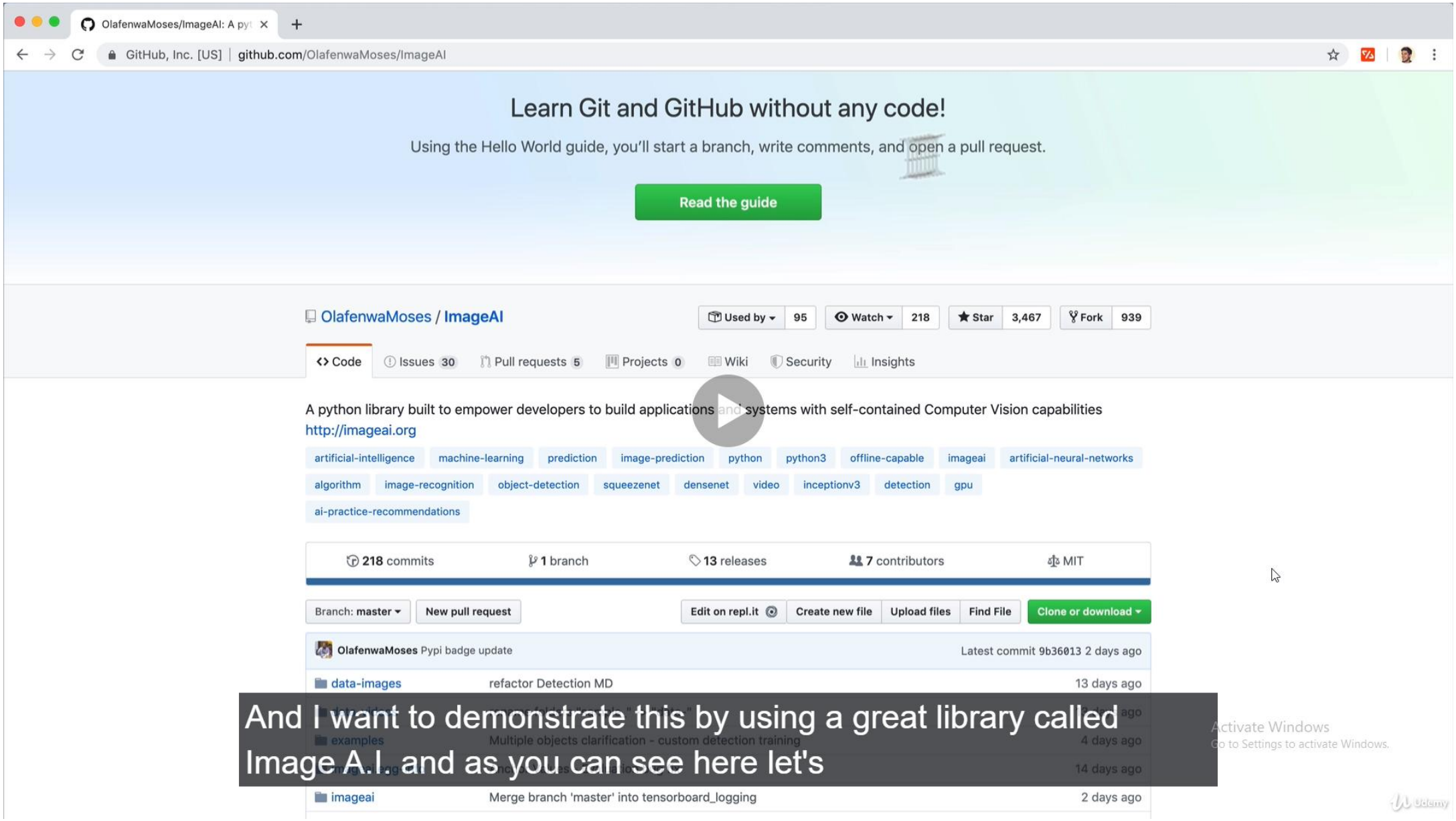


And I want to demonstrate this by using a great library called Image A.I. and as you can see here let's

OlafenwaMoses / ImageAI

Used by 95 | Watch 218 | Star 3,467 | Fork 939

Code | Issues 30 | Pull requests 5 | Projects 0 | Wiki | Security | Insights

A python library built to empower developers to build applications and systems with self-contained Computer Vision capabilities

http://imageai.org

artificial-intelligence machine-learning prediction image-prediction python python3 offline-capable imageai artificial-neural-networks algorithm image-recognition object-detection squeezenet densenet video inceptionv3 detection gpu ai-practice-recommendations

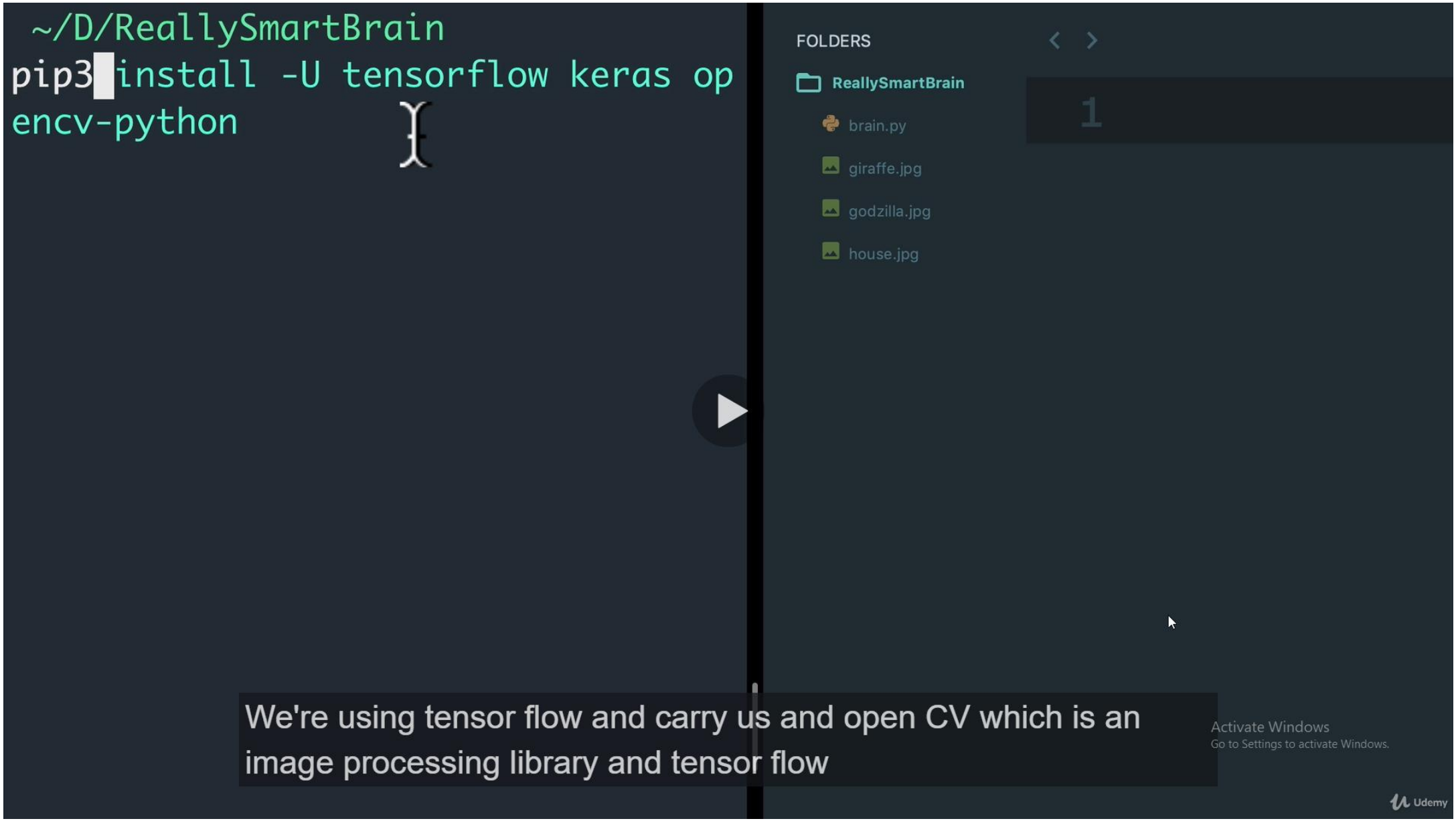
218 commits | 1 branch | 13 releases | 7 contributors | MIT

Branch: master | New pull request | Edit on repl.it | Create new file | Upload files | Find File | Clone or download

OlafenwaMoses Pypi badge update — Latest commit 9b36013 2 days ago

data-images	refactor Detection MD	13 days ago
data-videos		13 days ago
examples	Multiple objects clarification - custom detection training	4 days ago
imageai.egg-info	Anchor Values - Evaluation bug fix	14 days ago

create an entire business around a model that is freely accessible to us.


