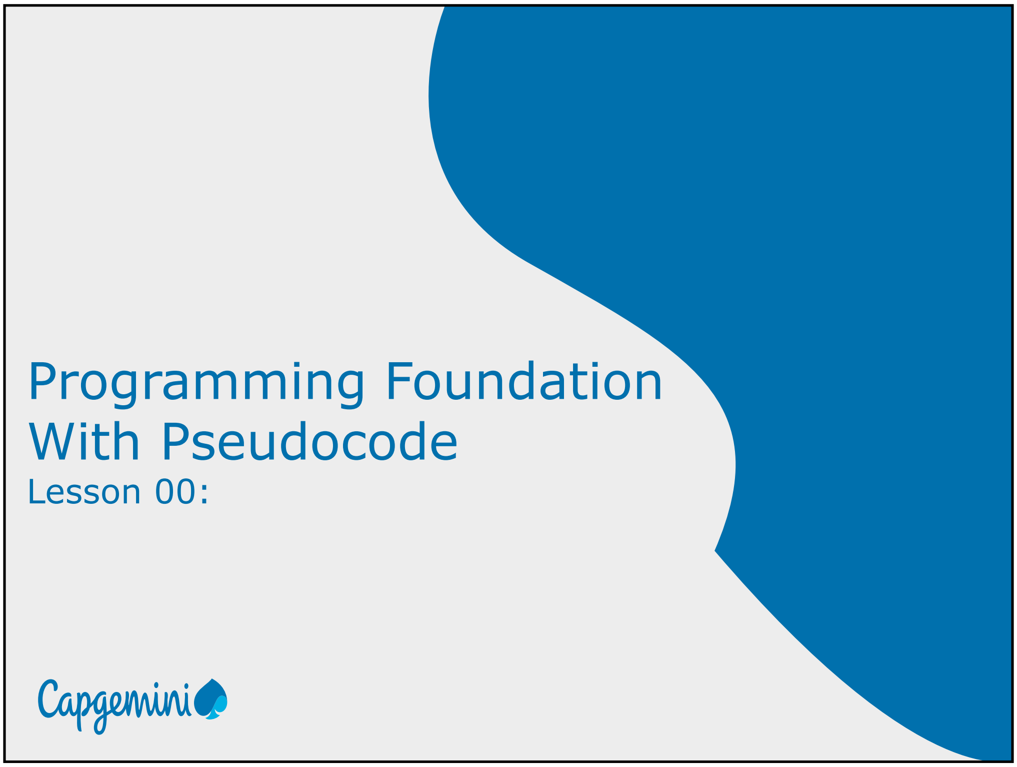
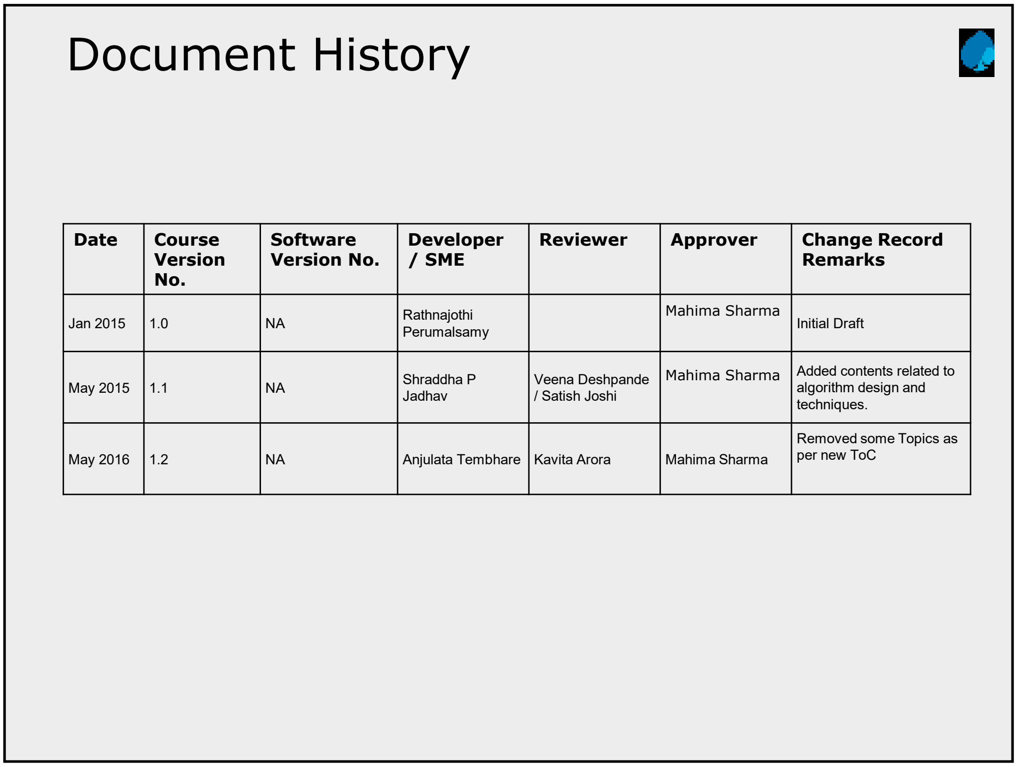


