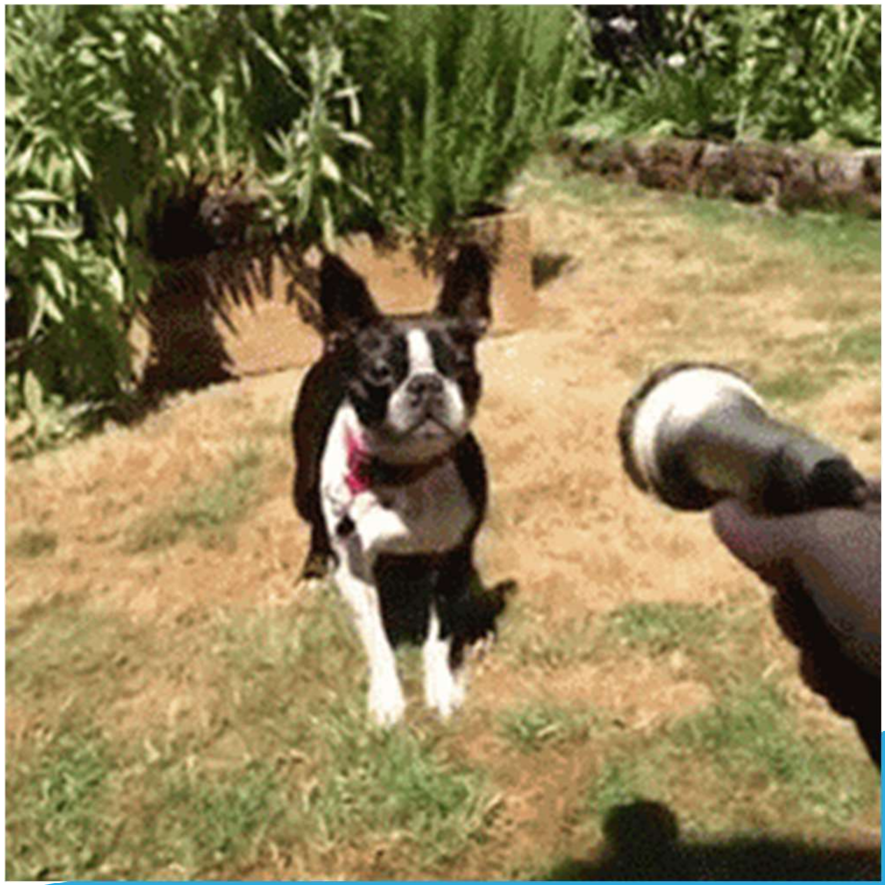
MODULE 1: INTRODUCTION TO PROGRAMMING

Introduction to Objects Using Strings

TECH ELEVATOR
A Stride Company

Feeling a little….

