A Question

What are the five largest cities in the United States?

(Write down your answers in order on a piece of scratch paper)

What We Do Best And What Computers Do Best Are VERY DIFFERENT Things

- People are extremely good at:
  - Resolving ambiguity
  - Taking context (the particular situation) into account when processing information

- Computers are very good at:
  - Following explicit instructions over, and over, and over…
  - Never tiring of the same old routine

- Computer are NOT very good at:
  - Resolving ambiguity
  - Figuring out the “right” meaning based on a particular situation

So if we want to tell a computer what to do, we must do so precisely and unambiguously.

What’s Different About Programming Languages?

- The Alphabetize CD’s algorithm (see FIT 9) was precise enough for a person to execute successfully, but computers must have greater precision
  - English is too ambiguous and imprecise

- Programming languages are formal notations specifically designed for specifying algorithms – that means each “word” or “sentence” in a programming language has one and only one interpretation
  - The terms are precise and unambiguous!

- The programming language we will study this quarter is JavaScript-a scripting language

Basics of Programming

To specify algorithms, especially to a computer, we must be precise. To be precise, we need a language that is more exact than our own. A programming language offers this advantage. All programming languages have a basic set of features
What’s Different About Programming Languages?

- Programming involves two critical and interrelated tasks:
  - Figuring out/understanding intuitively what steps need to be taken
  - Figuring out how to specify those steps precisely

Introduction to Programming Concepts

- There are just a few general concepts that apply to virtually all programming languages

- Once you have been exposed to them, you will practice your language proficiency using Java Script in your web pages

- Because there is, literally, hundreds of ways to arrive at the same end product, we’ll show you a few paths to get you started.

Order Matters

- **CONCEPT:** Programming languages execute instructions in order (unless told to do otherwise…we’ll get to that point later)

- The first things listed in a program get done first

- Each instruction is executed one at a time – then the computer goes on to execute the next instruction

- Remember your web pages? The computer (browser) executed the HTML code in the order you wrote the statements

General Concepts

- **CONCEPT:** Being able to store, “remember”, change and access data allows us to write programs that do the same thing but with different data each time.

- The following programming concepts are key:
  - Variables, Names, Values
  - Assignments
  - Expressions
  - Conditionals
  - Iteration

- We will cover the first several of these concepts today

Also important:
- Objects
- Properties
- Events
**Variables**

- **CONCEPT:** *Variable* is the term for a place in memory where the program can store, access, and restore information. Names are easier to reference than number sequences.

  All variables have the following three properties:
  1. A name so that the program can refer to the variable (a location in memory)
  2. A means to store a (new) value in the variable
  3. A means to get (or make a copy of) the value stored in the variable

**Names of Variables**

- Using the term "variable" reminds us that the value can change, that it can vary.

  The names used for variables are arbitrarily provided:
  - Variable names must begin with a letter
  - Variable names can contain any letter, numeral or `_`
  - Most languages are case sensitive: `a` is different than `A`

  Good variable names are meaningful and accurate:
  - `Total, avgOfClass, temp, etc. But not x, tToO, y83928 etc.`

**Values of Variables**

- Values refer to the information stored in the variable (location in memory)

- Variables can take on different types of values:
  - Numbers: 2, -9, 36452729, 2.3, 3.14159, -666.99
  - Character sequences or strings: "2", "dog", "die90wk", " ", " "
  - Boolean Values: True or False

- In most programming languages, each variable should only hold one type of value. This is to:
  - Let the computer know how much memory will be needed to store
  - Allow the computer to help detect errors in the code.

**Declaring Variables**

- Variable declaration tells the computer:
  - That you want a location in memory (the variable)
  - The way in which you will refer to that location in memory throughout your program (the variable name)
  - What type of information you will store in that location in memory, so the computer will know how much space to set aside (the variable type)
  - JavaScript often determines type by the value stored

- **JavaScript - some examples of declaring variables:**
  - `var fname; // declare a variable called name`
  - `var fname, address, city; // declare 3 variables: fname, address, city`
Assigning Values to Variables

- CONCEPT: Computers must be told what value to assign to variables.

- CONCEPT: The general form of an assignment statement is
  `<variable name> <assignment symbol> <expression>`

  - Each language may use a different assignment symbol:
    - `=`
    - `:=`
    - `Å`

  - Assignment means “gets”, “becomes” or “is assigned” and we read it left to right: `A = B` A is assigned B

  - All three components must always be present.

JavaScript: Some examples of variable assignment (placing a value in its container)

```javascript
var myAge = 33;           // declare a variable and assigns it the value 33
destination = "Chicago"; // assign the value "Chicago" to the variable destination
avgOfMidterm = 27;       // assigns value of 27 to the variable avgOfMidterm
avgOfClass = avgOfMidterm; // assigns whatever value is in the variable avgOfMidterm to the variable avgOfClass
```

A Series of Assignments

- Now you work it out …

```javascript
var rock;
var paper;
var scissor;
rock = 2;
scissor = 8;
rock = 4;
rock = scissor;
scissor = 19;
paper = scissor;
rock = scissor + paper;
rock = scissor / paper;
```

Question:
- What’s in rock?
- What’s in paper?

What is the Value of Dude?

```javascript
var dude = 0;           // you can also declare variables and assign them values at the same time
                           // declare a variable and assigns it the value 33
                           dude = dude + 1;
                           // assigns whatever value is in the variable Dude to the variable dude
```

- Questions:
  1. What value does the variable dude contain at the end of this code?
  2. What is this code doing?
  3. What would be a better variable name for dude?
Expressions

CONCEPT: Expressions are a means of performing the actual computation in a program. They are formulae made from variables and operators, e.g. calculator operations: +, -, *, /, ^

- weeks = days / 7;  //divide value of days by 7
- totalAfterTax = totalPrice * 1.087;  //multiply the two values
- FullName = "Grace " + " Whiteaker"; // add 2 strings together-
  // this is called
  // concatenation
  // result: "Grace Whiteaker"
  // stored in FullName

Expressions and Assignment

The Fundamental Rules of Assignment:

- The general form of an assignment statement is
  <variable name> <assignment symbol> <expression>

- The flow of information is always right - to – left

- The expression is evaluated before the assignment is made
  - score = score + 3;  // if the value in score before this
    // line of code was 5, the 5 is added
    // to 3 and then stored back into
    // score, eliminating the previous
    // value