MODULE 1: INTRODUCTION TO PROGRAMMING

Collections Part 1 - Lists

TECH ELEVATOR
A Stride Company

Yesterday

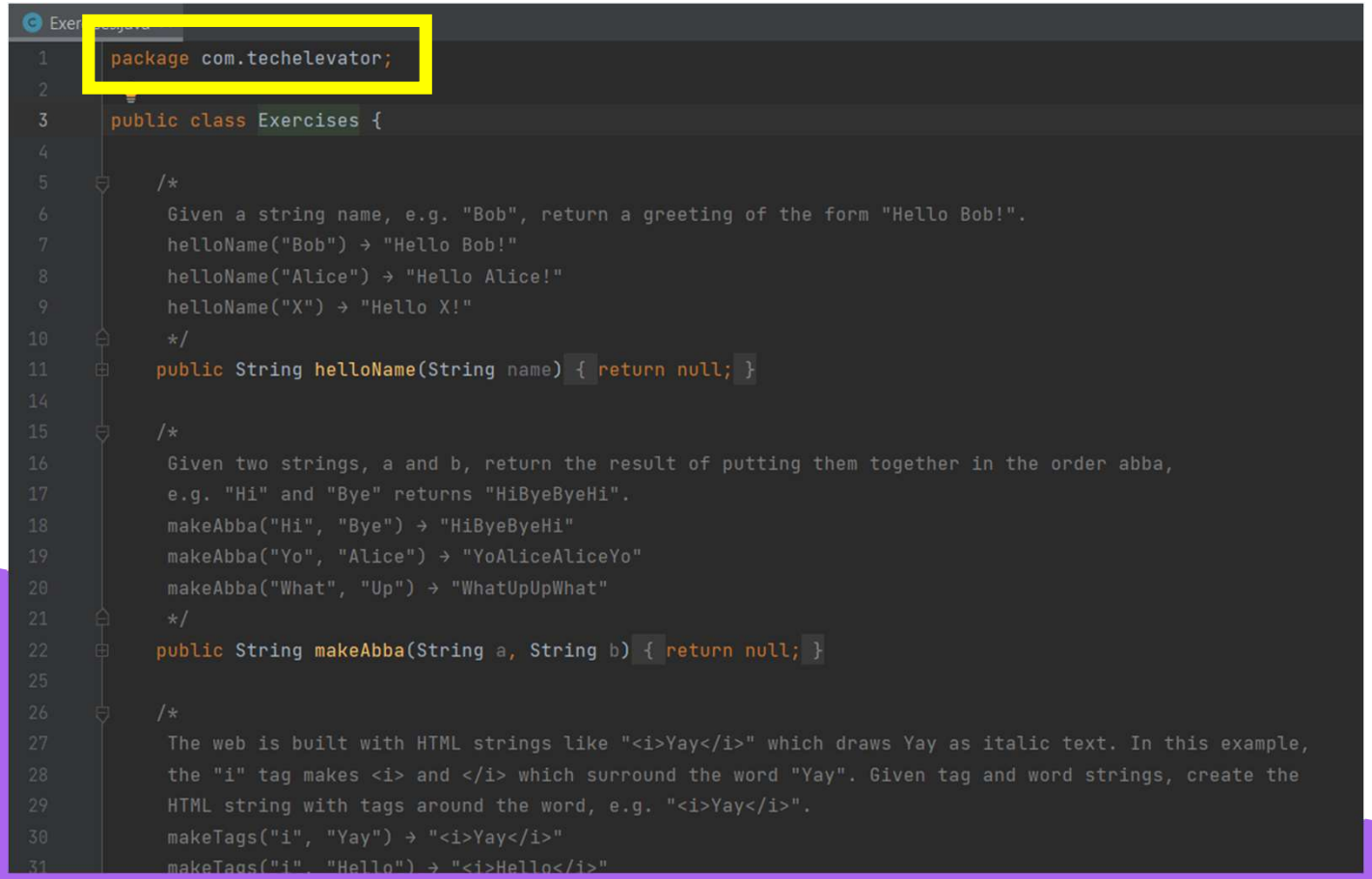
- What is an object?
- What is a class?
- What is the relationship between object and class?
- What is a value type or primitive variable?
- What is a reference type variable?
- Why are there two types?