~/D/ReallySmartBrain
pip3 install -U tensorflow keras op
encv-python
FOLDERS
ReallySmartBrain
brain.py
giraffe.jpg
godzilla.jpg
house.jpg
1
We're using tensor flow and carry us and open CV which is an image processing library and tensor flow
Activate Windows
Go to Settings to activate Windows.
Udemy

- Python 3.5.1 (and later versions) (Support for Python 2.7 coming soon
- Tensorflow 1.4.0 (and later versions)
- OpenCV
- Keras 2.x

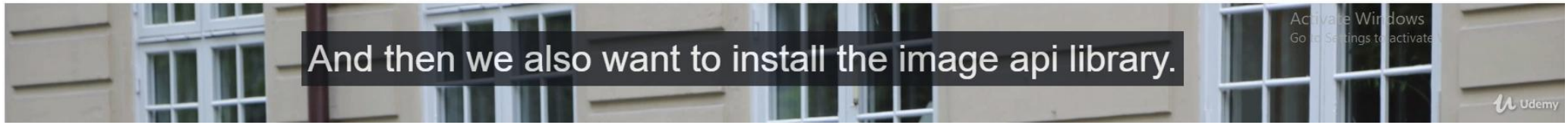
```
pip install -U tensorflow keras opencv-python
```

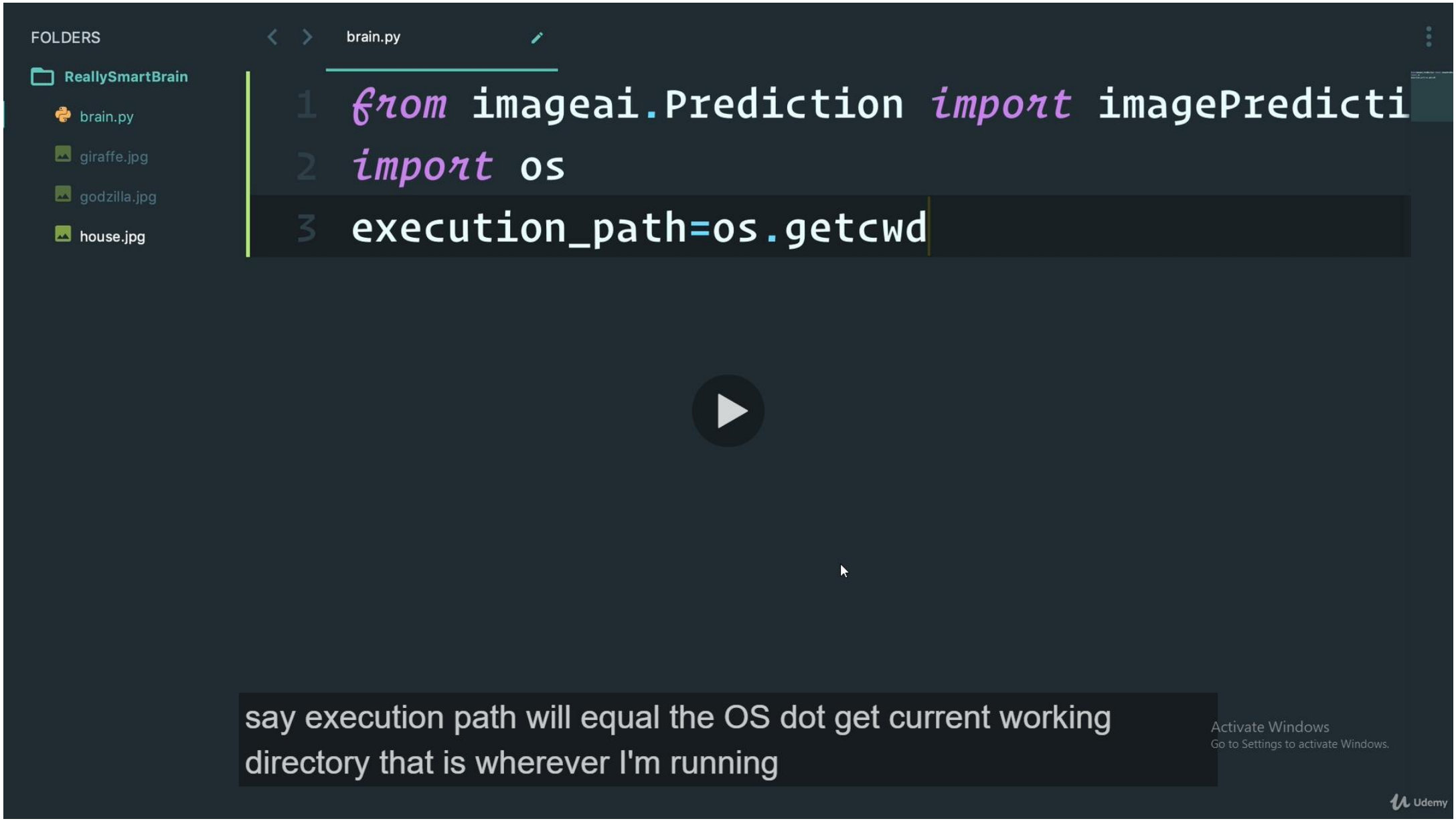
Installation

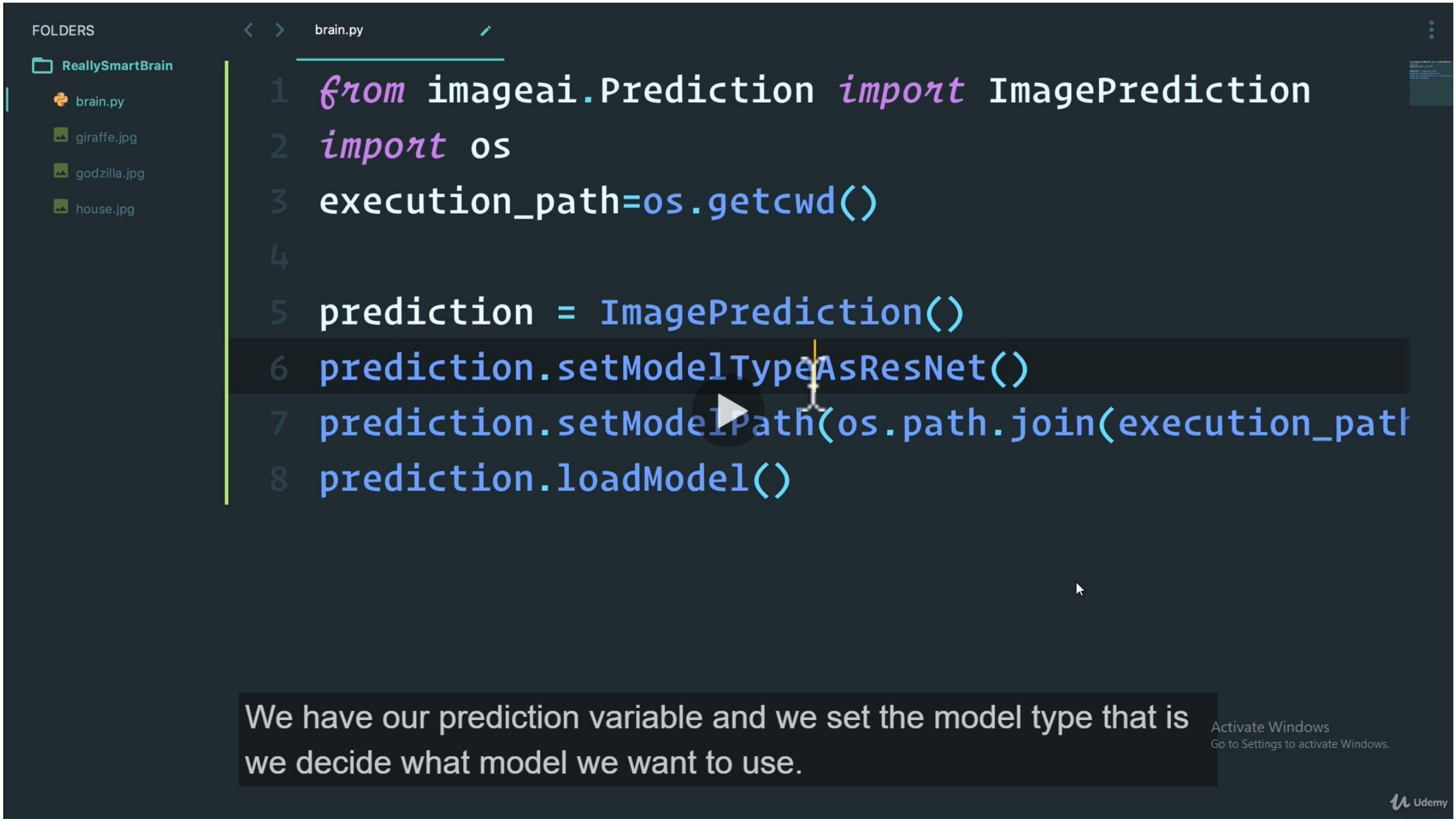
To install ImageAI, run the python installation instruction below in the command line:

```
pip3 install imageai --upgrade
```

