Adrien Protzel

1. Programming languages, each with unique strengths and weaknesses, are like diverse tools suited for different aspects of complex tasks.
2. Languages and their syntax is a groundwork on which API’s are made to compliment or implement already built blueprints of functions and features. An API for a language is like a set of pre-made functions that can be easily plugged into a project and used. An API cannot be universal and must be made to a certain language and often to a certain version due to syntax, which is the way the language reads commands and then executes those commands such as empirical vs functional.

Python syntax/feature: Creating a list

my\_list = [1, 2, 3, 4, 5]

Pandas: Creating a list dataframe

import pandas as pd

df = pd.DataFrame(my\_list, columns=['Numbers'])

1. Python: the initial release was simple and small. They had the basic functions and mathematics to cover most operations with a single solution for each. Their newest version in py3k has removed many additions from version 2x to reduce redundant functions to go back to having one good way to complete a task. While they are more functional than ever, they have tried to come back around to having simplicity of a single solution to a problem. Even then, there are still many solutions made by many people from over the years. The latest version also added Unicode compatibility which the initial version did not have. There are now many changes to initial functions like print to print() which now caused many old python programs to be incompatible with the newest version.