Here’s a **detailed practice material guide** for each segment of your online MCQ test preparation. These resources and example questions will help solidify your understanding of key topics.

1. Object-Oriented Programming (OOP)

Key Topics:

- **Core Principles**: Encapsulation, Inheritance, Polymorphism, Abstraction.
- **Access Modifiers**: private, public, protected.
- **Keywords**: `this`, `super`, `final`, `static`.
- **Class Relationships**: Aggregation, Composition, Association.

Practice Questions:

1. What is the difference between `Abstract Class` and `Interface`?

Identify which design principle is violated:

```
 class Rectangle {
    int width, height;
    int calculateArea() { return width * height; }
}
class Square extends Rectangle {
    void setWidth(int w) { width = w; height = w; }
}
```

2. **Answer**: Liskov Substitution Principle (LSP) violation.

Resources:

- [GeeksforGeeks - OOP Concepts]
- [JavaTPoint - OOP Basics]

---

2. Database

Key Topics:

- **SQL Queries**: `SELECT`, `JOIN`, `GROUP BY`, `HAVING`, Subqueries.
- **Normalization**: First, Second, and Third Normal Forms.
- **Transactions**: ACID properties and examples.
- **Indexes**: How they work and their impact.

Practice Questions:

Write a query to fetch the second-highest salary from an `employees` table.

```
 SELECT MAX(salary) FROM employees WHERE salary < (SELECT MAX(salary) FROM employees);
```

1.

Normalize the following relation:

 STUDENT(ID, Name, Subject, SubjectMarks)

2. Into:

    - **1NF**: Each field should contain only atomic values.
    - **2NF**: No partial dependency on composite keys.
    - **3NF**: No transitive dependency.

Resources:

- SQL Tutorial - W3Schools
- Normalization

---

3. Data Structure & Algorithm

Key Topics:

- **Sorting**: Quick Sort, Merge Sort.
- **Searching**: Binary Search, Linear Search.
- **Data Structures**: Arrays, Linked Lists, Stacks, Queues, Trees, Graphs.
- **Complexity**: Big-O for common operations.

Practice Questions:

1. Write a function to reverse a linked list.
2. Find the shortest path in an unweighted graph using BFS.

Resources:

- Big O Notation Cheat Sheet
- Sorting Algorithms Visualization
- DSA Practice on LeetCode

---

4. Problem-Solving

Key Topics:

- **Logical Thinking**: Breaking problems into smaller parts.
- **Dynamic Programming**: Fibonacci, Knapsack problem.
- **Greedy Algorithms**: Activity Selection Problem.

Practice Questions:
Solve this:

```
 def find_sum(n):
    if n == 0:
        return 0
    return n + find_sum(n - 1)
print(find_sum(5))
```

1. **Answer**: 15.

2. Implement a function to check if two strings are anagrams.

Resources:

- [Dynamic Programming Guide]

---

5. Output Tracing

Key Topics:

- **Debugging**: Understanding loops, conditionals, recursion.
- **Code Flow**: Identifying logical errors.

Practice Questions:
Trace the output of:

```
 let x = 5;
for (let i = 0; i < x; i++) {
    x -= 1;
    console.log(i);
}
```

1. **Answer**: `0 1 2`.

Recursive function trace:

```
def factorial(n):
  if n == 0:
     return 1
  return n * factorial(n - 1)

print(factorial(4))
```

2. **Answer**: `24`.

Resources:

- Debugging in Python
- Debugging Tools for JavaScript

6. Analytical Ability

Key Topics:

- **Logic Puzzles**: Sudoku, pattern matching.
- **Mathematical Reasoning**: Percentage, ratio, probability.

Practice Questions:

1. A train running at 60 km/h crosses a pole in 9 seconds. What is the length of the train?
   Answer: `150 meters`.

Solve:

If 3 apples cost $12, how much do 5 apples cost?

2. **Answer**: `$20`.

Resources:

- Analytical Reasoning Puzzles

7. Software Development Life Cycle (SDLC)

Key Topics:

- **Models**: Waterfall, Agile, Spiral.
- **Phases**:
  - Requirement Analysis.
  - Design: ER Diagrams, Flowcharts.
  - Implementation: Code development.
  - Testing: Unit, Integration, System Testing.

Practice Questions:

1. What is the main advantage of Agile over the Waterfall model? **Answer**: Iterative development and faster feedback loops.

2. In which SDLC phase are test cases written? **Answer**: Testing Phase.

Resources:

- SDLC Models Overview
- Agile vs Waterfall

Quick Practice Platforms

1. **MCQ Practice**:
   - InterviewBit MCQs
   - W3Schools Quizzes
2. **Problem-Solving**:
   - HackerRank
   - LeetCode
3. **Concepts**:
   - GeeksforGeeks
   - Codecademy