Pro Dev Session

Pro Dev Session

You may want to write this down.

Effectively using version control is an essential part of being a developer. You need to learn git well.

Today you will be learning about code review.

Raise your hand for assistance if you get stuck. We will be using this model for the duration of the course.

Pro Dev Session

You may want to write this down.

Code Review is a peer review process in which every line of code that is committed is reviewed by a teammate prior to being eligible to be merged into master.

Pro Dev Session

You may want to write this down.

The benefits of code review are:

- Fewer bugs in production
- Greater team cohesion
- Knowledge transfer across teams

Pro Dev Session

You may want to write this down.

When reviewing code you will follow the steps below:

1. Find the pull request and visually inspect the code. Does it follow proper conventions? Are variables and methods named well? Do all the functions belong in this class or is their low cohesion? Is it well commented? Can you easily follow what the code is doing? Is there a more efficient way to do this?
2. Locally, use git fetch <branchname> and git checkout <branchname> to retrieve a local copy of this branch. Run the code. Does it work as intended? As a user, can you break it, say if you use the incorrect input type? Is the output (if any) clear and appropriate?
3. In GitHub, note any edits you think would be beneficial. Remember the rules of constructive feedback that you've learned before.
4. Once all changes have been made. Merge the code.

Code Review Practice

INDEPENDENT PRACTICE

It’s time to fly. Focus. Work hard. Ask for help when you need it.

Goals:

- Following the steps on the previous slide, choose a partner and perform a code review on one of their previous homework commits.

20 minutes!

Stand Up!

Project Day

Project Day

THIS IS GOING TO BE FUN

Towards the end of each module we will have dedicated time to apply all the concepts we've learned in a group environment.

Why?

- You've learned a ton this week! The best way to solidify those learnings is to spend some time applying the concepts in new ways.

- This allows us to mimic a workplace as closely as possible. We'll be working in teams with sprint planning meetings, scrums, and retros.

Project Day

THIS IS GOING TO BE FUN

You will work in teams of 3-4, practicing paired programming for the entire activity.

Switch driver and navigator every 30 minutes.

You will essentially be working as 2 independent teams of 2 whose code needs to interact. This means coding to spec is very important.

Project Day

INDEPENDENT PRACTICE

It's time to fly. Focus. Work hard. Ask for help when you need it.

Work in teams of 4 split into 2 pairs.

Goals:

- Have a sprint planning meeting to assign features to pairs (guess at what you can complete in the
- Work for 30 minutes
- Have a scrum and adjust expectations if needed.
- Work for 30 minutes
- Rinse and Repeat

All Day

Project Day

THIS IS GOING TO BE FUN

We are building our own ArrayLists that hold Strings.

Specs:

- Both classes should implement the List interface.
- ArrayLists should use arrays as their underlying data structure for storing data.
- ArrayLists should be dynamic in length (meaning we need to move elements to a new larger array when necessary)

*Do not Google implementing ArrayLists. This is about growth and failing in safe space so you can succeed later!

Project Day

THIS IS GOING TO BE FUN

Implement as many of the features below as possible:

- Build an interface called List with the following methods:
  - add which takes in a String and returns a nothing
  - remove which takes in an int and returns nothing
  - get which takes in an int and returns a String
  - size which returns an int
- Create an ArrayList class that implements the List interface
- ArrayList should have 2 private properties: an empty array of Strings with a length of 10 and an int property called listSize that defaults to zero.
- The add method should add the String parameter to the first empty spot in the array and increment size.
- The size method should return the listSize
- The get method should return the String at the index indicated by the parameter

Once you complete the above, instantiate an ArrayList in your main method and verify that everything works before continuing.

- Update the add method so that if the array is full, it copies all the elements to an array double the current array length and updates the array variable to point to this new array.
- The remove method should remove the String at the index specified by the parameter by shift all

Project Day

THIS IS GOING TO BE FUN

If you finish early, read up on LinkedLists and try to implement a LinkedList (without watching a tutorial).

Project Day

INDEPENDENT PRACTICE

It's time to fly. Focus. Work hard. Ask for help when you need it.

Work with your team.

Goals:

- Have a retrospective
- What went well?
- Did you complete everything you planned?
- How did you share code?
- Could it have gone better?
- What would you do differently next time?

10 min

IT'S TIME FOR OUR WEEKLY RETROSPECTIVE

This is your chance to drive your own education. Let's work together to make our team stronger.

Wrap Up

Module 1 Lesson 5

SUMMATIVE HOMEWORK

This homework is due in 1 week.

Due Tuesday July 16 at 11:59pm

- Summative Assessment: Object-Oriented Programming in Java

Presentations