Image Prediction




FOLDERS
brain.py
ReallySmartBrain
brain.py
giraffe.jpg
godzilla.jpg
house.jpg
1 from imageai.Prediction import imagePredicti
2 import os
3 execution_path=os.getcwd
say execution path will equal the OS dot get current working directory that is wherever I'm running
Activate Windows
Go to Settings to activate Windows.
Udemy


FOLDERS
brain.py
ReallySmartBrain
brain.py
giraffe.jpg
godzilla.jpg
house.jpg
1 from imageai.Prediction import ImagePrediction
2 import os
3 execution_path=os.getcwd()
4
5 prediction = ImagePrediction()
6 prediction.setModelTypeAsResNet()
7 prediction.setModelPath(os.path.join(execution_path
8 prediction.loadModel()
We have our prediction variable and we set the model type that is we decide what model we want to use.
Activate Windows
Go to Settings to activate Windows.
Udemy

- First Prediction
- Prediction Speed
- Image Input Types
- Multiple Images Prediction
- Prediction in MultiThreading
- Documentation

ImageAI provides 4 different algorithms and model types to perform image prediction. To perform image prediction on any picture, take the following simple steps. The 4 algorithms provided for image prediction include **SqueezeNet**, **ResNet**, **InceptionV3** and **DenseNet**. Each of these algorithms have individual model files which you must use depending on the choice of your algorithm. To download the model file for your choice of algorithm, click on any of the links below:

- SqueezeNet *(Size = 4.82 mb, fastest prediction time and moderate accuracy)*
- ResNet50 by Microsoft Research *(Size = 98 mb, fast prediction time and high accuracy)*
- InceptionV3 by Google Brain team *(Size = 91.6 mb, slow prediction time and higher accuracy)*
- DenseNet121 by Facebook AI Research *(Size = 31.6 mb, slower prediction time and highest accuracy)*

Great! Once you have downloaded this model file, start a new python project, and then copy the model file to your project folder where your python files (.py files) will be . Download the image below, or take any image on your computer and copy it to your python project's folder. Then create a python file and give it a name; an example is `FirstPrediction.py` . Then write the code below into the python file:

FirstPrediction.py

And these are really popular models that are already pre-built.

```
from imageai.Prediction import ImageClassification
import os

execution_path = os.getcwd()

prediction = ImageClassification()
```

- First Prediction
- Prediction Speed
- Image Input Types
- Multiple Images Prediction
- Prediction in MultiThreading
- Documentation

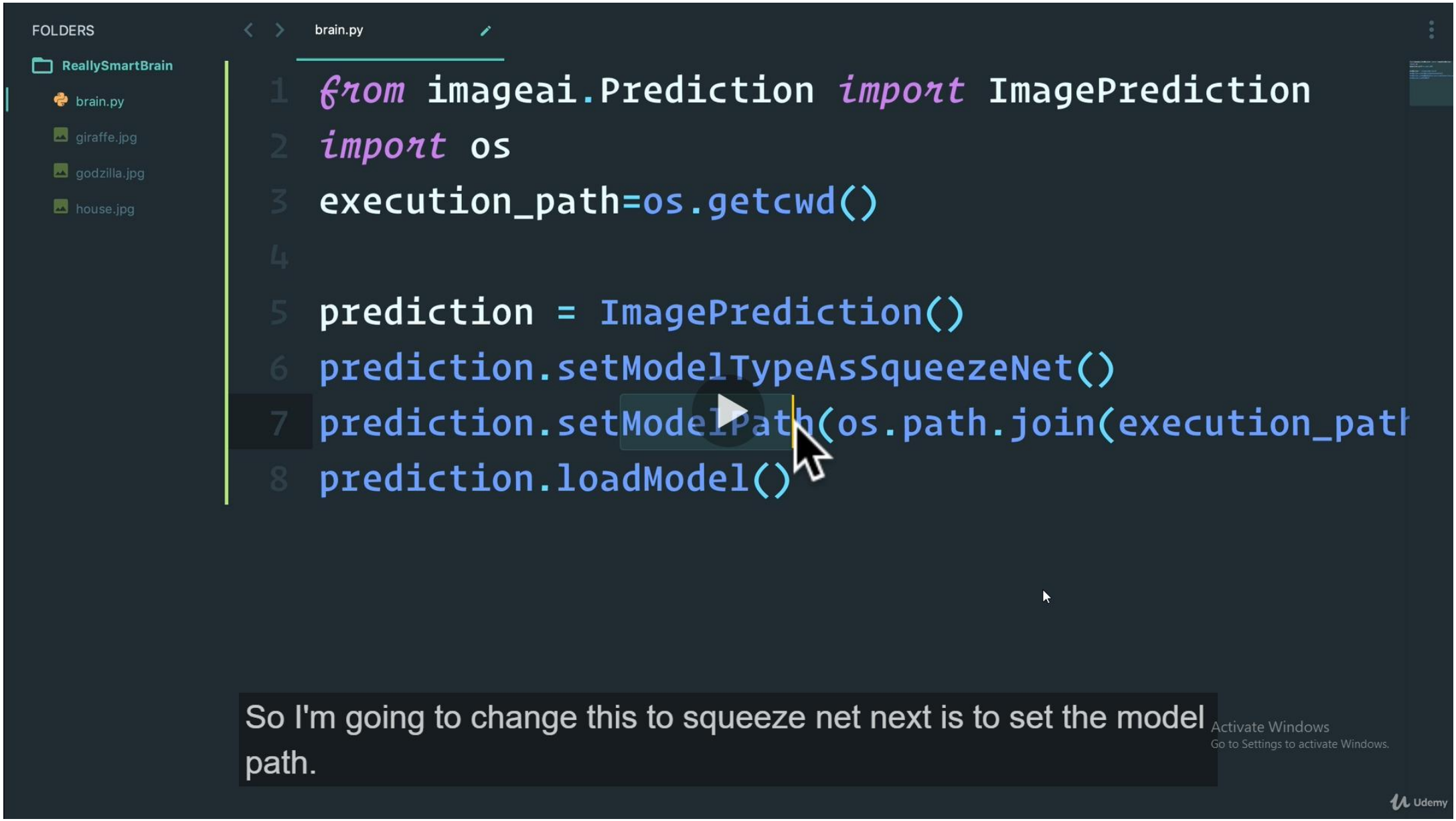
ImageAI provides 4 different algorithms and model types to perform image prediction. To perform image prediction on any picture, take the following simple steps. The 4 algorithms provided for image prediction include **SqueezeNet**, **ResNet**, **InceptionV3** and **DenseNet**. Each of these algorithms have individual model files which you must use depending on the choice of your algorithm. To download the model file for your choice of algorithm, click on any of the links below:

- **SqueezeNet** *(Size = 4.82 mb, fastest prediction time and moderate accuracy)*
- **ResNet50** by Microsoft Research *(Size = 98 mb, fast prediction time and high accuracy)*
- **InceptionV3** by Google Brain team *(Size = 91.6 mb, slow prediction time and higher accuracy)*
- **DenseNet121** by Facebook AI Research *(Size = 31.6 mb, slower prediction time and highest accuracy)*

Great! Once you have downloaded this model file, start a new python project, and then copy the model file to your project folder where your python files (.py files) will be . Download the image below, or take any image on your computer and copy it to your python project's folder. Then create a python file and give it a name; an example is `FirstPrediction.py`. Then write the code below into the python file:

model for image processing and then resonate a very famous model by Microsoft.