Right tool for the right job

Collections

- **Collections** classes live in a package or namespace and come from the framework's standard library of classes

Packages



```java
package com.techelevator;

public class Exercises {

    /*
     Given a string name, e.g. "Bob", return a greeting of the form "Hello Bob!".
     helloName("Bob") → "Hello Bob!"
     helloName("Alice") → "Hello Alice!"
     helloName("X") → "Hello X!"
     */
    public String helloName(String name) { return null; }

    /*
     Given two strings, a and b, return the result of putting them together in the order abba,
     e.g. "Hi" and "Bye" returns "HiByeByeHi".
     makeAbba("Hi", "Bye") → "HiByeByeHi"
     makeAbba("Yo", "Alice") → "YoAliceAliceYo"
     makeAbba("What", "Up") → "WhatUpUpWhat"
     */
    public String makeAbba(String a, String b) { return null; }

    /*
     The web is built with HTML strings like "<i>Yay</i>" which draws Yay as italic text. In this example,
     the "i" tag makes <i> and </i> which surround the word "Yay". Given tag and word strings, create the
     HTML string with tags around the word, e.g. "<i>Yay</i>".
     makeTags("i", "Yay") → "<i>Yay</i>"
     makeTags("i", "Hello") → "<i>Hello</i>"
```

Collections: List<T>

Zero-indexed like an array	An ordered set of elements accessible by index	Allows duplicates	**BUT** it can grow and shrink as you add and remove items
			• You can add and remove from the middle even

Declaring and Initializing Lists

- List<T>
  - T is just short hand for and Object Type: Integer, String, Double, etc.
- Declaration:
  - List<String> animalNames;
- Initialization:
  - animalNames = new ArrayList<>();
- All in one:
  - List<String> animalNames = new ArrayList<>();

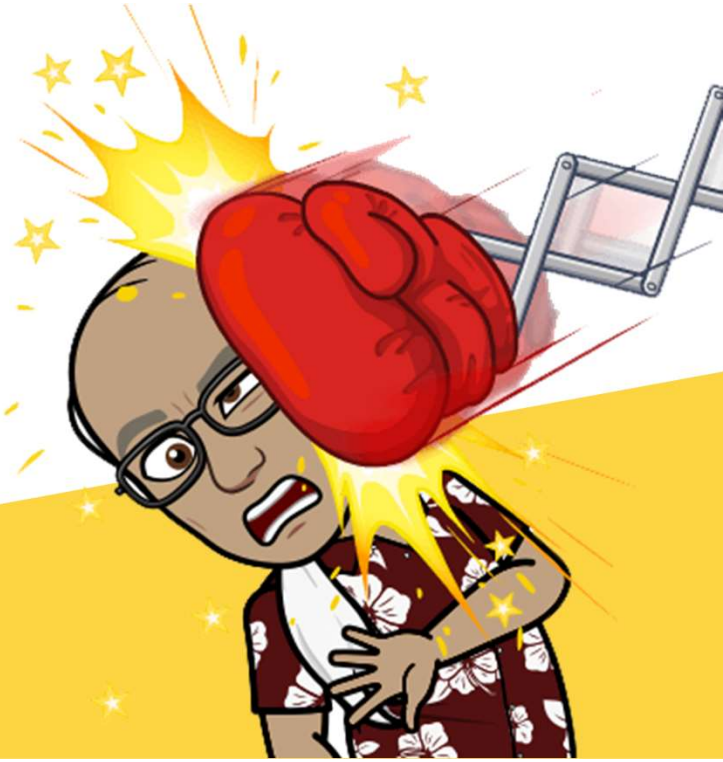
Autoboxing and Unboxing

Primitives cannot be in any of the collections

Primitives are **autoboxed** into a class with a **Primitive Wrapper**.