Programming Foundation With Pseudocode

Lesson 00:

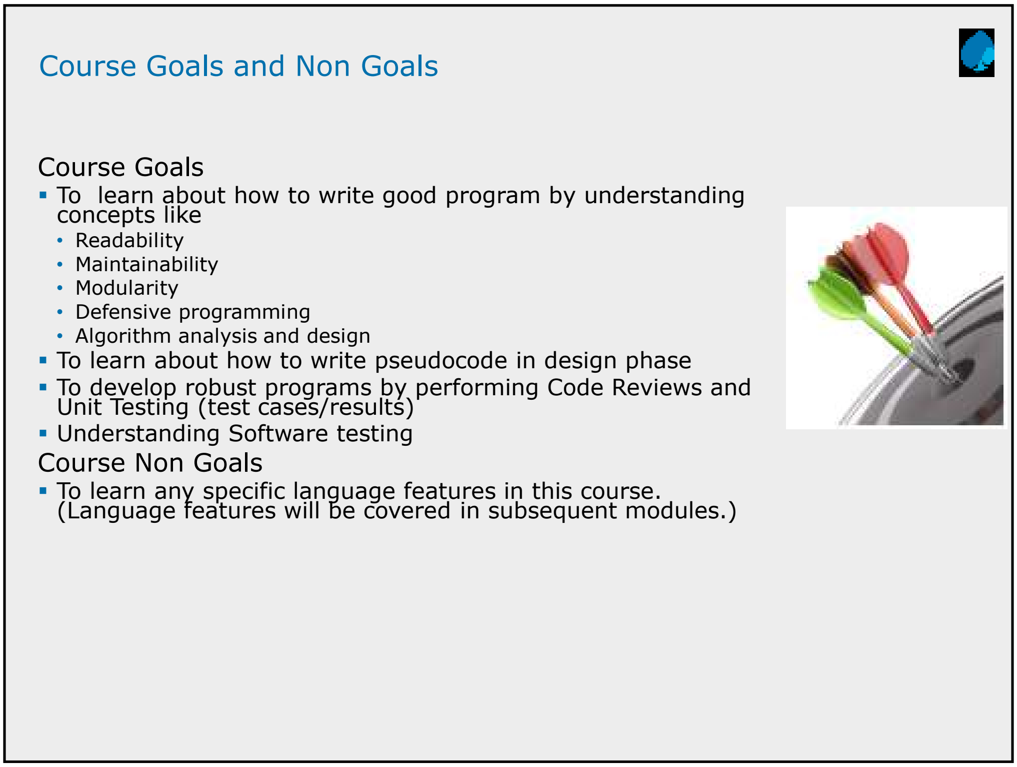


Document History

Date	Course Version No.	Software Version No.	Developer / SME	Reviewer	Approver	Change Record Remarks
Jan 2015	1.0	NA	Rathnajothi Perumalsamy		Mahima Sharma	Initial Draft
May 2015	1.1	NA	Shraddha P Jadhav	Veena Deshpande / Satish Joshi	Mahima Sharma	Added contents related to algorithm design and techniques.
May 2016	1.2	NA	Anjulata Tembhare	Kavita Arora	Mahima Sharma	Removed some Topics as per new ToC

Keep this as a hidden slide.

