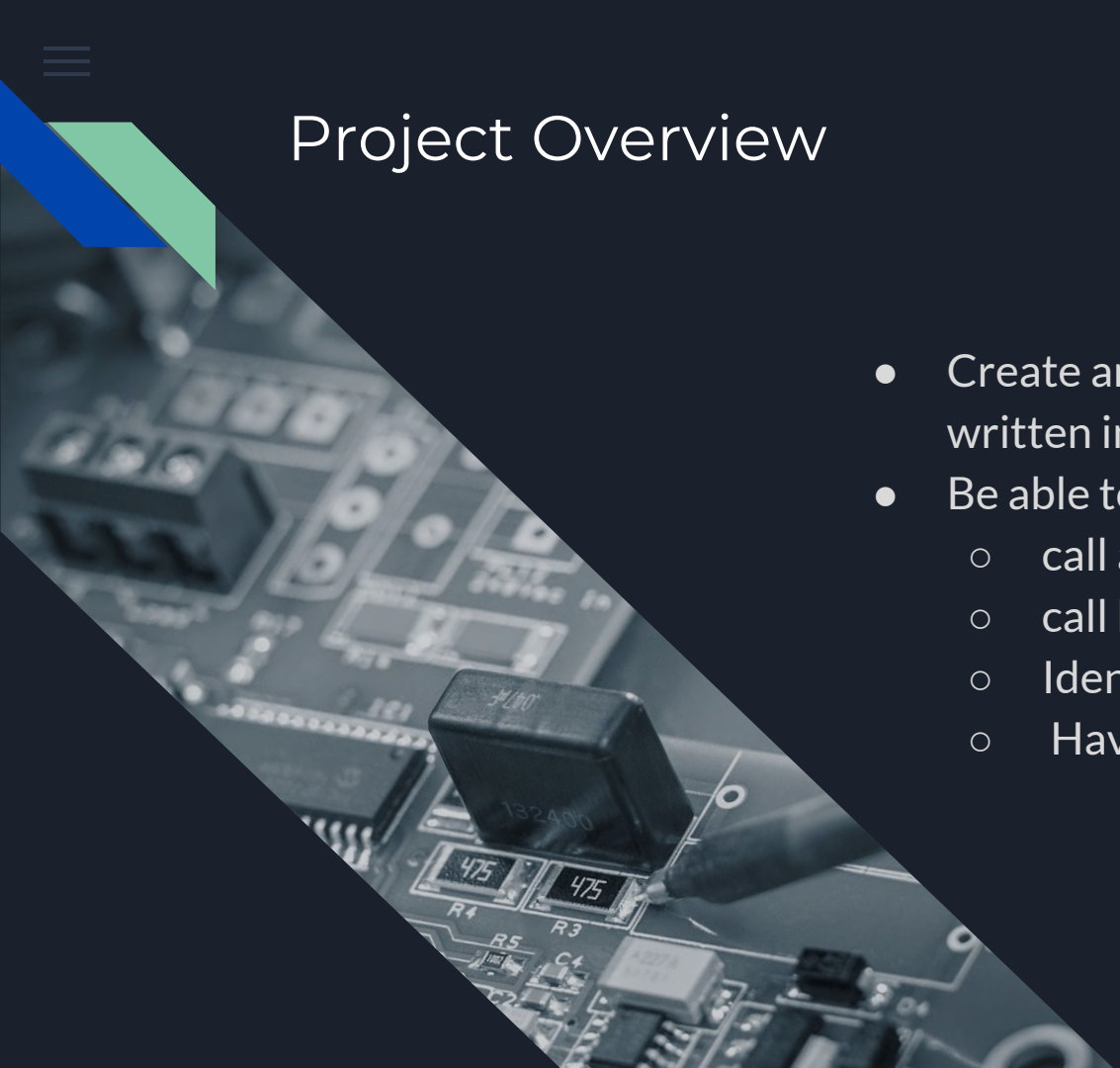
Programming Languages - Python Interpreter

Agenda

Project Overview
Approach
Project Timeline
Improvements
Demo

Project Overview



- Create an interpreter of Python that is written in Python
- Be able to:
  - call and edit variables
  - call basic function and loops
  - Identify Strings and Integers
  - Have Python like indentation

