

# Java\_Assignment-06

## 🚺 Enums - Online Food Ordering System 🍕

**Scenario:** Build a **food ordering system** where users can select their food category using an Enum.

#### Steps:

- Create an Enum FoodCategory With Values: STARTER, MAIN\_COURSE, DESSERT, BEVERAGE.
- Create a class FoodOrder with a method selectCategory(FoodCategory category).
- Based on the category, display sample food items.
- Take user input to select a category and display available items.

#### Example Output:

#### Select Food Category:

- 1. STARTER
- 2. MAIN\_COURSE
- 3. DESSERT
- 4. BEVERAGE

Enter choice: 2

#### Available MAIN\_COURSE:

- Biryani
- Pasta
- Fried Rice

## 🔼 Scanner - ATM Transaction 👗

**Scenario:** Simulate an **ATM system** where users can:

Check Balance

Java\_Assignment-06

- **V** Deposit Money
- Withdraw Money

#### Steps:

- Use scanner to take user input for:
  - Initial balance
  - Deposit amount
  - Withdrawal amount (Check if enough balance is available)
- Display the updated balance after every transaction.

#### Example Output:

Enter Initial Balance: 5000

Choose Option:

- 1. Check Balance
- 2. Deposit
- 3. Withdraw

Enter choice: 2

**Enter Deposit Amount: 2000** 

New Balance: 7000

### DateTime - Movie Ticket Booking 📧

**Scenario:** Create a **Movie Ticket Booking System** where users can book tickets for a future date.

#### Steps:

- Display today's date using LocalDate.now().
- Ask the user to enter a booking date (yyyy-MM-dd format).
- Check if the date is valid & in the future.
- Display booking confirmation or an error message if the date is invalid.

#### Example Output:

Java\_Assignment-06 2

Today's Date: 2025-03-19

Enter Booking Date (yyyy-MM-dd): 2025-03-22

▼ Booking Confirmed for 2025-03-22!

If the user enters a past date:

X Invalid Date! Please choose a future date.

### 💶 Exception Handling - Student Grade Calculator 📚

**Scenario:** Take user input for **marks** and calculate the grade. Handle invalid inputs and division errors.

#### Steps:

- Take total marks & obtained marks using Scanner.
- Try-Catch for input mismatch (if the user enters invalid data).
- Throw an exception if obtained marks are greater than total marks.
- Finally block to display "Calculation Complete."

#### Example Output:

Enter Total Marks: 100 Enter Obtained Marks: 110

X Error: Obtained marks cannot be greater than total marks!

## 亙 Custom Exception - Bank Loan Eligibility 🏫

**Scenario:** Check whether a user is **eligible for a loan** based on salary & credit score.

#### Steps:

- Create a **Custom Exception** LoanEligibilityException.
- Take Salary and Credit Score as user input.
- If Salary < 30000 or Credit Score < 700, throw LoanEligibilityException.</li>
- Otherwise, approve the loan.

Java\_Assignment-06

#### Example Output:

Enter Salary: 25000 Enter Credit Score: 650

X LoanEligibilityException: You are not eligible for a loan!

### 🚀 Submission Guidelines:

- ✓ Implement each program separately & test the output.
- ✓ Use meaningful variable & method names.
- √ Handle edge cases properly.

Java\_Assignment-06