I will be doing research using this video. It is from the same group as the video I used for python research, so there should hopefully be some overlap in teachings and focus to avoid turbulence in the early-learning phase. Current time: 24:48.

NOTES

Set up beautifulsoup just by installing it in pycharm and then using 'from bs4 import BeautifulSoup'

Open html file (that is in the project directory) by simply using
with open('file.html', 'action') as variable:
Do stuff

For example:

```
from bs4 import BeautifulSoup

with open('webscrape.html', 'r') as html_file:
   content = html_file.read()
   print(content)
```

^ this (from my example code) takes the html code from the html file I made and prints it to the console

You want to download parsers, lxml seems to work. Download it by clicking python packages (at the bottom) and then searching lxml (it should be in PyPI) and then installing it.

```
from bs4 import BeautifulSoup

with open('webscrape.html', 'r') as html_file:
   content = html_file.read()
   soup = BeautifulSoup(content, 'lxml')
   print(soup.prettify())
```

^ Use BeautifulSoup to prettify the code printed to console.

```
from bs4 import BeautifulSoup

with open('webscrape.html', 'r') as html_file:
   content = html_file.read()

   soup = BeautifulSoup(content, 'lxml')
   tags = soup.find('h1') #/findAll, find_all
   print(tags)
```

^ Use BeautifulSoup to print h1s' code. Find stops after first element, findAll (or find_all) prints them all.

```
from bs4 import BeautifulSoup

with open('webscrape.html', 'r') as html_file:
   content = html_file.read()
   soup = BeautifulSoup(content, 'lxml')
   tags = soup.find_all('h1')

   for tag in tags:
       print(tag.text)
```

^ Iterate over the retrieved information to cleanly show the text itself.

```
from bs4 import BeautifulSoup

with open('google.html', 'r') as html_file2:
   content = html_file2.read()

   soup2 = BeautifulSoup(content, 'lxml')

   tags2 = soup2.find_all('div', class_='FPdoLc lJ9FBc')

   for tag in tags2:
       tag_text = tag.text
       print(tag_text.split()[0:3])

       print(f'{tag_text} is the text version, {tag} is the html version')
```

^ Extract data from (downloaded) google homepage html file.