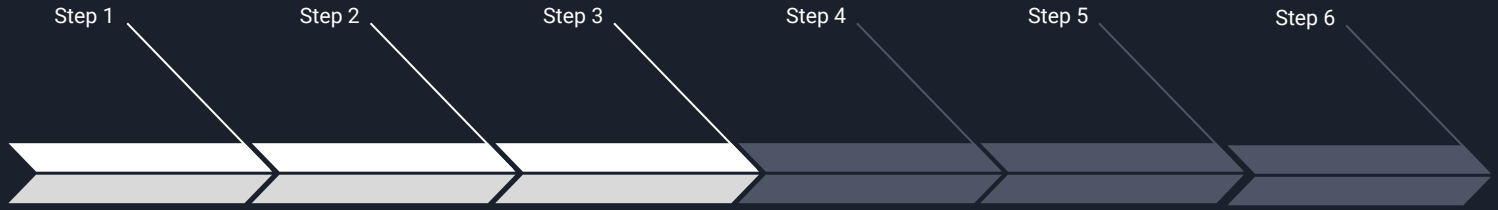
Approach

01 Understand Python Language and figure out the methods to develop for the project

02 Create code using Top Down Programming methodology to create functions and classes

03 Test, Optimize, and improve

Project Steps



Define primary functions

Start with Top Down Programming to define basic functions that will be used

Create Global Dictionary

Created variable to be able to store and maintain the state

Evaluate Expressions

Create a way for interpreter to evaluate and return results of expression

Catch Syntax Errors

Create a clean easy way to show errors and the name of what is what

Multi-line Functionality

This allows for the code to act more like Python and follow Python style indentation

Expand Functionality

Build upon the existing code base to develop the functionality that will be demoed

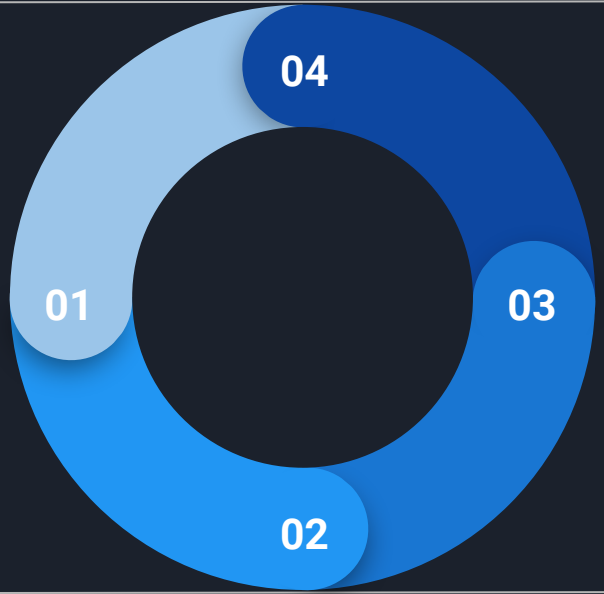
Key Implementations

Multi-Line

Allow the code to operate using multi-line functionality

If Statements

Program if and if/else statement to allow for more complex functionality



Dynamic Variables

Allow variables to be called names and updated with simple commands

Loops

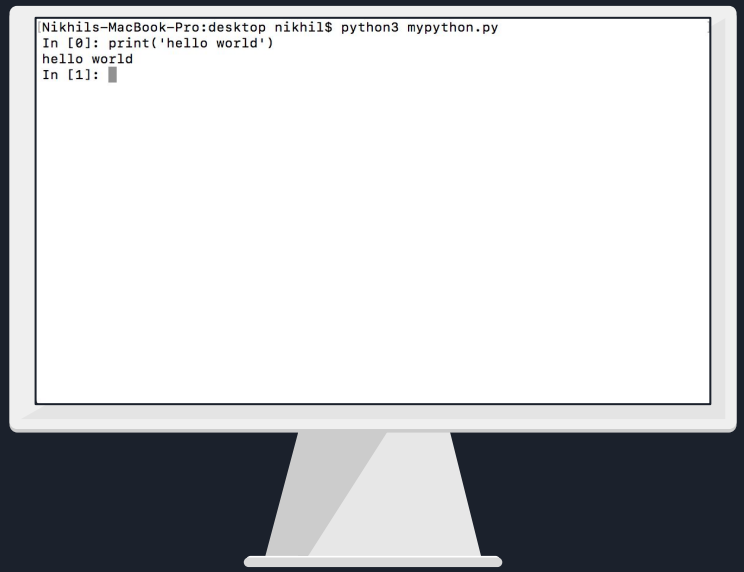
Create loops to be able to run more complex functions

Demo

Showcase of the work that has been done

Steps To Open and Run:

- In command line run the code ‘python3 mypython2.py’
- Run code as if it was python
- To execute code: “RUN <enter>”



Thank you!