- int becomes Integer
- double becomes Double
- boolean become Boolean
- Etc.

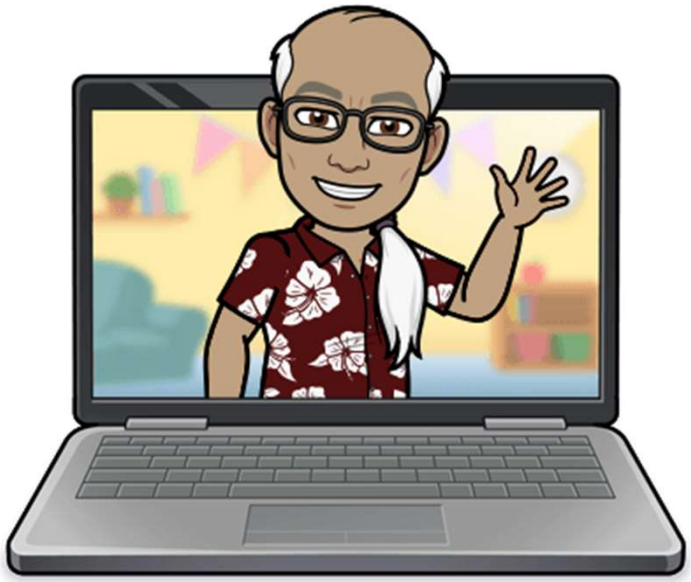
Moving from object to primitive is **unboxing**



Working with Lists

- List<String> animals = new ArrayList<>();
- animals.add(“Koala”);
- String aussieAnimal = animals.get(0);
- animals.remove(“Koala”);

LET’S CODE!



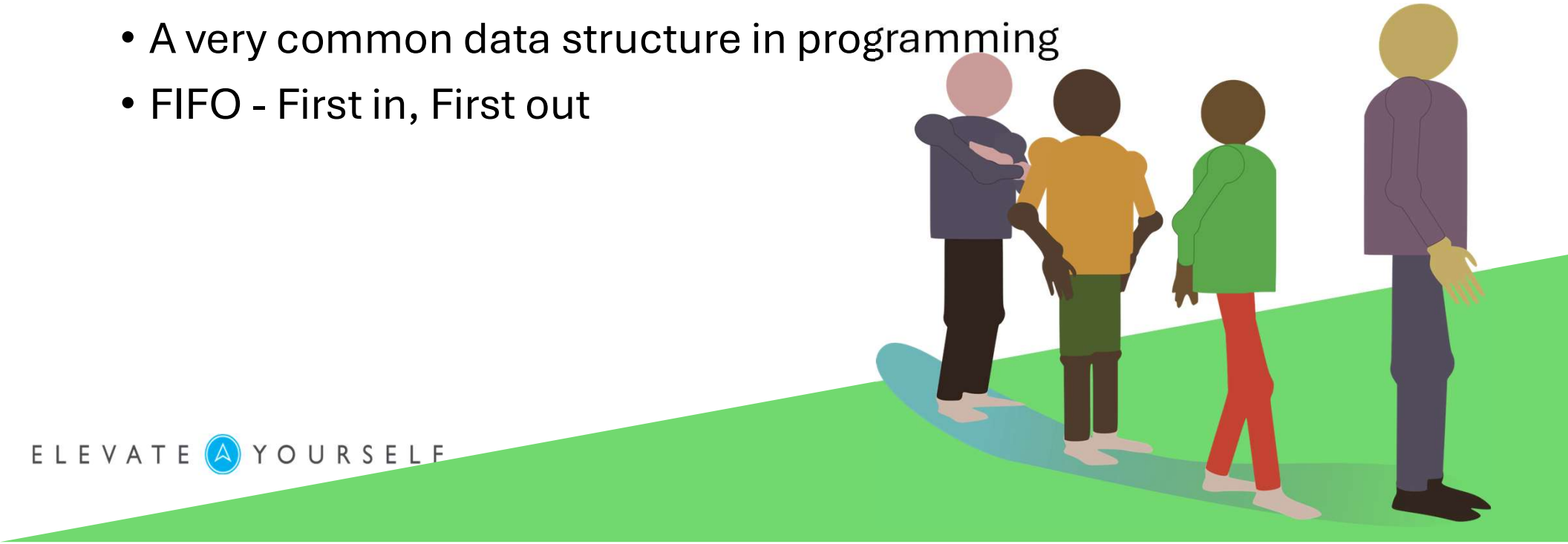
Foreach

```
for (String word : wordsList)
        {
           System.out.println(word);
        }
```

