

Overview of Week 1

#	DATE	LECTURES	IPAD NOTES	STUDY MATERIALS
1	WED, Sep 9 to TUE, Sep 15	<p> Playlist for Week 1 (≈ 4 hours)</p> <ul style="list-style-type: none"> • Part 1A: Software Development Process • Part 1B: Overview of Core Topics • Part 1C: Client vs. Supplier in OOP • Part 1D: Contract, Design, DbC • Part 2A: Bank Accounts V1 (no contracts) • Part 2B: Bank Accounts V2 (precondition) • Part 2C: Bank Accounts V2 (weak precondition) • Part 2D: Bank Accounts V3 (class invariant) • Part 2E: Bank Accounts V4 (faculty implementation) • Part 2F: Bank Accounts V5 (postcondition) • Part 2G: DbC – Java vs. Eiffel • Part 3: Checking Contracts at Runtime 	<p> PRE</p> <p> POST</p>	<ul style="list-style-type: none"> • <u>Tutorials</u> <ul style="list-style-type: none"> ▪ Introductory Tutorials (for Lab0) • <u>Slides</u> <ul style="list-style-type: none"> ▪ Lecture 1: Design by Contract (DbC) ▪ Lecture 1: Design by Contract (DbC) [4-up] • <u>Questions?</u> <ul style="list-style-type: none"> ▪ Post Your Questions in this Document. • <u>Diagrams</u> <ul style="list-style-type: none"> ▪ Roadmap of Core Topics ▪ Runtime Assertion Checking for Contracts - ACCOUNT Class ▪ Runtime Assertion Checking for Contracts - General Case • <u>Source Code</u> <ul style="list-style-type: none"> ▪ DbC in Java (with exceptions and assertions) ▪ DbC in Eiffel (with native contracts)

Learning Objectives of Week 1

Upon completing this lecture, you are expected to understand:

1. **Design by Contract** (**DbC**): Motivation & Terminology
2. Supporting **DbC** (Java vs. Eiffel):
Preconditions, Postconditions, Class Invariants
3. *Runtime Assertion Checking* of Contracts

Milestones of Week 1

- Complete **Quiz 1** (based on lectures)
- Submit **Lab0** (based on intro. tutorials)

Looking Ahead of Week 2

- **Lab 1** (important to finish Lab0 and W1 lectures)