Note to coordinators: Not to be printed for the class book.



Course Goals and Non Goals

Course Goals

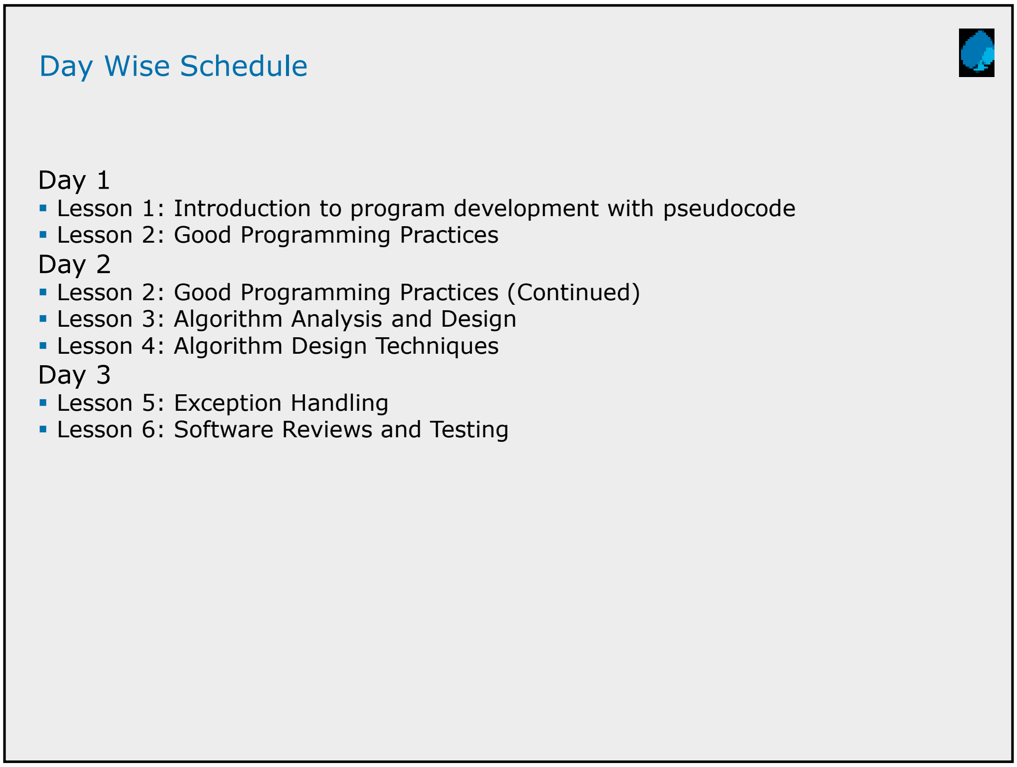
- To learn about how to write good program by understanding concepts like
  - Readability
  - Maintainability
  - Modularity
  - Defensive programming
  - Algorithm analysis and design
- To learn about how to write pseudocode in design phase
- To develop robust programs by performing Code Reviews and Unit Testing (test cases/results)
- Understanding Software testing

Course Non Goals

- To learn any specific language features in this course. (Language features will be covered in subsequent modules.)

Intended Audience

Novice Developers



Day Wise Schedule

Day 1

- Lesson 1: Introduction to program development with pseudocode
- Lesson 2: Good Programming Practices

Day 2

- Lesson 2: Good Programming Practices (Continued)
- Lesson 3: Algorithm Analysis and Design
- Lesson 4: Algorithm Design Techniques