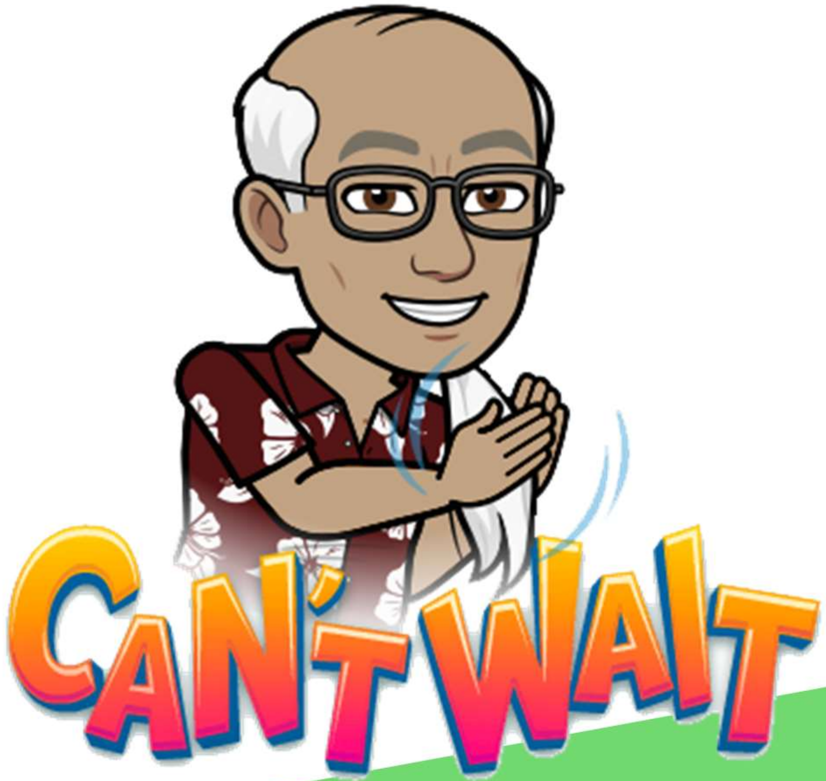


ELEVATE YOURSELF

Last Time

- What is one way to get information from the user?
- How do we give information to the user?

Objects

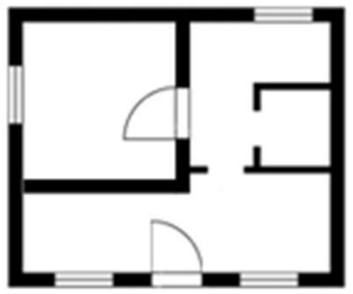


Objects

An **object** is an in-memory data structure that combines state and behavior into a usable and useful abstraction.

Classes

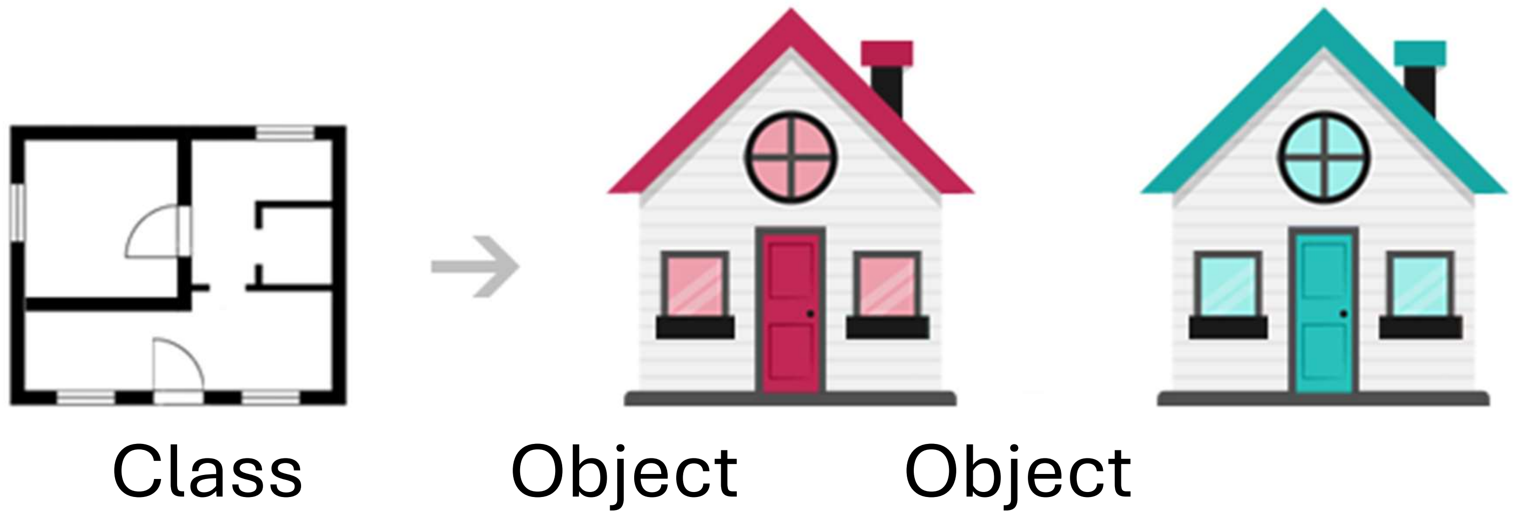
- A **class** is a grouping of variables and methods in a source code file from which we can generate objects.


