Day 1 Lesson Plan

First Day Stuff

20 mins

- Syllabus[1]
- Clean Code Introduction (6 mins)[2]
- Questions?

Activity 1: Refresher on objects

50 mins

- Brief introduction to classes and objects[3] (work with neighbor(s))

Reading before next class:

- Clean code chapter 2: names[4]

  - Can skip the "avoid encodings" section.

- List 5 useful advices about naming that you learned from this section.

- What is the most important thing the name we choose for an object, class or method should do?

- Identify problems with the following names:

  - d
  - numbersList
  - phoneString
  - getAccount() and getAccountInfo()

- What is the book's recommendation about "overloaded constructors"?

---

[1]../syllabus.html
[2]https://learning.oreilly.com/videos/clean-code/9780134661742/9780134661742-CODE_01_00_00
[3]../activities/activity1-1classesObjects.html
[4]https://learning.oreilly.com/library/view/clean-code/9780136083238/chapter02.html#ch2

Homework 1

- On your own: Watch Clean Code-Remake (54m)[5]

- Answer following questions:

  - What is the main drawback of "messy code that works"?
  - What happens when new members are added to the team?
  - Why are "big redesign projects" not very likely to succeed?
  - Define what each of the following signs of code rot means:
    * rigidity
    * fragility
    * inseparability
    * opacity
  - Explain the phrase "the only way to go fast is to go clean".
  - What are some characteristics of "clean code"?
  - What is the *boyscout rule*?

Paired Homework 1

- Work with a partner.

- Code palindrome example in IntelliJ[6]

- The goal of this assignment is to become accustomed to IntelliJ and how it works. Pay attention to the different parts of the IntelliJ screen and messages and visual cues it provides.

- List 5 interesting facts you learned from this session or questions that you now have about IntelliJ.

---

[5]../videos/01-clean_code.html
[6]../activities/activity1-2palindrome.html