Retro Game Building Dev Day

* Set up a GitHub repository
* Clone repository on PC in SourceTree
* Set up virtual environment on PC with correct version of Python (Anaconda Python version 2.7 for this project)
  + Important to do this as we might interfere with someone else’s project running on a different version of Python if we import libraries
  + There can be multiple versions of Python on a PC, but only one can be the default/core version
  + Don’t use the core for any projects, we use virtualenv (python library running off the core version)
  + Create a virtual environment named after the project, tell it what version of Python to use, and then activate it
  + Can pip install all libraries only in this environment, and they won’t conflict with any other virtual environment, or the core version on the PC
  + Can launch Jupyter from a virtual environment
  + Pip –freeze > requirements.txt will create a text file with the name of all the libraries installed on the virtual environment
    - Push to github and someone else can download and install the correct libraries and versions of the libraries for the project
* Type python in the cmd window to check correct version is installed/running, then ctrl+Z to exit shell
* Code goes in the git repository, libraries go in the virtual env
* Create folder on C drive called environments, which will contain projects/libraries/copy of python etc.
* Pip install virtualenv
* Cmd window, cd to environments: virtualenv nameofproject
* Cd to nameofproject
* Activate the environment: Scripts\activate.bat
* Pip install pygame
* Create a new script in notepad++ and save any scripts in the git repo folder, not the environments folder
  + Never name the file after an actual python library
* Cd into the git repo, whilst still in the virtual environment, and run the script by typing python scriptname.py
* Create execute script for the game (see run\_game.bat) so the player doesn’t have to run the command in cmd window
* Commit and push all changes in source tree so the files can be accessed on github

Pygame library documentation: <https://www.pygame.org/docs/>

Color Picker: <https://www.google.co.uk/search?q=colorpicker>

Put music in the game: <https://nerdparadise.com/programming/pygame/part3>

Explaining virtual environments and pip: <https://docs.python-guide.org/dev/virtualenvs/>

Using jupyter with virtual environments: <https://anbasile.github.io/programming/2017/06/25/jupyter-venv/>

<http://technivore.org/posts/2016/02/27/windows-jupyter-three-ways.html>