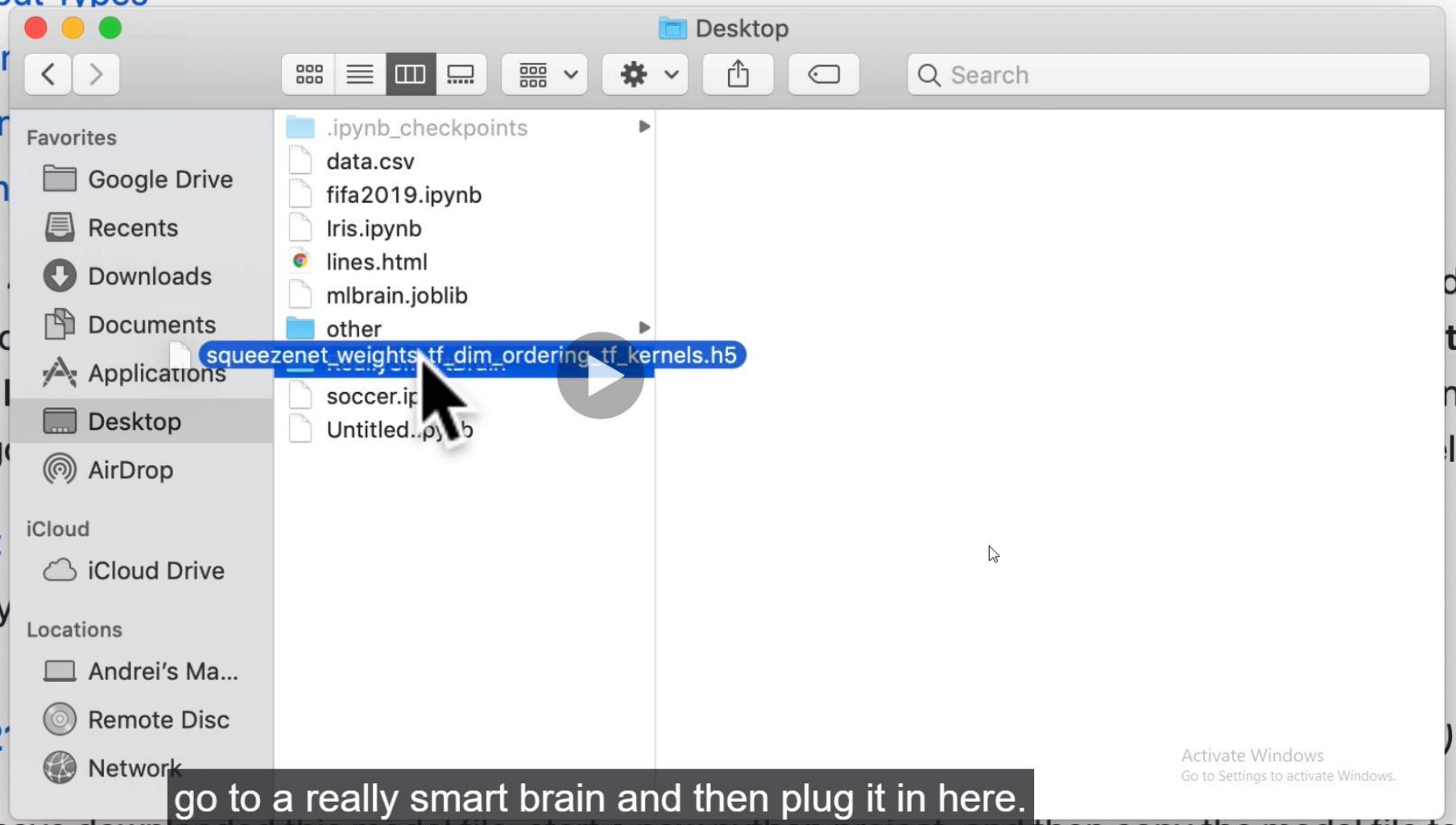
FirstPrediction.py


FOLDERS
brain.py
ReallySmartBrain
brain.py
giraffe.jpg
godzilla.jpg
house.jpg
1 from imageai.Prediction import ImagePrediction
2 import os
3 execution_path=os.getcwd()
4
5 prediction = ImagePrediction()
6 prediction.setModelTypeAsSqueezeNet()
7 prediction.setModelPath(os.path.join(execution_path
8 prediction.loadModel()
So I'm going to change this to squeeze net next is to set the model path.
Activate Windows
Go to Settings to activate Windows.
Udemy

- Prediction Speed
- Image Input Types
- Multiple Ir
- Prediction
- Documen

mageAI provides
picture, take the fo
nceptionV3 and
choice of your alg

- SqueezeNet
- ResNet50 by
- InceptionV3
- DenseNet12



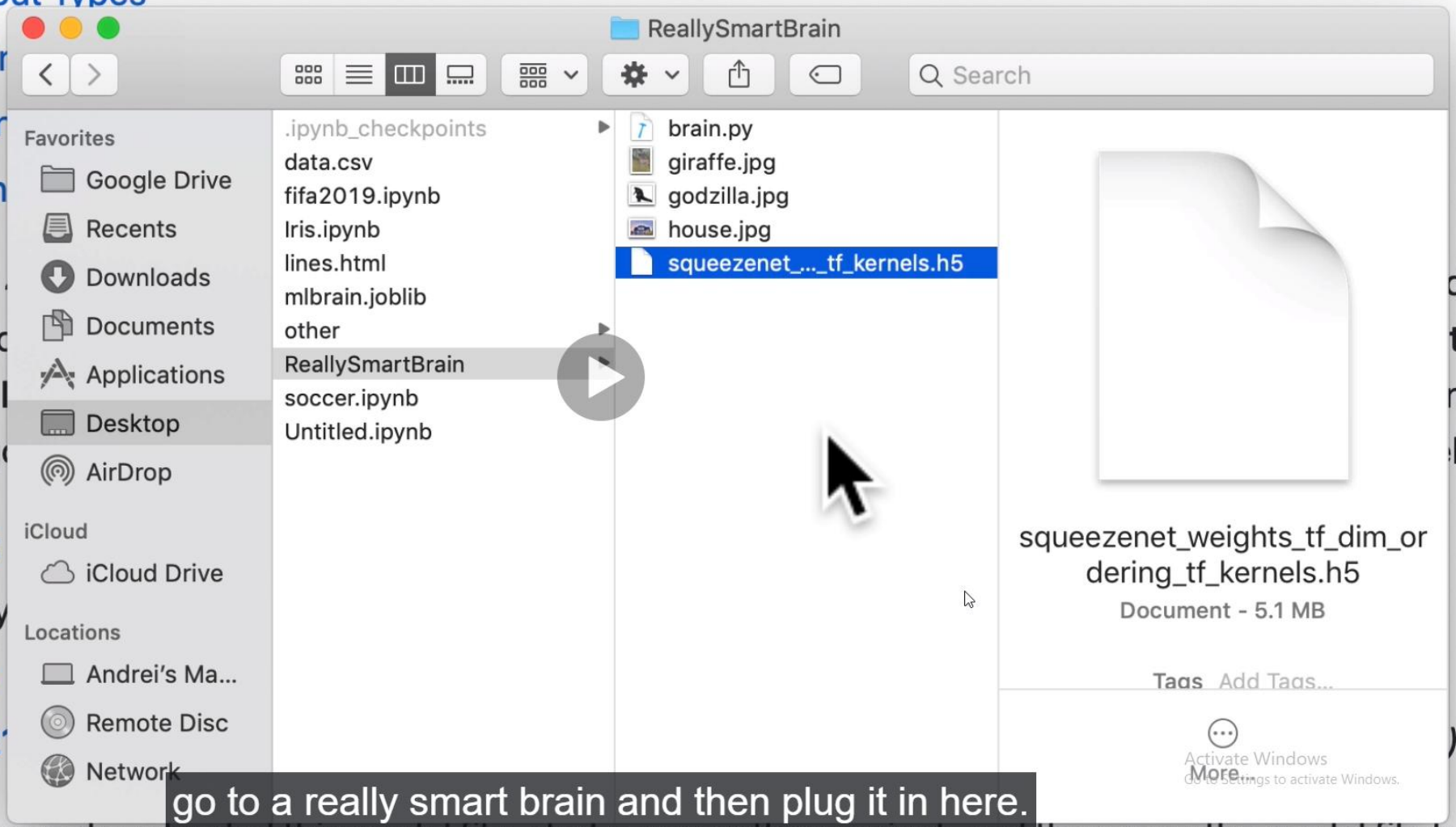
go to a really smart brain and then plug it in here.

Great! Once you have downloaded this model file, start a new python project, and then copy the model file to

