Welcome! Everyone is coming to this class with different backgrounds in computer science, so we all have something to learn from each other in the fall. Your summer assignment: **learn enough of a new programming language** – any programming language – that you can write an *original* and *interesting* program with it and share that program with us.

1. **The Program:** The program must **original** and **interesting**. Original: it’s your own idea, not the completion of a tutorial project on one of the numerous learn-to-code websites. *Students who have taken AP CSP may not use Code.org.* Interesting: this is open to interpretation, but my basic rule-of-thumb is that it's something you want to show off to your family, friends, or class. Simple and original is fine! The grading of this first assignment is lenient – it’s mostly your pride at stake. If you are totally new to programming (as many of you are) don’t worry if your program feels too simple. I care more about originality and creativity for this assignment, so use your imagination and have fun. No judgement.

2. **The Reflection:** Answer the following questions (typed) after you’re done as your *reflection*:

   1. What does your program do?
   2. Why do you like your program?
   3. What resources (programs, websites, etc.) did you use to create it and/or gain inspiration?
   4. What surprised you in the process of writing it? What was the hardest part of the process? (Be specific!)
   5. What else would you like to learn how to do in class this year?

3. **The Demo:** We will also demo our programs as soon as everyone is set up with a computer, so plan a quick, two minute demonstration of your program.

**Due date:** First day of class (turn in *printed copies* of your program and your reflection questions). If you cannot print, please email me your files before the first class.

Note: We’ll be focusing on the Java programming language this year (it’s what they test on the AP exam), so I recommend learning something *else* (NOT Java) to broaden your perspective. Also, Java is designed to be great for huge, industrial-strength projects, so it’s not the ideal *first* language to learn (i.e., it’s not the easiest for newbies to teach themselves). (Also note that Java ≠ JavaScript.) Finally, for this assignment at least, HTML and CSS don’t count as programming languages.

That’s it: go get programming! (Resources are on the back if you need help getting started.)
Suggestions (if You’re Stuck)

In recent years the number of online learn-to-code websites has exploded and I won't pretend to be familiar with all of them. (Though some cost money, there are plenty of excellent free ones, so don't feel pressured to pay.) I've listed a few I'm familiar with below, but you may just want to start with an online search for programming tutorials for an area that interests you.

If you want to learn...

- JavaScript with KhanAcademy: [https://www.khanacademy.org/cs](https://www.khanacademy.org/cs)
- Android programming with AppInventor, a Java-based "blocks" language: [http://appinventor.mit.edu](http://appinventor.mit.edu)
- iOS or macOS programming with Swift Playgrounds for the iPad or Mac (running Catalina or later): [https://www.apple.com/swift/playgrounds/](https://www.apple.com/swift/playgrounds/) (If you already know how to program and want to try the “real” version of Swift with xCode: [https://www.apple.com/swift](https://www.apple.com/swift))
- Python, JavaScript, or Ruby with CodeAcademy: [https://codecademy.com](https://codecademy.com)
- Alice, a simplified, graphical, 3D version of Java (also using "blocks"): [https://alice.org](https://alice.org)
- Scratch, a blocks-based Java language: [https://scratch.mit.edu](https://scratch.mit.edu)
- Glitch [https://glitch.com](https://glitch.com)

The possibilities are endless! All you need is a computer and an internet connection. If you’re lacking decent access to either of those, find a way to e-mail me <alex.jacoby@k12.dc.gov> and we’ll figure something out!

**Final Reminder:** If you do follow an online tutorial, your "interesting program" can’t be identical to one of the programs in the tutorial. You may use one of their programs as a starting point, but the final program must involve a substantial amount of your own creativity, and you should explain how you extended it in your reflections. I'd much rather see something simple that YOU created than something fancy that was largely done by someone else. Have fun!