Week 1 Lab

In the week 1 lab we will:

* Confirm your environment
* Create a GitHub account, a repository, share that to the instructor, issue a Pull Request for your Karel solutions
* Program Karel to solve multiple world-problems

Confirm your environment

This course will make use of at least the following freely available software:

* Git / GitHub CLI
* VS Code
* Java (version 8)
* NodeJS
* Chrome or FireFox

We will install these programs as we arrive at their usage points. The computers in our classroom may *not* be used to run these applications. Instead you must either run them on a vSphere instance or onto your own personal computer (preferable if available).

For this week 1 lab we will install Git and Java as detailed below.

GitHub

We will use Git / GitHub extensively throughout this course. Start by installing the GitHub CLI onto your computer(s). Once installed run the following command to login:

git auth login

Follow the prompts to complete login.

This GitHub repository is where you will find supporting materials for the course:

<https://github.com/akennis/sunyorange-csc227>

You should clone this repository onto the computer(s) that you use for course work (e.g. a vSphere instance or your personal computer).

Please create a GitHub repository under your own account and name it ‘sunyorange-csc227’. You can choose to make your repository public or to keep it private, but in that case please give your instructor access via his GitHub account:

https://github.com/akennis

Clone your repository onto your computer(s). Once available locally, edit (or create if necessary) your README.md file. Add any text (e.g. a description of the course) to the file and save your change. You can then run this command to confirm that you have a modified file:

git status

Run these commands to create a commit for your change:

git add .

git commit -m ‘Readme update’

Now run this command to create a pull request for your change:

gh pr create –fill

Follow the prompts and once the PR is up tell your instructor and he will review / approve it. Once approved please do a Squash and Merge from the PR web page and conclude by deleting the branch.

Confirm the final result by viewing the commit under your main branch.

Karel Setup

Please find the file JSKarel.jar under the karel directory in this repository (which you should already have cloned onto your computer):

<https://github.com/akennis/sunyorange-csc227>

Your computer must have *Java 8* on it (newer Java versions don’t work for this program). Java 8 can be found here: <https://www.oracle.com/java/technologies/javase/javase8-archive-downloads.html>

Confirm that you can open Karel by running the following command:

java -jar JSKarel.jar

Note: if your computer has multiple versions of Java installed you will have to run the specific Java 8 executable – on my Ubuntu machine the command looks like:

/usr/lib/jvm/java-8-openjdk-amd64/bin/java -jar JSKarel.jar

Karel Programming

Saving into the karel directory of your own repo, write solution (.k) files to all of the world (.w) files in the labs/lab-1 directory of the course repository. When ready issue a pull request containing your solutions following the same process as outlined above. Your instructor will review your work, leaving comments where appropriate, and once those have been addressed via follow-on commits, approve your work to be merged (don’t forget to squash during merge).