Java Question Bank

Question 1.)

You are developing a Java program for a simple library management system using an ArrayList to manage books in the library. Each book should have attributes including the title, author, and ISBN number. Implement a Book class with methods to perform the following actions:

1. Allow users to add a new book to the library.
2. Allow users to remove a book from the library by providing its ISBN number. 3. Enable users to search for a book by title and display its details.
4. Display the list of all books currently in the library.

Question 2.)

You are developing a Java program to manage an online shopping cart. Implement code to handle the following built-in exceptions:

1. ArrayIndexOutOfBoundsException
2. NumberFormatException
3. ArithmeticException

Question 3.)

You are developing a Java application for managing vehicles in a rental service. Create a class named Vehicle with attributes for make, model, and year. Implement a constructor that initializes these fields. Next, design a subclass named Car with an additional attribute for numberOfDoors. Provide two constructors for the Car class: one that accepts all fields including make, model, year, and numberOfDoors, and another that accepts only make and model, chaining to the superclass constructor. Validate the input in each constructor and provide appropriate getter and setter methods to access and modify the attributes of the Vehicle class. Ensure proper usage of inheritance principles.

Question 4.)

Create a Java program to manage an employee database for a company using an ArrayList. Each employee should have attributes such as employee name, employee ID, and department. Implement an Employee class with methods to perform the following operations:

1. Add a new employee to the database.
2. Update an employee's department using their employee ID.

3. Remove an employee from the database using their employee ID.
4. Display the list of all employees along with their details.

Question 5.)

You are developing a Java program to manage user inputs in a ticket booking system. Implement code to handle the following built-in exceptions:

1. InputMismatchException3
2. IllegalArgumentException
3. IndexOutOfBoundsException

Question 6.)

You are creating a Java program to represent different types of employees in a company. Implement a class named Employee with attributes for name, employeeID, and department. Provide a constructor to initialize these fields. Next, create a subclass named Manager with an additional attribute for numberOfTeams managed. The Manager class should have two constructors: one that accepts all fields, and another that accepts only name and employeeID, chaining to the superclass constructor. Include methods to validate input data and implement getter and setter methods for the attributes in the Employee class. Demonstrate the inheritance hierarchy by creating instances of both classes and displaying their details.

Question 7.)

Develop a Java program for a simple banking system using an ArrayList to manage bank accounts. Each bank account should have attributes including the name of the account holder, account number, and initial balance. Implement the BankAccount class with methods to perform the following actions: (Using array list)

● Allow users to deposit money into their account.
● Allow users to withdraw money from their account if they have sufficient balance. ● Display the current balance of the account.
● Enable users to transfer money from one account to another, provided they have sufficient balance. **Question**

8.)

You're developing error handling for a Java program that manages payments in an e-commerce application. Write code to handle the following three, specific payment gateway errors:

● Timeout error: Implement exception handling to catch situations where the payment gateway response times out.
● Invalid card details: Handle exceptions arising from attempts to process payments with invalid card information.

● Insufficient funds: Implement exception handling to manage cases where users attempt to make payments without sufficient funds in their account.

Question 9.)

You're developing a Java program to manage individuals within an educational institution. Create a class named "Person" with attributes for the name and age of an individual. Implement a constructor that accepts both fields as arguments. Next, design a subclass named "Student" with an additional attribute for the grade level. Provide two constructors for the Student class: one that accepts all fields including name, age, and grade level, and another that accepts only the name and age, chaining to the superclass constructor. Ensure that your program demonstrates proper inheritance principles, and validate the constructors to ensure valid data is provided during object instantiation,also implement appropriate getter and setter methods to access and modify the attributes of Person class.

Question 10)

Create a class Student with:

● int studentId, String name, double grade.
● A constructor to initialize these fields.
● Methods:
  ○ updateGrade(double newGrade): Updates the grade, but should not accept negative values (handle using exception handling).
  ○ display(): Prints student details.

Question 11)

Write a Java program that reads a string from the user and attempts to convert it to an integer using Integer.parseInt(). If the input is not a valid integer, handle the NumberFormatException. Additionally, handle NullPointerException if the input is null. Use a finally block to print "Conversion attempt completed."

Question 12)

Patient Management System

You are managing a patient database for a hospital. Each patient has a unique patient ID, a name, a diagnosis, and the number of days admitted. You need to implement a solution using appropriate Java collection classes to efficiently perform the following operations: a. Add a new patient to the database.
 b. Remove a patient from the database.
 c. Find all patients with a specific diagnosis.
 d. Find all patients admitted for more than a given number of days.