Day 3

- Lesson 5: Exception Handling
- Lesson 6: Software Reviews and Testing

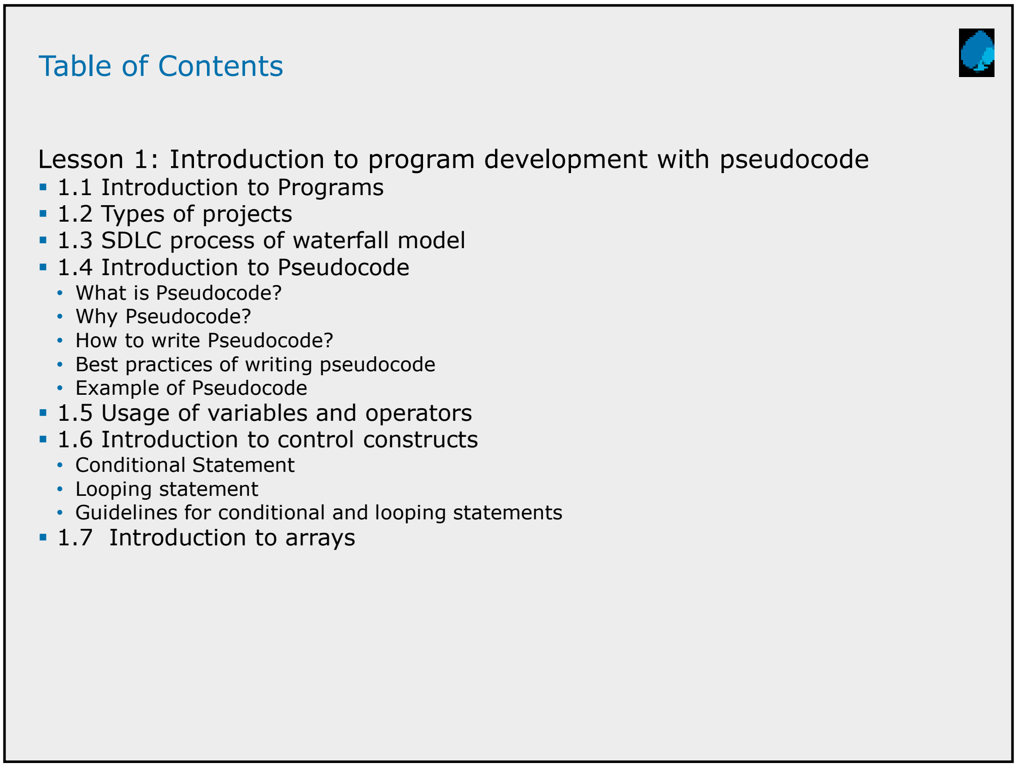


Table of Contents

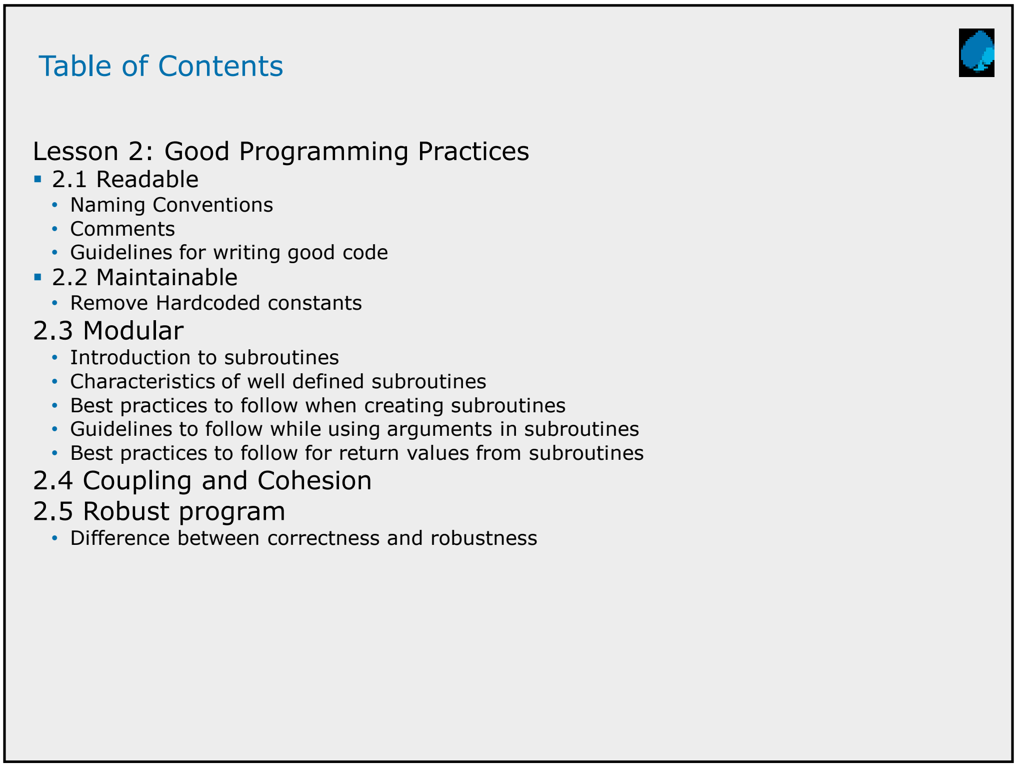


Table of Contents

Lesson 2: Good Programming Practices

- 2.1 Readable
  - Naming Conventions
  - Comments
  - Guidelines for writing good code
- 2.2 Maintainable
  - Remove Hardcoded constants

2.3 Modular

- Introduction to subroutines
- Characteristics of well defined subroutines
- Best practices to follow when creating subroutines
- Guidelines to follow while using arguments in subroutines
- Best practices to follow for return values from subroutines

