", weight = 9

Output: Shipping fee = \$0.0

## 7. Graduation Eligibility Check

**Problem:** Write a program that checks if a student is eligible for graduation. The conditions are:

- A student must have a GPA of 2.5 or higher.
- The student must have completed at least 120 credits.

- If the student has a GPA below 3.0, they must also pass a final comprehensive exam.

**Input:** gpa, credits, final\_exam\_passed

**Example:**

Input: gpa = 2.9, credits = 130, final\_exam\_passed = True

Output: Eligible for graduation

Input: gpa = 2.4, credits = 125, final\_exam\_passed = False

Output: Not eligible for graduation

---

## 8. Tax Filing Status Determination

**Problem:** Write a program that determines a person's tax filing status. The rules are:

- If the person is married, their filing status is "Married".
- If the person is single, but has dependents, their status is "Head of Household".
- If the person is single and has no dependents, their status is "Single".

**Input:** married, has\_dependents

**Example:**

Input: married = False, has\_dependents = True

Output: Filing status = Head of Household

Input: married = True, has\_dependents = False

Output: Filing status = Married

---

## 9. Loan Repayment Plan

**Problem:** Design a program to determine a person's loan repayment plan. The conditions are:

- If the loan amount is less than \$50,000, they must repay it within 5 years.
- If the loan amount is between \$50,000 and \$100,000, they must repay it within 10 years.
- If the loan amount is greater than \$100,000, they must repay it within 15 years.
- Additionally, if the person's annual income is less than \$30,000, they are allowed an extra 2 years to repay.

**Input:** loan\_amount, annual\_income

**Example:**

Input: loan\_amount = 75000, annual\_income = 28000

Output: Repayment term = 12 years

Input: loan\_amount = 45000, annual\_income = 50000

Output: Repayment term = 5 years

---

## 10. Ticket Upgrade Decision

**Problem:** Create a program that determines if a person can upgrade to a higher class on a flight. The conditions are:

- A person can upgrade if they have enough points (at least 1000).
- If the flight is overbooked, they need 20% fewer points (800).
- If they are a frequent flyer, they need 10% fewer points (900), and 30% fewer if the flight is also overbooked (700).

**Input:** points, is\_overbooked, is\_frequent\_flyer

**Example:**

Input: points = 850, is\_overbooked = True, is\_frequent\_flyer = True  
Output: Eligible for upgrade

Input: points = 950, is\_overbooked = False, is\_frequent\_flyer = False  
Output: Not eligible for upgrade

---

**Key Learning Points:**

- These problems involve complex conditions, combining multiple if-else structures.
- The aim is to encourage students to use nested and multi-conditional checks, logical operators, and data validation