Blueprint

ELEVATE YOURSELF

Classes

- A **class** is a grouping of variables and methods in a source code file that from which we can generate objects.



Creating Objects

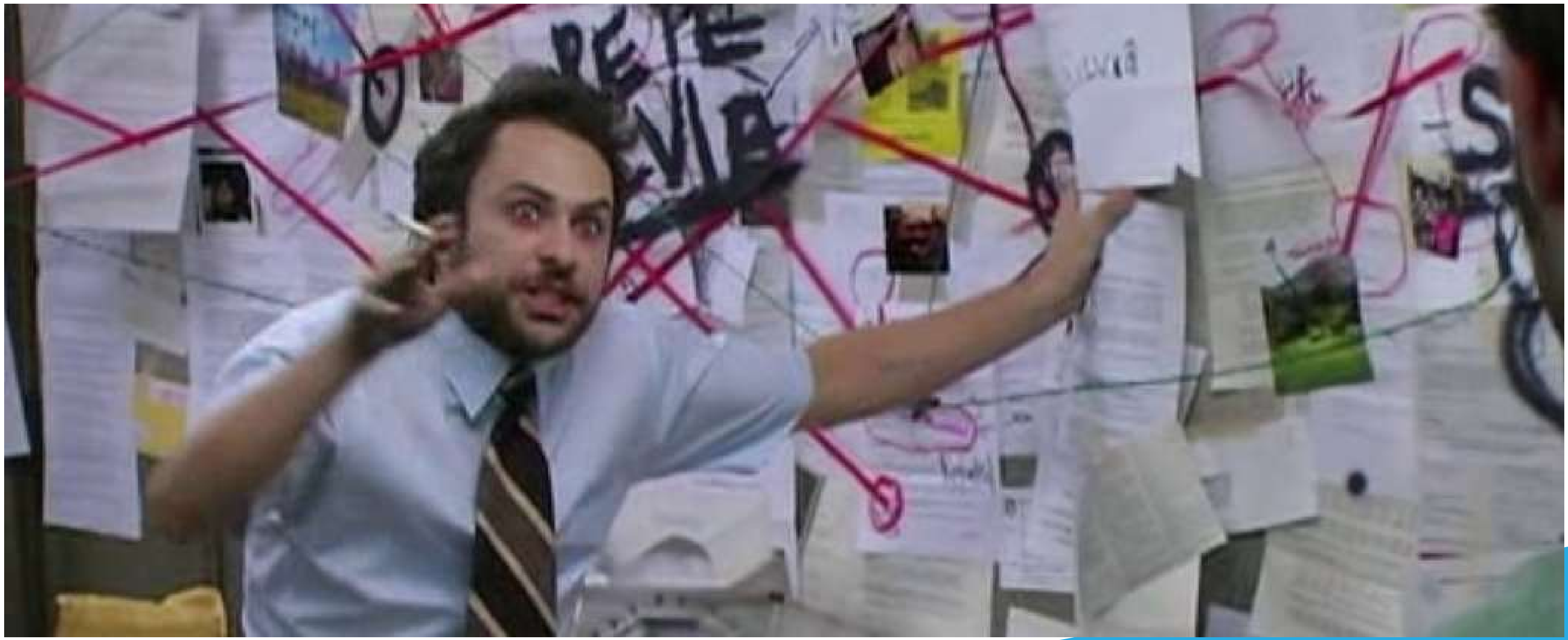
- First, declare a variable with the type of the object
  - House houseAt901Penn;
- Next, instantiate the new object
  - houseAt901Penn = new House();
- Or, instantiate and initialize the object
  - houseAt901Penn = new House(3,2.5,”Red”);
- All at once
  - House houseAt901Penn = new House(3,2.5,”Red”);

Value Types and Reference Types

- int
- boolean
- double
- float
- char
- byte

- Arrays
- Strings
- Objects
- Anything that uses “new”