2.4 Coupling and Cohesion

2.5 Robust program

- Difference between correctness and robustness

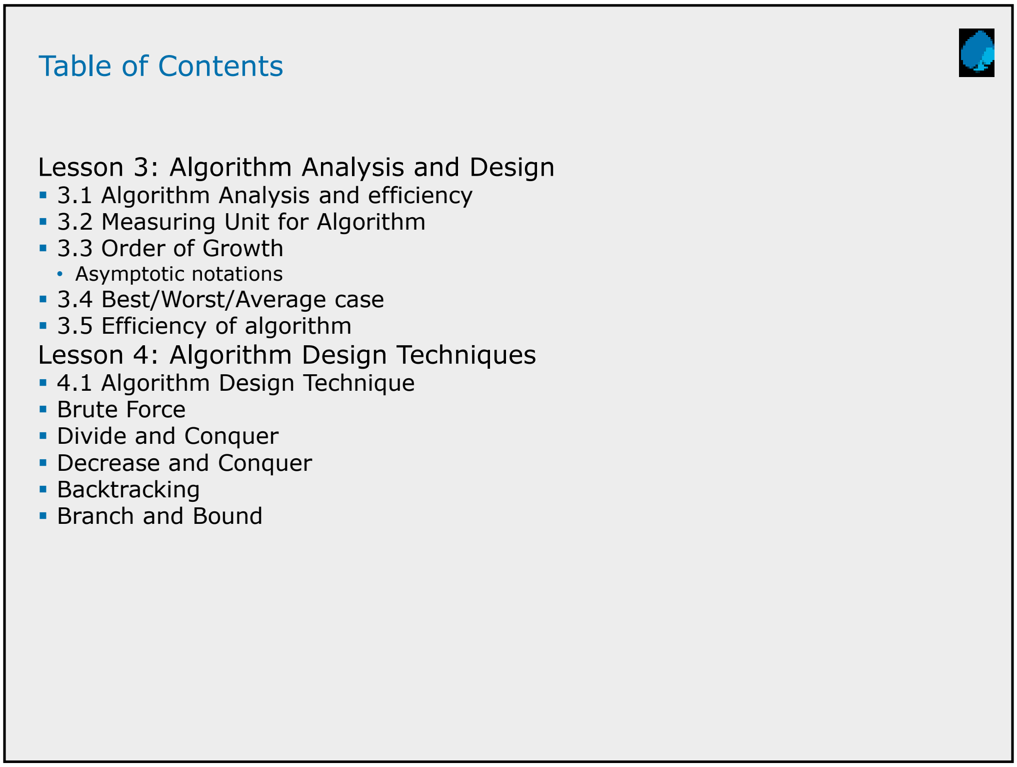


Table of Contents

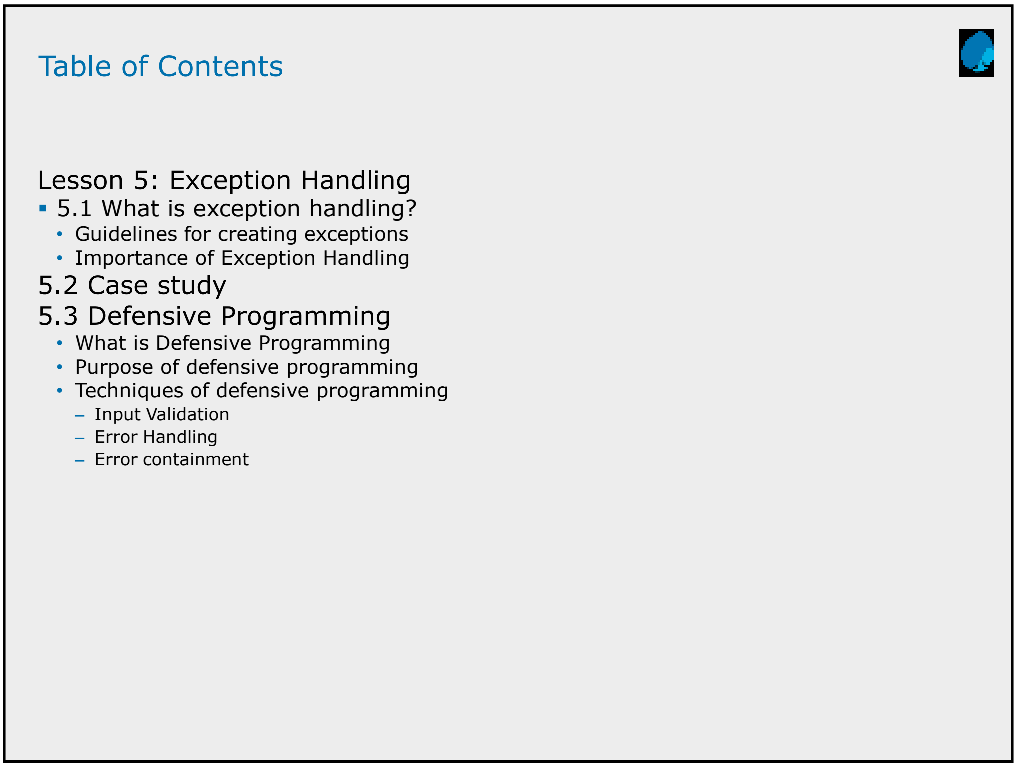


Table of Contents

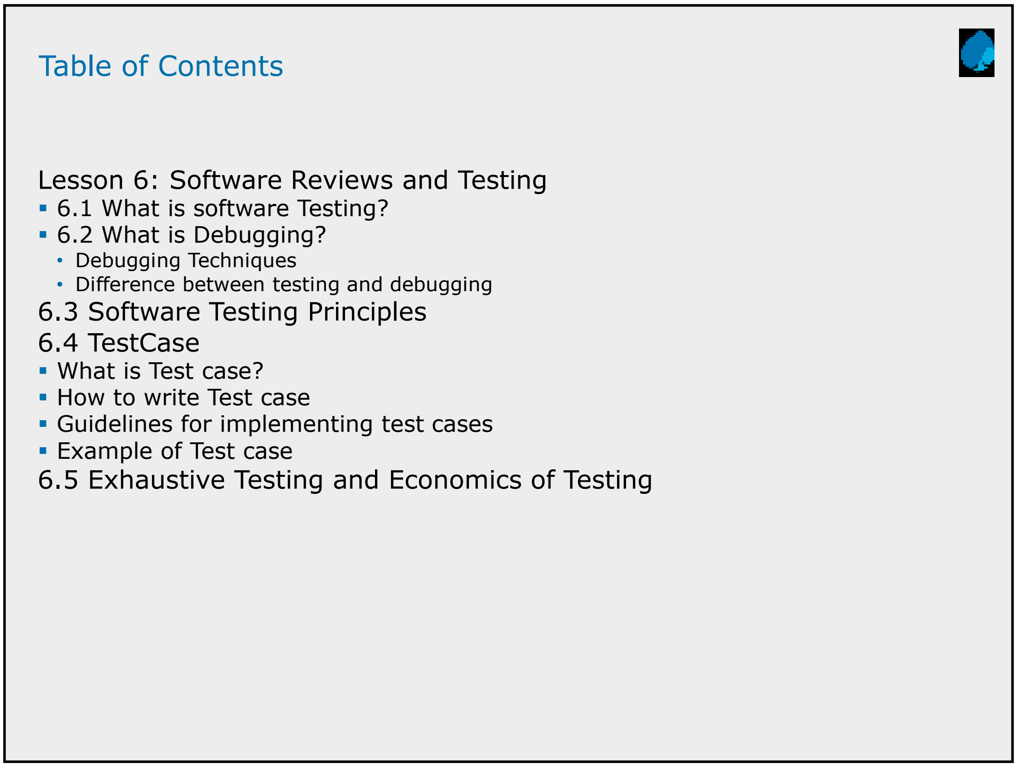