- Prediction Speed
- Image Input Types
- Multiple In
- Prediction
- Documen

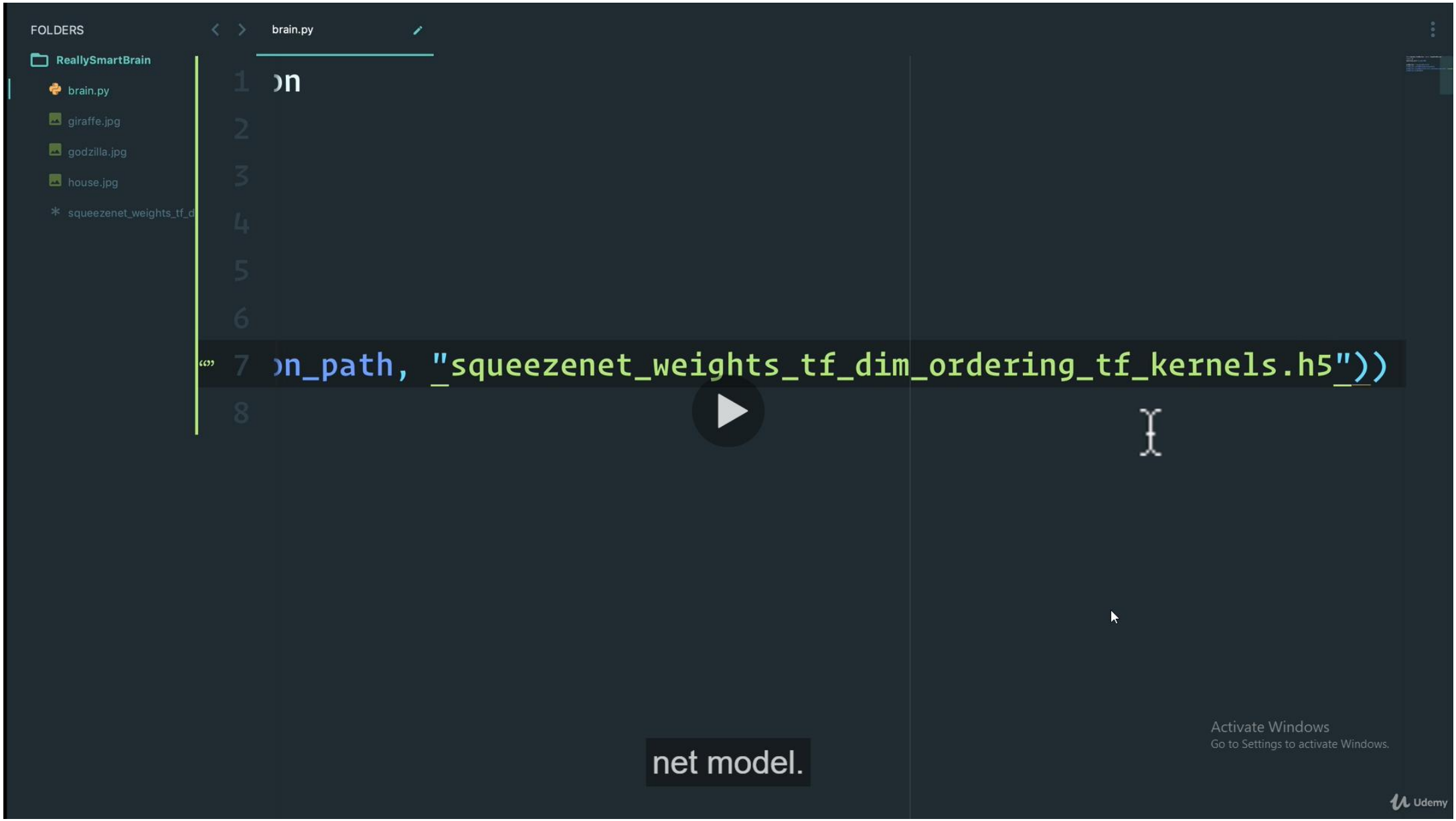
mageAI provides
picture, take the fo
nceptionV3 and
choice of your alg

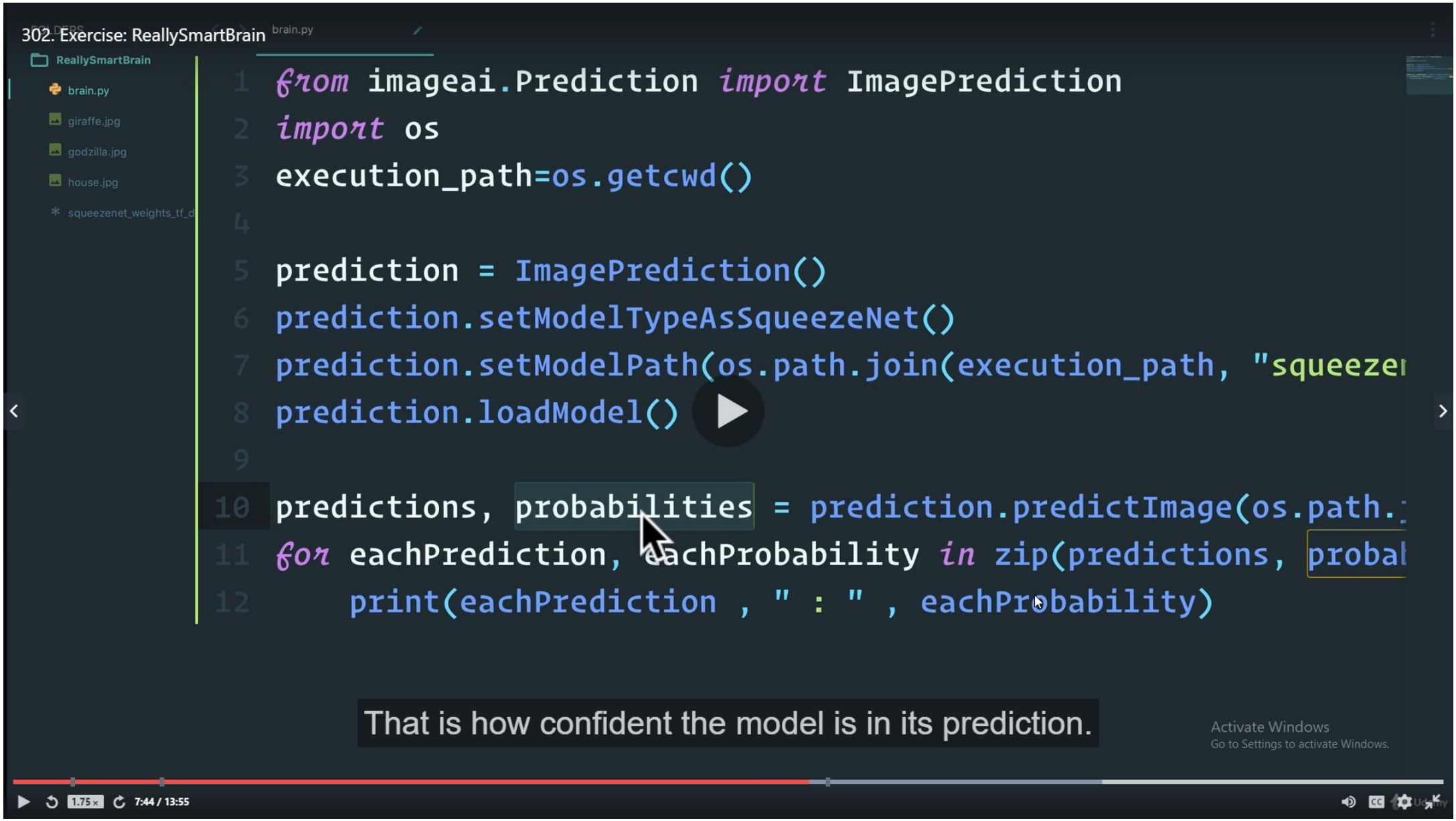
- SqueezeNet
- ResNet50 by
- InceptionV3
- DenseNet12

