Dev Environment Setup

Tools install and setup

Before we can start developing software, we need to set up our development environment. This means installing our development tools, and configuring their settings.

I use Linux, so these steps may be slightly different for Windows or Mac.

You can google for instructions on how to install them if you run into problems. Look for the official docs from the official website for each program.

Install

You may already have some of these things already installed.

You can check if they are installed by running the following commands in your terminal program:

```
python --version
pip --version
sqlite3 --version
git --version
```

If a version number shows in the terminal then the program is already installed.

The first program you need to install is Visual Studio Code from - https://code.visualstudio.com/

If you are using a Mac:
- Open VS Code and install it to the command line
- Open VS Code
- press SHIFT+CMD+P
- and then type: Shell Command: Install 'code' command in PATH

Then, install the following programs

- python3 (https://wiki.python.org/moin/BeginnersGuide/Download)
- python3-pip (https://pypi.org/project/pip/)

- to install pip on Mac
  -open the Terminal program and type the following lines:

  ```
  curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py
  python3 get-pip.py
  ```

- python-is-python3 (Linux only - Ubuntu 20.04+)
- pipenv (https://pypi.org/project/pipenv/)
- pytest (https://docs.pytest.org/en/6.2.x/)
- pylint (https://pylint.org/)
- flake8 (https://flake8.pycqa.org/en/latest/)
- black (https://pypi.org/project/black/)
- sqlite3 (https://sqlite.org/quickstart.html)
- Git (https://git-scm.com/)
- VS Code Extensions: (How to install - https://code.visualstudio.com/docs/editor/extension-marketplace)
  - Python (ms-python.python)
  - Pylance (ms-python.vscode-pylance)
  - Test Explorer UI (hbenl.vscode-test-explorer)
  - Python Test Explorer for Visual Studio Code (littlefoxteam.vscode-python-test-adapter)
  - Test Adapter Converter (ms-vscode.test-adapter-converter)
  - Python Docstring Generator (njpwerner.autodocstring)
  - Bracket Pair Colorizer (coenraads.bracket-pair-colorizer)
  - EditorConfig (editorconfig.editorconfig)
  - Visual Studio Intellicode (visualstudioexptteam.vscodeintellicode)
  - REST Client (humao.rest-client)
  - SQLite (alexcvzz.vscode-sqlite)

Setup

VS Code

- Settings

(https://code.visualstudio.com/docs/getstarted/settings)

You can modify your VS Code settings by pressing `CTRL+,` or go to File → Preferences → Settings. In the top right corner there's a file icon with a magnifying glass. Hover on it and it should say Open Settings (JSON). Click that and copy/paste these settings there.

```
"python.formatting.provider": "black",
"python.linting.flake8Path": "/usr/bin/black",
"python.linting.flake8Enabled": true,
"python.linting.flake8Path": "/usr/bin/flake8",
"python.linting.pylintEnabled": true,
"python.linting.pylintPath": "/usr/bin/pylint",
"python.linting.enabled": true,
"python.testing.pytestArgs": [
  "tests",
  "-v"
],
"python.testing.unittestEnabled": false,
"python.testing.nosetestsEnabled": false,
"python.testing.pytestEnabled": true,
"[python]": {
  "editor.codeActionsOnSave": {
    "source.organizeImports": true
  },
  "editor.defaultFormatter": null,
  "editor.formatOnSave": true,
  "editor.formatOnSaveMode": "file",
  "editor.insertSpaces": true,
  "editor.tabSize": 4,
  "editor.useTabStops": true
},

"rest-client.environmentVariables": {
  "$shared": {
    "jsoncontent": "Content-Type: application/json"
  },
  "Dev Env": {
    "flaskserver": "http://localhost:3000"
  }
},
```

- Keyboard Shortcuts

(https://code.visualstudio.com/docs/getstarted/keybindings)

You can modify your VS Code key bindings by pressing `CTRL+K CTRL+S` or go to File → Preferences → Keyboard Shortcuts. In the top right corner there's a file icon with a magnifying glass. Hover on it and it should say Open Settings (JSON). Click that and copy/paste these settings there.

```
[
  {
    "key": "ctrl+shift+s",
    "command": "workbench.action.files.saveAll"
  },

  {
    "key": "ctrl+up",
    "command": "editor.action.insertCursorAbove",
    "when": "editorTextFocus"
  },
  {
    "key": "ctrl+shift+up",
    "command": "-editor.action.insertCursorAbove",
    "when": "editorTextFocus"
  },

  {
    "key": "ctrl+down",
    "command": "editor.action.insertCursorBelow",
    "when": "editorTextFocus"
  },
  {
    "key": "ctrl+shift+down",
    "command": "-editor.action.insertCursorBelow",
    "when": "editorTextFocus"
  },

  {
    "key": "ctrl+shift+/",
    "command": "editor.action.blockComment",
    "when": "editorTextFocus && !editorReadonly"
```

```
  },
  {
    "key": "ctrl+shift+a",
    "command": "-editor.action.blockComment",
    "when": "editorTextFocus && !editorReadonly"
  }
]
```

Git

Install git from git-scm.com.
If you are on Windows, open the Git Bash terminal. If you're on Mac or Linux, open a command terminal.

Type `cd` and press enter. This will put you in your User home directory. You can see the directory path with `pwd`. On Windows this will be something like `C:\Users\your-username`. On Linux it will be something like `/home/your-username`. Not sure what it is on Mac, but probably something similar to Linux.

You will need to create a file in your User home directory named .gitconfig. You can do this using VS Code by running the following command.

```
code ~/.gitconfig
```

And put the following content in it.

```
[user]
    email = your@email.com
    name = Firstname Lastname
[push]
    default = simple
    followTags = true
[core]
    editor = code
```

Python

Next you will need to create another file in your User home directory named .bashrc. You can do this again with VS Code by running the following command in your terminal.

```
code ~/.bashrc
```

And put the following content in it.

```
export PATH="$HOME/bin:$HOME/.local/bin:$PATH"
export PIPENV_VENV_IN_PROJECT=true
export PYTHONDONTWRITEBYTECODE=1
```

Now close your command terminal and re-open it so that it will read the contents of your .bashrc file.

Github Account

You will need to have a Github account in order to connect to the class assignments. We will use Github throughout the course to store and share our code.

Sign Up

Go to https://github.com/ and create an account - if you don't already have one.

You can - optionally - set an SSH key by clicking on your account logo in the top-right corner, and then go to Settings → SSH and GPG keys. There is a link there to instructions on how to create and use an SSH key.

This will allow you to interact with your Github account without needing to enter your username and password all the time.

First Assignment

Our assignments will be posted in a Github classroom. You can access the first assignment from here: https://classroom.github.com/a/LsguYrSQ

This assignment was created by Github to help you get familiar with how to use it. It's not going to be graded, but I recommend working through it if you have never used Git before.