- Convenience method to iterate through collection
- Cannot modify the contents during iteration

Collections: Queue<T>

- Queues are just Lists, but used in a certain way to get a certain result
- A very common data structure in programming
- FIFO - First in, First out



FIFO – Queue<T>

```
Queue<String> zooAnimals = new LinkedList<>();
```

First In:

```
zooAnimals.offer("Panda")
zooAnimals.offer("Kangaroo")
```

zooAnimals.size() is 1

zooAnimals.size() is 2

First Out:

```
String thisAnimal = zooAnimals.poll();
```

zooAnimals.size() is 1

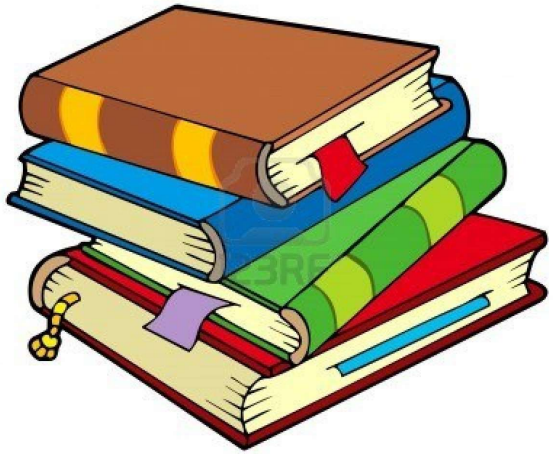
Processing a Queue

- Keep going until you are done with the Queue
- How do you know when you are done?

```
while(zooAnimals.size() > 0)
{
        String currentAnimal = zooAnimals.poll();
        System.out.println(currentAnimal);
}
```

Collections: Stack<T>

- Stacks are, again, Lists of elements but with different behavior
- Another very common data structure in programming
- LIFO - Last in, First out



LIFO – Stack<T>

Stack<String> safariAnimals = new Stack<>();

Last In:

safariAnimals.push(“Lion”)

safariAnimals.push(“Elephant”)

safariAnimals.size() is 1

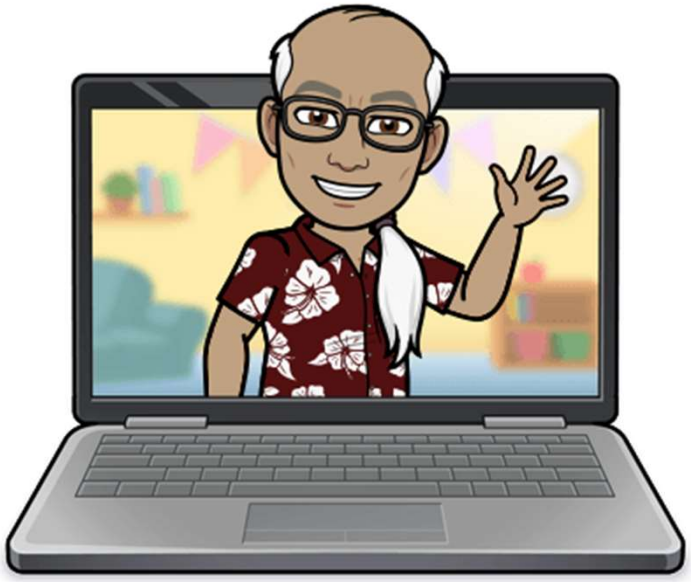
safariAnimals.size() is 2

First Out:

String thisAnimal = safariAnimals.pop();

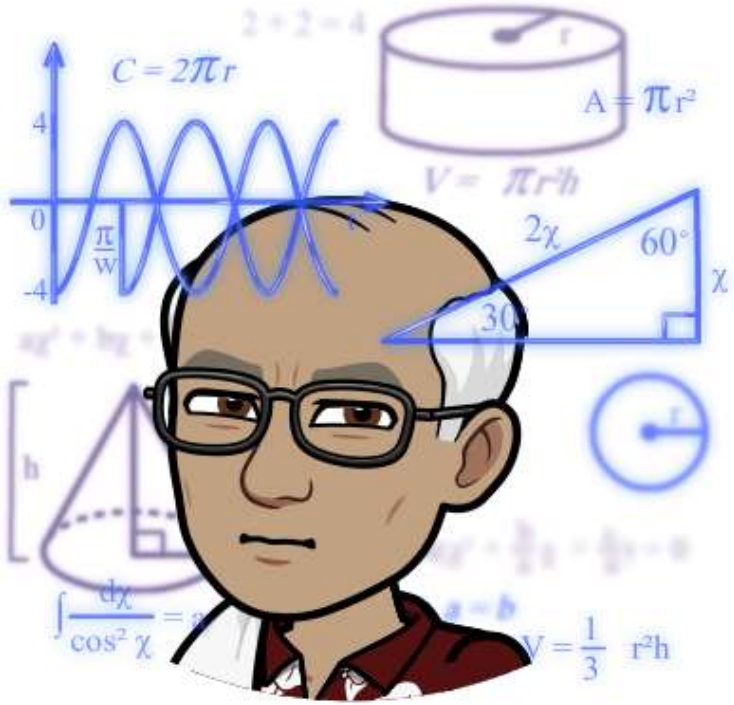
safariAnimals.size() is 1

LET’S CODE!



Collections

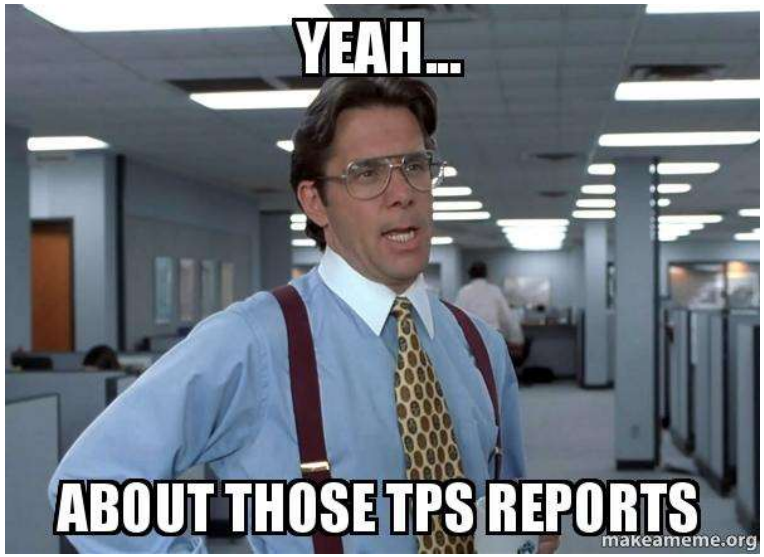
- Arrays
- Lists
- Queues
- Stacks



WHAT QUESTIONS DO YOU HAVE?

Pair Programming

- https://reign.techelevator.com/
- Login with email
- Password:
  - TechElevatorStudent



Reading for tonight:
Collections Part 2