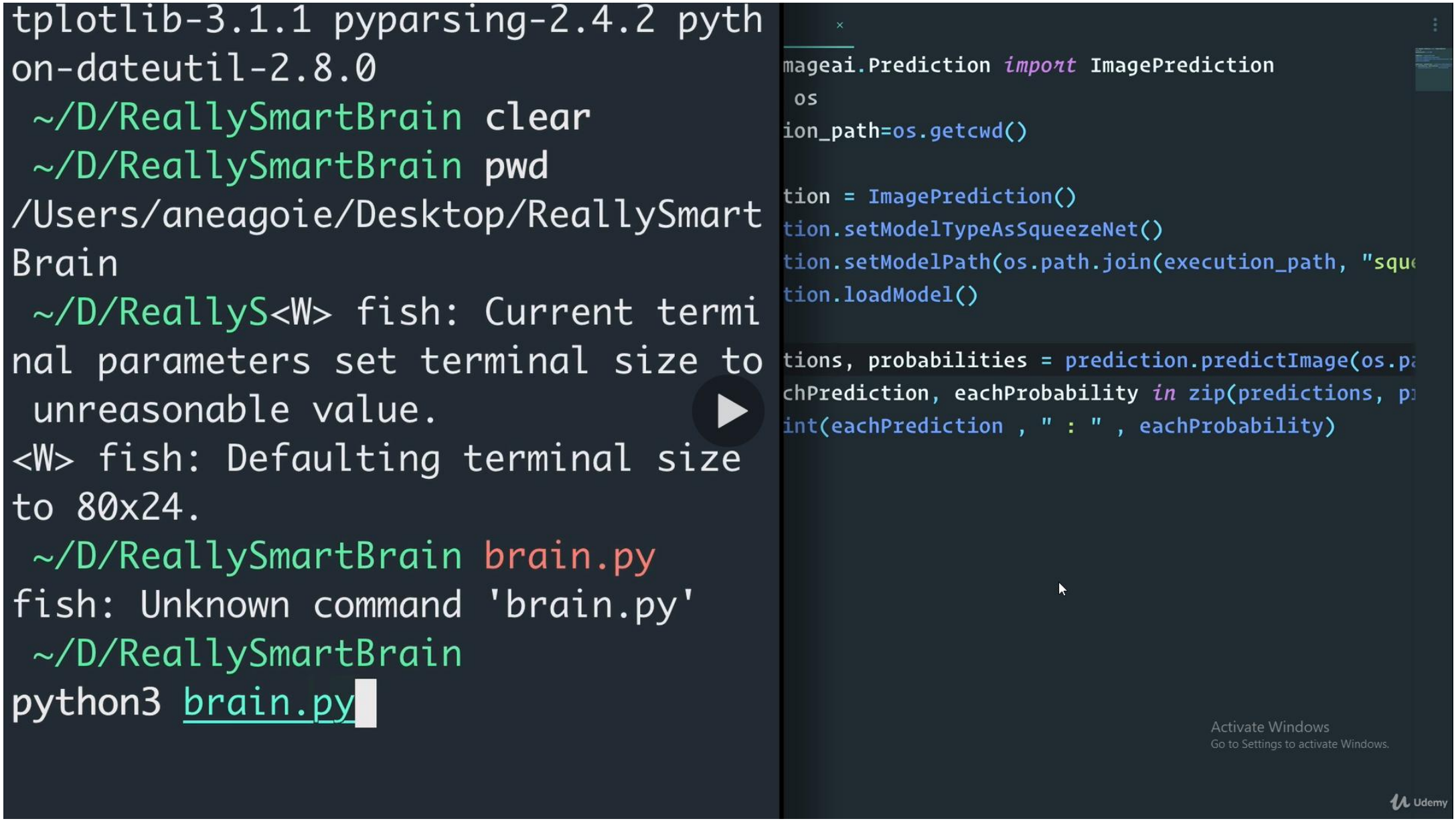


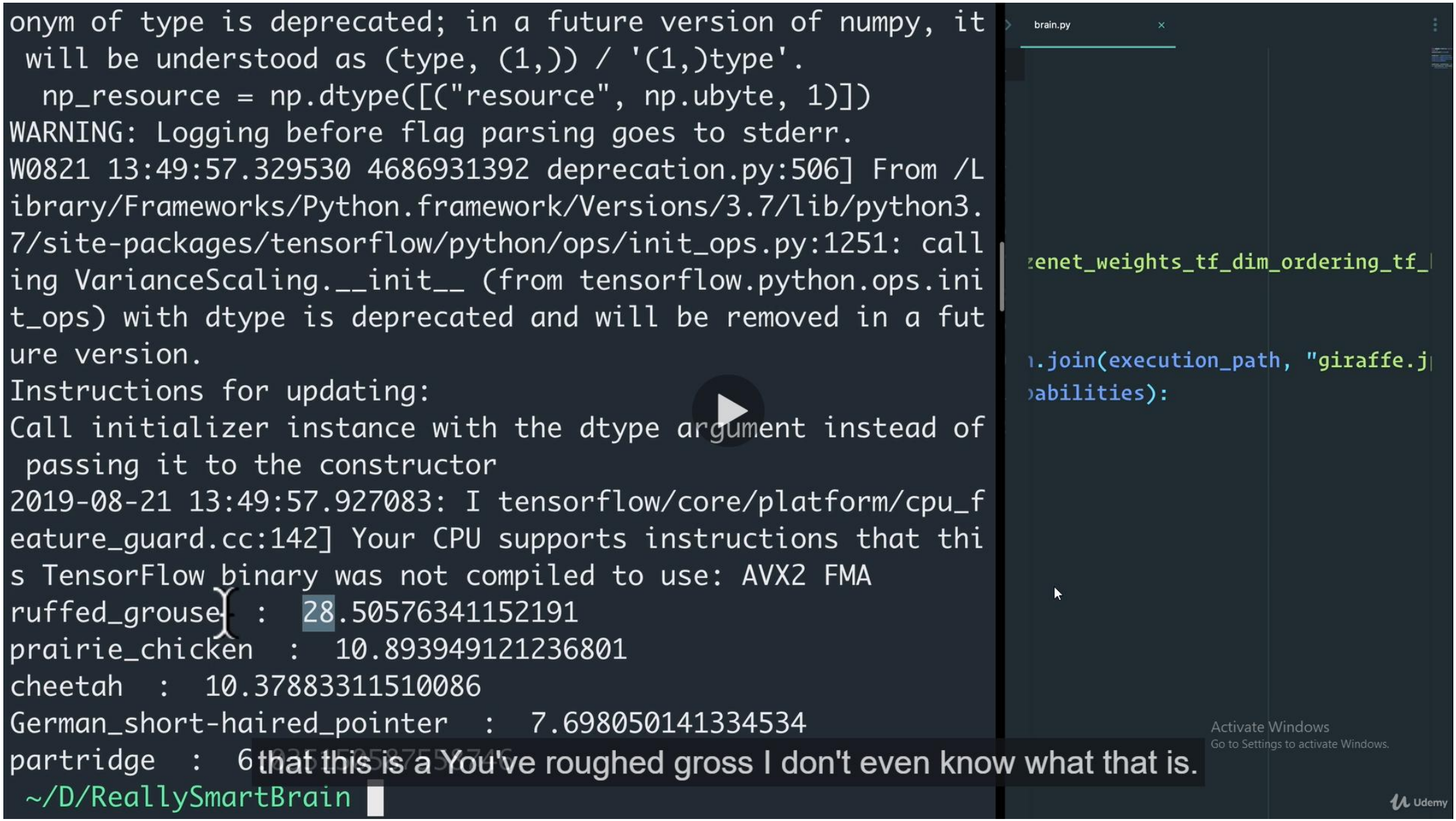
go to a really smart brain and then plug it in here.

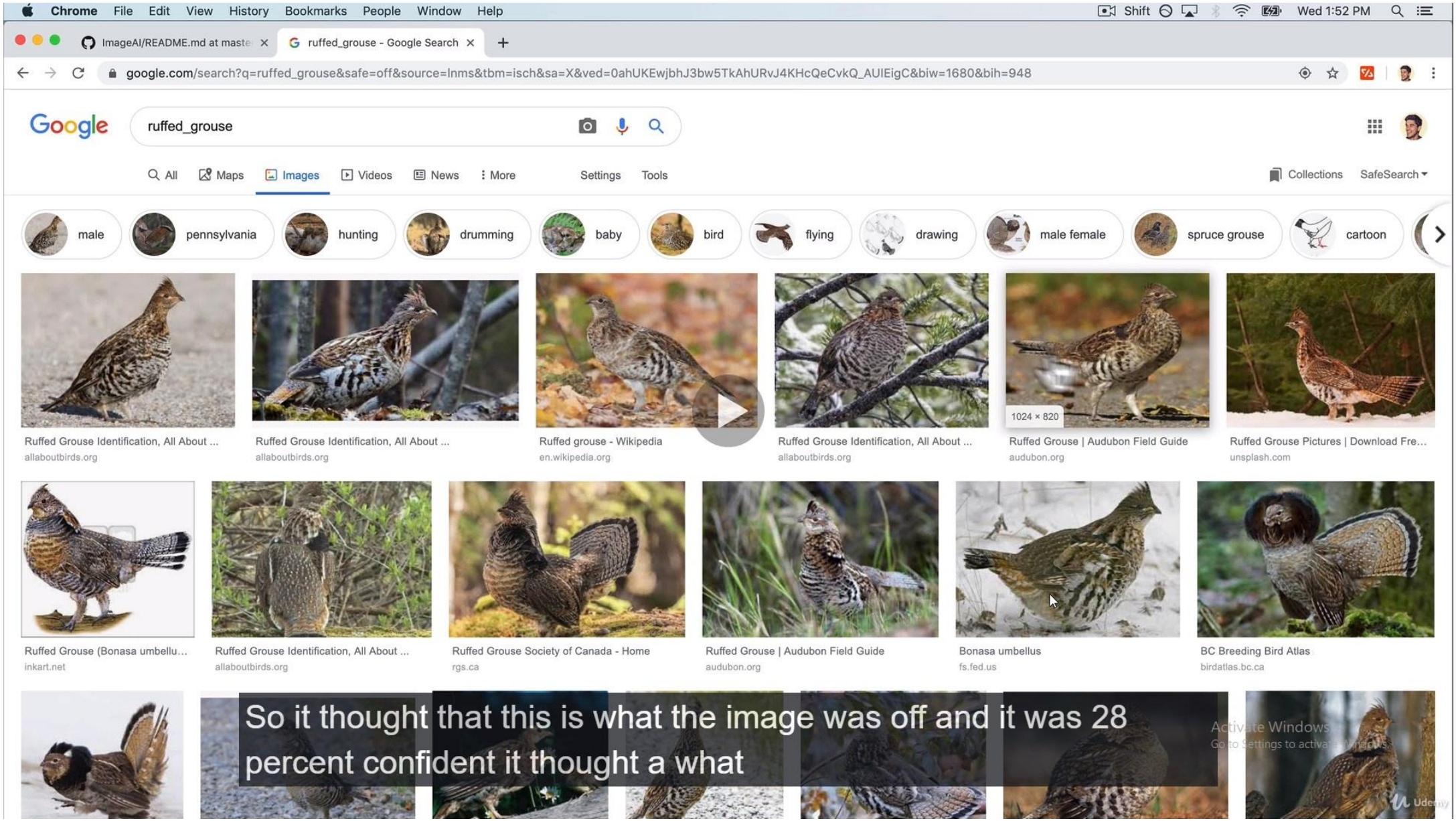
Great! Once you have downloaded this model file, start a new python project, and then copy the model file to


FOLDERS
brain.py
ReallySmartBrain
brain.py
giraffe.jpg
godzilla.jpg
house.jpg
squeezenet_weights_tf_d
1 on
7 on_path, "squeezenet_weights_tf_dim_ordering_tf_kernels.h5"))
net model.
Activate Windows
Go to Settings to activate Windows.
Udemy


