

# Module 1

## Testing Types

- **Dynamic Testing:** Executing code.
- **Static Testing:** Without execution. Examples:
  - Code Review
  - Requirements Review
  - Design Review
- **Black-Box Testing:** No knowledge of implementation. Examples:
  - Login page functionality
  - Cart checkout functionality
- **Grey-Box Testing:** Some knowledge of implementation. Examples:
  - Database query testing
  - API requests testing
- **White-Box Testing:** Full knowledge of implementation.

## Functional vs Non-Functional Testing

- **Functional Testing:** Specific functionality checks. Examples:
  - Login Page Testing
  - Shopping cart functionality
- **Non-Functional Testing:** Beyond functionality. Examples:
  - Load Testing
  - Security Testing

## Testing Levels

- **Unit Testing:** Smallest program parts, precise fault detection.
- **System Testing:** Whole system, high-level fault detection.
- **Integration Testing:** Testing combined units. Approaches:
  - Big Bang
  - Top-Down
  - Bottom-Up

# Module 2

## Testing Approaches

- **Exploratory Testing:** No predefined tests.

- **Specification-Based Testing:** Against specifications.
- **Model-Based Testing:** Against expected behavior models.
- **Fuzz Testing:** Random inputs.

## Partition and Boundary Testing

- **Partition Testing:** Input domain partitioning.
- **Boundary Testing:** Testing boundary values.

## Coverage Testing

- **Statement Coverage:** Execute every statement.
- **Branch Coverage:** Ensure each path is taken.
- **Condition Coverage:** Execute conditions as True and False.
- **Path Coverage:** Execute every independent path.

# Module 3

## Code Coverage

- **Structural Coverage:** Internal code structure.
- **Structural Testing:** Verifying internal code structure.
- **Criteria Subsumption:** Criterion A subsumes B if satisfying A implies satisfying B.

# Module 4

## Review and Analysis

- **Types of Review:**
  - Ad-hoc reviews
  - Peer deskcheck
  - Pair Programming
  - Walkthrough
  - Team Review
  - Formal Inspection
- **Fault Injection:** Testing by introducing faults.
- **Mutation Testing:** Slight changes to test syntactic elements.