Table of Contents

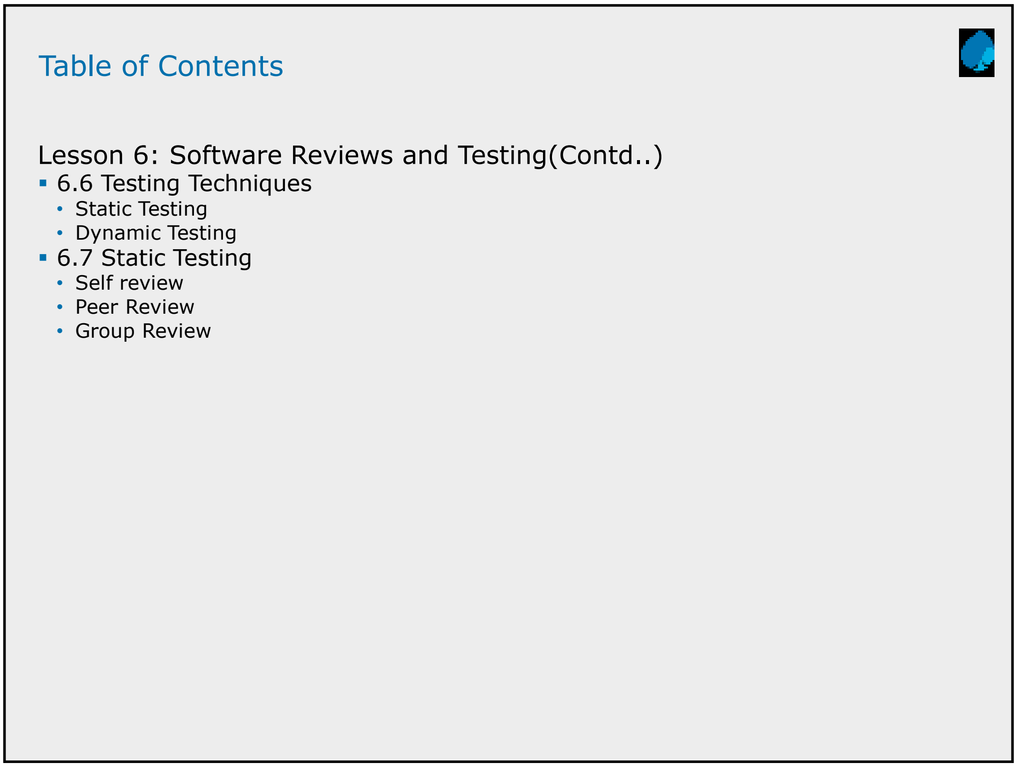


Table of Contents

Lesson 6: Software Reviews and Testing(Contd..)

- 6.6 Testing Techniques
  - Static Testing
  - Dynamic Testing
- 6.7 Static Testing
  - Self review
  - Peer Review
  - Group Review

Table of Contents

6.8 Dynamic Testing

- Blackbox Testing
- WhiteBox Testing

6.9 Testing Approaches

- Unit Testing
- Integration Testing
- System Testing
- Verification and Validation testing
- Acceptance Testing
- Regression testing

Next Step Courses

Any programming language