302. Exercise: ReallySmartBrain
brain.py
ReallySmartBrain
brain.py
giraffe.jpg
godzilla.jpg
house.jpg
squeezenet_weights_tf_d
from imageai.Prediction import ImagePrediction
import os
execution_path=os.getcwd()
prediction = ImagePrediction()
prediction.setModelTypeAsSqueezeNet()
prediction.setModelPath(os.path.join(execution_path, "squeezer
prediction.loadModel()
predictions, probabilities = prediction.predictImage(os.path.
for eachPrediction, eachProbability in zip(predictions, probal
print(eachPrediction , " : " , eachProbability)
That is how confident the model is in its prediction.
Activate Windows
Go to Settings to activate Windows.
1.75x
7:44 / 13:55


tplotlib-3.1.1 pyparsing-2.4.2 pyth
on-dateutil-2.8.0
~/D/ReallySmartBrain clear
~/D/ReallySmartBrain pwd
/Users/aneagoie/Desktop/ReallySmart
Brain
~/D/ReallyS<W> fish: Current termi
nal parameters set terminal size to
unreasonable value.
<W> fish: Defaulting terminal size
to 80x24.
~/D/ReallySmartBrain brain.py
fish: Unknown command 'brain.py'
~/D/ReallySmartBrain
python3 brain.py
mageai.Prediction import ImagePrediction
os
ion_path=os.getcwd()
tion = ImagePrediction()
tion.setModelTypeAsSqueezeNet()
tion.setModelPath(os.path.join(execution_path, "squ
tion.loadModel()
tions, probabilities = prediction.predictImage(os.p
chPrediction, eachProbability in zip(predictions, p
int(eachPrediction , " : " , eachProbability)
Activate Windows
Go to Settings to activate Windows.
Udemy


onym of type is deprecated; in a future version of numpy, it will be understood as (type, (1,)) / '(1,)type'.
np_resource = np.dtype([("resource", np.ubyte, 1)])
WARNING: Logging before flag parsing goes to stderr.
W0821 13:49:57.329530 4686931392 deprecation.py:506] From /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages/tensorflow/python/ops/init_ops.py:1251: calling VarianceScaling.__init__ (from tensorflow.python.ops.init_ops) with dtype is deprecated and will be removed in a future version.
Instructions for updating:
Call initializer instance with the dtype argument instead of passing it to the constructor
2019-08-21 13:49:57.927083: I tensorflow/core/platform/cpu_feature_guard.cc:142] Your CPU supports instructions that this TensorFlow binary was not compiled to use: AVX2 FMA
ruffed_grouse : 28.50576341152191
prairie_chicken : 10.893949121236801
cheetah : 10.37883311510086
German_short-haired_pointer : 7.698050141334534
partridge : 6
~/D/ReallySmartBrain
brain.py
enet_weights_tf_dim_ordering_tf_
.join(execution_path, "giraffe.j
abilities):
that this is a You've roughed gross I don't even know what that is.
Activate Windows
Go to Settings to activate Windows.
Udemy


Chrome File Edit View History Bookmarks People Window Help
Wed 1:52 PM
ImageAI/README.md at maste
ruffed_grouse - Google Search
google.com/search?q=ruffed_grouse&safe=off&source=lnms&tbm=isch&sa=X&ved=0ahUKEwjbhJ3bw5TkAhURvJ4KHcQeCvkQ_AUIEigC&biw=1680&bih=948
Google
ruffed_grouse
All Maps Images Videos News More Settings Tools
Collections SafeSearch
male pennsylvania hunting drumming baby bird flying drawing male female spruce grouse cartoon
Ruffed Grouse Identification, All About ...
allaboutbirds.org
Ruffed Grouse Identification, All About ...
allaboutbirds.org
Ruffed grouse - Wikipedia
en.wikipedia.org
Ruffed Grouse Identification, All About ...
allaboutbirds.org
1024 × 820
Ruffed Grouse | Audubon Field Guide
audubon.org
Ruffed Grouse Pictures | Download Fre...
unsplash.com
Ruffed Grouse (Bonasa umbellu...
inkart.net
Ruffed Grouse Identification, All About ...
allaboutbirds.org
Ruffed Grouse Society of Canada - Home
rgs.ca
Ruffed Grouse | Audubon Field Guide
audubon.org
Bonasa umbellus
fs.fed.us
BC Breeding Bird Atlas
birdatlas.bc.ca
So it thought that this is what the image was off and it was 28 percent confident it thought a what
Activate Windows
Go to Settings to activate Windows.

ImageAI/README.md at maste | ruffed_grouse - Google Search

GitHub, Inc. [US] | github.com/OlafenwaMoses/ImageAI/blob/master/imageai/Prediction/README.md