Stack and Heap



ELEVATE YOURSELF

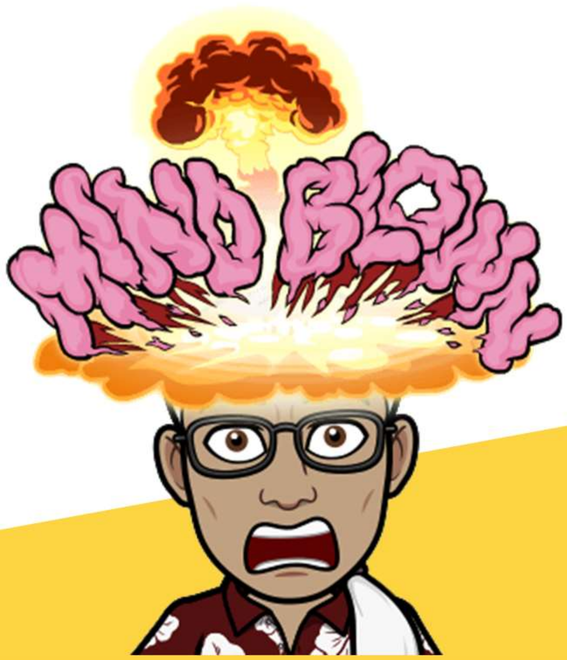
Our First Object: Strings

- Strings are a special case of an object
- Stored as a collection of chars
- Strings are immutable
    - Example: name.toUpper() **returns a string**, doesn’t change *name*.

- Initialization doesn’t require the “new” keyword
    - String foo = “Hello World”;
    - String bar;

Comparing Strings

- How do you see if two ints are equal to each other?
- How do we see if two strings are equal to each other?
- How do we see if two arrays are equal to each other?



ELEVATE YOURSELF

Common String Methods

- .length() : returns the length of a string
- .substring(): returns part of a string based on the parameters
- .contains(): returns a bool indicating if the string contains the parameter
- .startsWith(): returns a bool indicating if the string starts with the parameter
- .endsWith(): returns a bool indicating if the string ends with the parameter
- .indexOf(): returns an int indicating position within the string of the parameter

Common String Methods

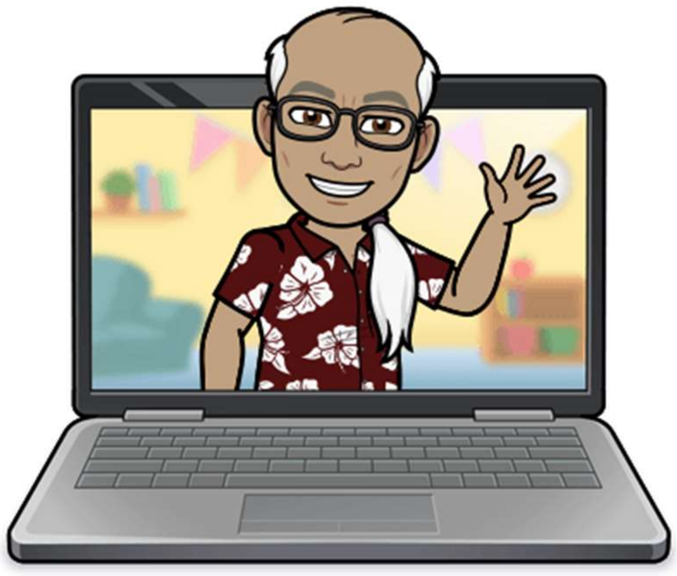
- .replace() : returns new string with characters replaced based on parameters
- .toLower(): returns string with all the characters lowercase
- .toUpper(): returns string with all the characters uppercase
- .equals(): returns a bool indicating if the parameter value equals the string value
- .split(): returns a string array based on the parameters
- String.join(): concatenates an array into a string separated by the specified character.

Recognize the Pattern

- Primitives
  - Value only
  - Stored on the Stack
  - Quick for processing

- Objects
  - Reference types
  - Address stored on the stack
  - Data stored on the Heap
  - Have properties
    - String foo = “Hello World”
    - int numberOfChars = foo.length
  - Have methods
    - String fooUpper = foo.toUpperCase()

LET'S CODE!



ELEVATE YOURSELF

WHAT QUESTIONS DO YOU HAVE?

Reading for tonight:

Collections Part 1