Pro Dev Session

Pro Dev Session

You may want to write this down.

Giving and receiving feedback are a vital part of being a good developer. Why? Because code review is one of the most important, open, honest feedback sessions you will ever have and it will become part of your daily life.

Also giving and receiving feedback is just part of being a good teammate.

Let’s focus today on giving feedback.

Pro Dev Session

You may want to write this down.

Below are some keys to giving great feedback:

- Ask First: Begin with something like “Could I provide some feedback here?” Respect if the answer is “no”._
- Specific Observation (Positive and Negative): State your observations both positive and negative specifically. “You have been doing an excellent job of managing expectations like when you discussed the risk of falling behind schedule with the client on Monday, however the way you communicated with the client was a little too informal, like when you told that kind of racy joke.
- Impact: State the impact of the observation using “I” terms. “While the client clearly likes you, I’m concerned that that kind of communication may undermine her faith in our ability to handle her account professionally.”
- Actionable Steps: List actionable steps to correct the issue. “Building professional rapport with clients takes a lot of practice. Let’s start by eliminating any questionable language or jokes. If you haven’t read it, you may enjoy the book *The Art of Connecting with People*. Feel free to expense a copy.”

Feedback Practice

INDEPENDENT PRACTICE

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

Work in GROUPS OF 4 to complete all of the goals below.

Goals:

- Individually take out a pencil and paper and draw an alligator (3 min)
- Go around the group critiquing each other's alligators. Be open, honest, and kind. Remember to give actionable feedback. (7 min)
- Give feedback on each other's feedback (Meta feedback!) (5 min)

15 minutes!

Check-in Time

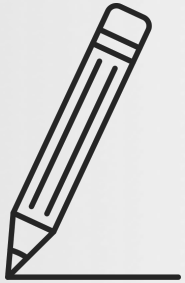
- What makes good feedback?
- What is an example of weak feedback?

Stand Up!

Review

Review

HARNESSING THE WHIRLWIND



Notebooks Ready? It’s time for a review.

Stay Seated &Take 3 Deep Breaths.

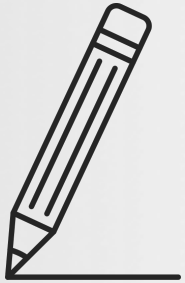
RELAX.

Now take a short walk. Clear your head. After a few minutes break, quickly review your notes. We'll start back in 10 minutes.

Scope

Scope

THE LIMITED REACH OF VARIABLES



Notebooks Ready? It's time for a mini lecture.

Scope

THE LIMITED REACH OF VARIABLES

Reminder: Scope refers to the area in which a variable can be referenced. In Java, variables exist within the closest set of curly braces.

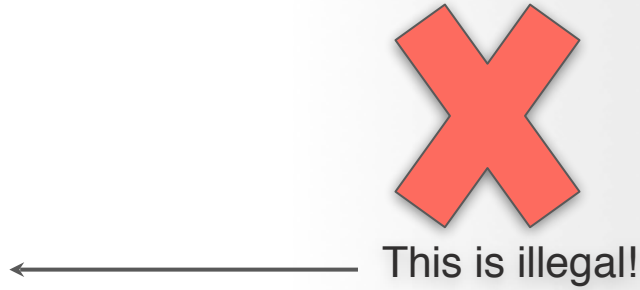
Scope

THE LIMITED REACH OF VARIABLES

Reminder: Scope refers to the area in which a variable can be referenced. In Java, variables exist within the closest set of curly braces.

```
public static int add(int a, int b) {
    return a + b;
}


System.out.println(a);
```



Scope

THE LIMITED REACH OF VARIABLES

Reminder: Scope refers to the area in which a variable can be referenced. In Java, variables exist within the closest set of curly braces.

```
public static int add(int a, int b) {
    System.out.println(a);
    return a + b;
}
```

A-Okay!

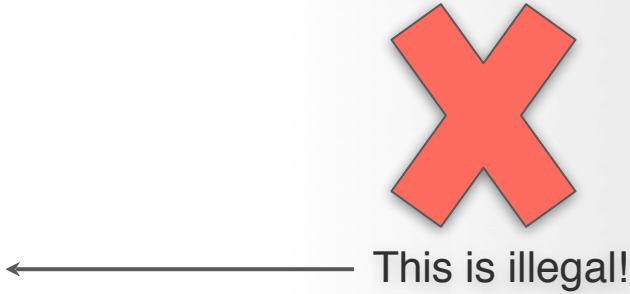
Scope

THE LIMITED REACH OF VARIABLES

This goes for conditionals, methods, and loops.

```
for (int i = 0; i < 4; i++){
    System.out.println(i);
}

System.out.println(i);
```



Scope

THE LIMITED REACH OF VARIABLES

You can create a variable that is available inside EVERY method by declaring outside the methods

Scope

WATCH & LEARN

Close your laptop. Eyes on my screen. Pay attention.

```
class App {
        public static int id = 4;

        public static void printId(){
                System.out.println(id);
        }

        public static void incrementId(){
                id++;
        }
}
```

Scope

INDEPENDENT PRACTICE

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

Work in PAIRS to complete all of the goals below.

Goals:

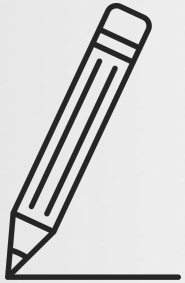
- Create a new project
- Create a main method
- Create a class level String variable called name
- Add 2 more methods: getName and setName
- Use getName to print the name and setName to take in a String and update the value of name.
- In the main method, use setName to set the name variable to your name
- Use getName to print the name

10 minutes!

lunch.

Debugging

CATCHING THOSE PESKY BUGS



Notebooks Ready? It’s time for a mini lecture.

Debugging

CATCHING THOSE PESKY BUGS

Debugging is invaluable. Sometimes our code is doing something other than what we think it is doing. The debugger allows us to see what is actually going on in a program vs. what we think is going on.

Debugging

CATCHING THOSE PESKY BUGS

The debugger allows us to set breakpoints in our code. The code will immediately stop executing when it hits this line and then you can step through the remainder of the program, one line at a time, viewing the values of each variable as you go.