ImageAI : Image Prediction

A **DeepQuest AI** project https://deepquestai.com

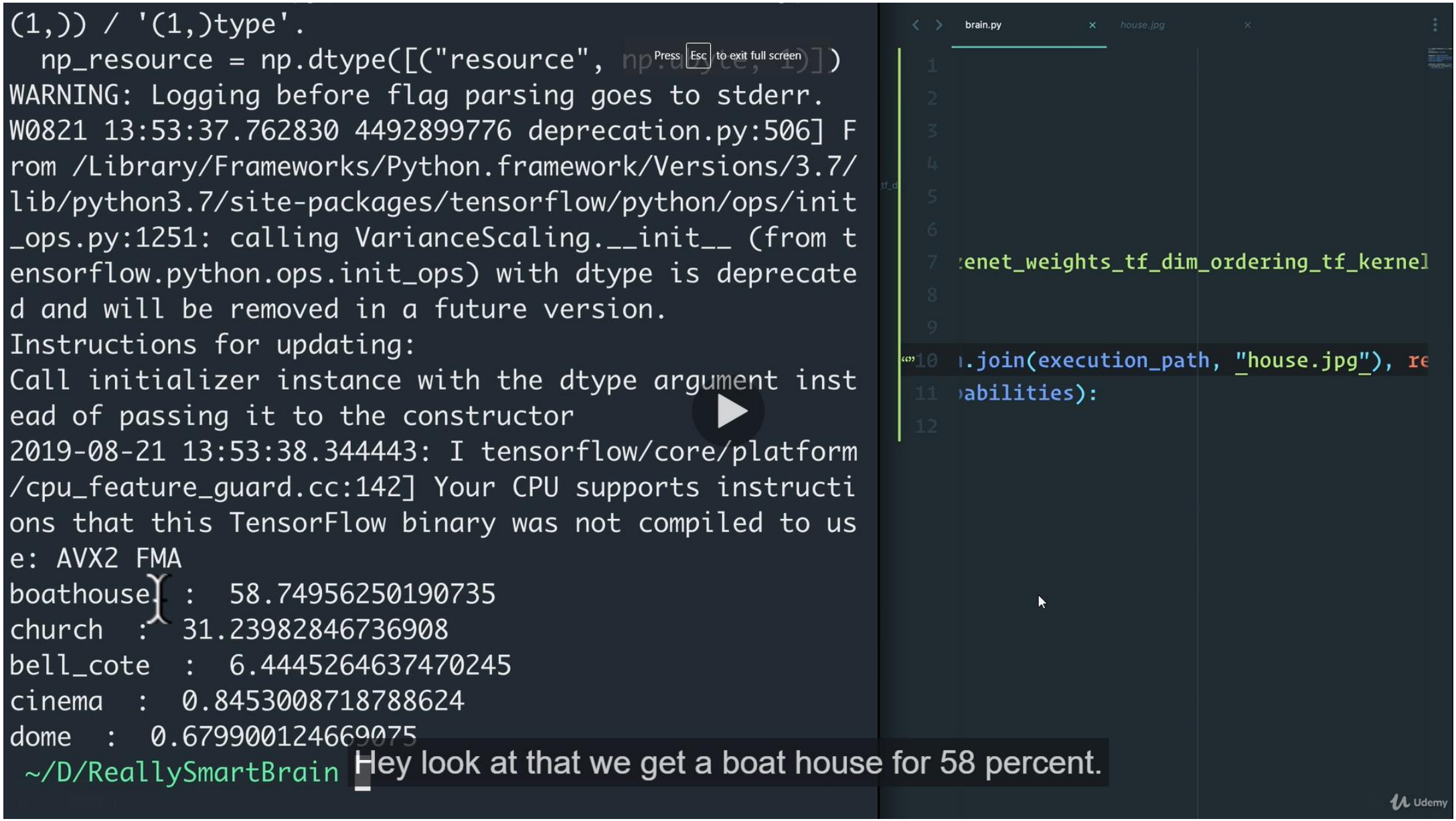
TABLE OF CONTENTS

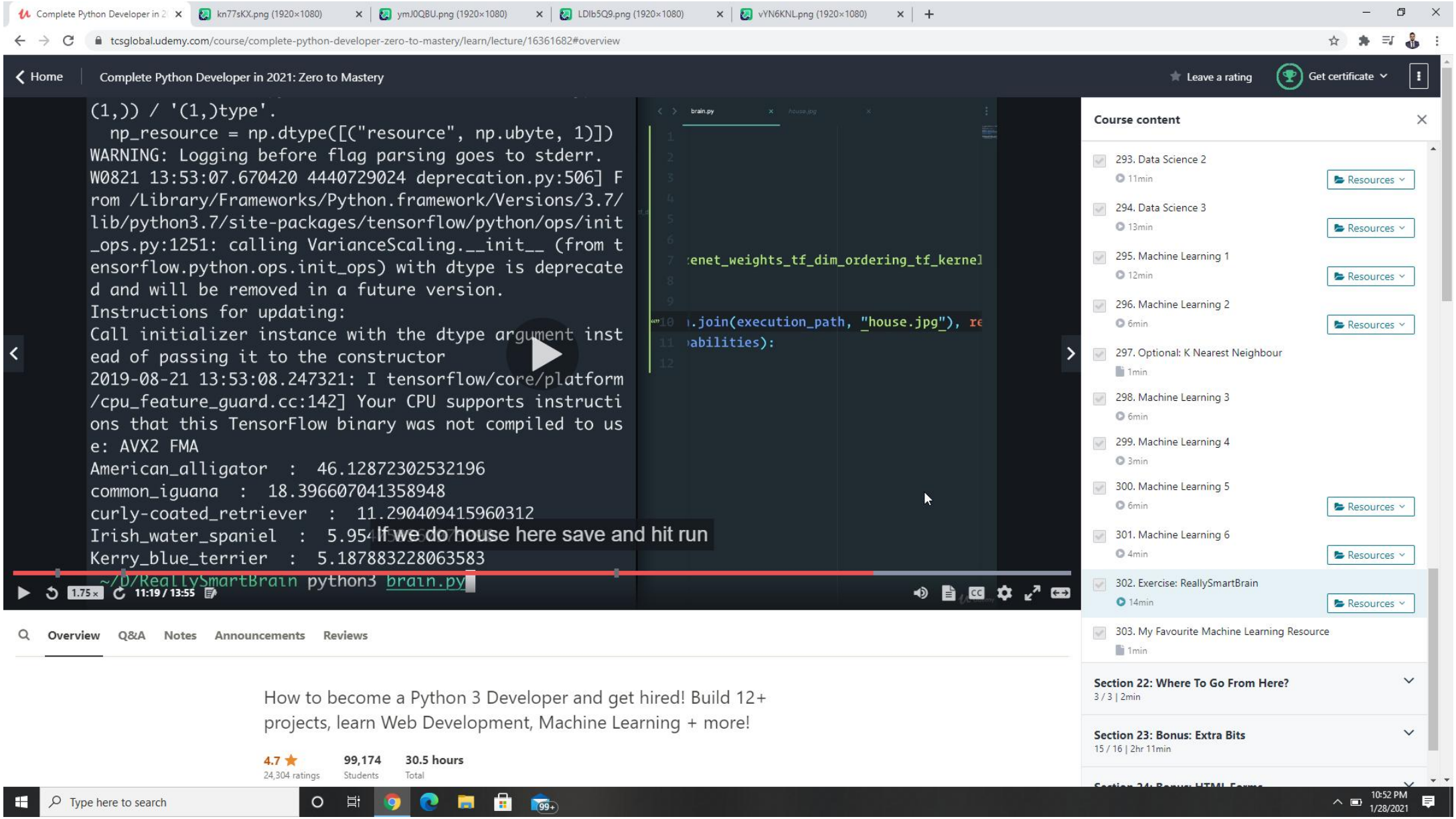
ImageAI provides 4 different algorithms and model types to perform image prediction. To perform image prediction on any picture, take the following simple steps. The 4 algorithms provided for image prediction include **SqueezeNet**, **ResNet**, **InceptionV3** and **DenseNet**. Each of these algorithms have individual model files which you must use depending on the choice of your algorithm. To download the model file for your choice of algorithm, click on any of the links below:

- **SqueezeNet** *(Size = 4.82 mb, fastest prediction time and moderate accuracy)*
- **ResNet50** by Microsoft Research *(Size = 98 mb, fast prediction time and high accuracy)*
- **InceptionV3** by Google Brain team *(Size = 91.6 mb, slow prediction time and higher accuracy)*
- **DenseNet** by Facebook AI Research *(Size = 31.6 mb, slow prediction time and highest accuracy)*

There are so many libraries so many tools that are given by the community to everybody who uses python.

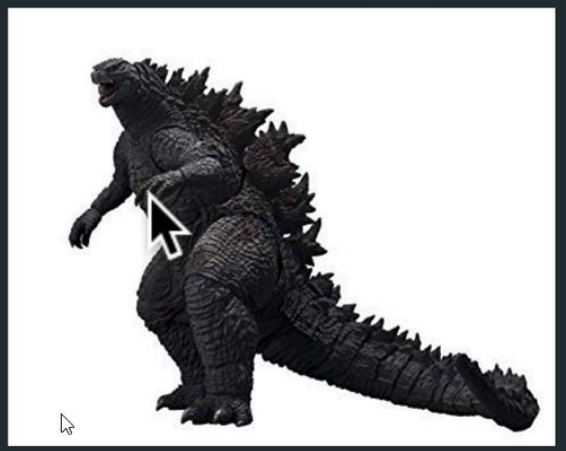
Great! Once you have downloaded this model file, start a new python project, and then copy the model file to your project


(1,)) / '(1,)type'.
np_resource = np.dtype([("resource",
Press Esc to exit full screen
WARNING: Logging before flag parsing goes to stderr.
W0821 13:53:37.762830 4492899776 deprecation.py:506] From /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages/tensorflow/python/ops/init_ops.py:1251: calling VarianceScaling.__init__ (from tensorflow.python.ops.init_ops) with dtype is deprecated and will be removed in a future version.
Instructions for updating:
Call initializer instance with the dtype argument instead of passing it to the constructor
2019-08-21 13:53:38.344443: I tensorflow/core/platform/cpu_feature_guard.cc:142] Your CPU supports instructions that this TensorFlow binary was not compiled to use: AVX2 FMA
boathouse : 58.74956250190735
church : 31.23982846736908
bell_cote : 6.4445264637470245
cinema : 0.84530087187886624
dome : 0.6799001246690075
~/D/ReallySmartBrain
brain.py
house.jpg
enet_weights_tf_dim_ordering_tf_kernel
.join(execution_path, "house.jpg"), re
abilities):
Hey look at that we get a boat house for 58 percent.
Udemy


Complete Python Developer in 2
kn77sKX.png (1920×1080)
ymJ0QBU.png (1920×1080)
LDIb5Q9.png (1920×1080)
vYN6KNL.png (1920×1080)
tcsglobal.udemy.com/course/complete-python-developer-zero-to-mastery/learn/lecture/16361682#overview
Home
Complete Python Developer in 2021: Zero to Mastery
Leave a rating
Get certificate
(1,)) / '(1,)type'.
np_resource = np.dtype([("resource", np.ubyte, 1)])
WARNING: Logging before flag parsing goes to stderr.
W0821 13:53:07.670420 4440729024 deprecation.py:506] From /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-packages/tensorflow/python/ops/init_ops.py:1251: calling VarianceScaling.__init__ (from tensorflow.python.ops.init_ops) with dtype is deprecated and will be removed in a future version.
Instructions for updating:
Call initializer instance with the dtype argument instead of passing it to the constructor
2019-08-21 13:53:08.247321: I tensorflow/core/platform/cpu_feature_guard.cc:142] Your CPU supports instructions that this TensorFlow binary was not compiled to use: AVX2 FMA
American_alligator : 46.12872302532196
common_iguana : 18.396607041358948
curly-coated_retriever : 11.290409415960312
Irish_water_spaniel : 5.954
Kerry_blue_terrier : 5.187883228063583
~/D/ReallySmartBrain python3 brain.py
brain.py
house.jpg
enet_weights_tf_dim_ordering_tf_kernel
.join(execution_path, "house.jpg"), re
abilities):
If we do house here save and hit run
1.75×
11:19 / 13:55
Course content
293. Data Science 2
11min
Resources
294. Data Science 3
13min
Resources
295. Machine Learning 1
12min
Resources
296. Machine Learning 2
6min
Resources
297. Optional: K Nearest Neighbour
1min
298. Machine Learning 3
6min
299. Machine Learning 4
3min
300. Machine Learning 5
6min
Resources
301. Machine Learning 6
4min
Resources
302. Exercise: ReallySmartBrain
14min
Resources
303. My Favourite Machine Learning Resource
1min
Section 22: Where To Go From Here?
3 / 3 | 2min
Section 23: Bonus: Extra Bits
15 / 16 | 2hr 11min
Overview
Q&A
Notes
Announcements
Reviews
How to become a Python 3 Developer and get hired! Build 12+ projects, learn Web Development, Machine Learning + more!
4.7
24,304 ratings
99,174
Students
30.5 hours
Total
Type here to search
10:52 PM
1/28/2021

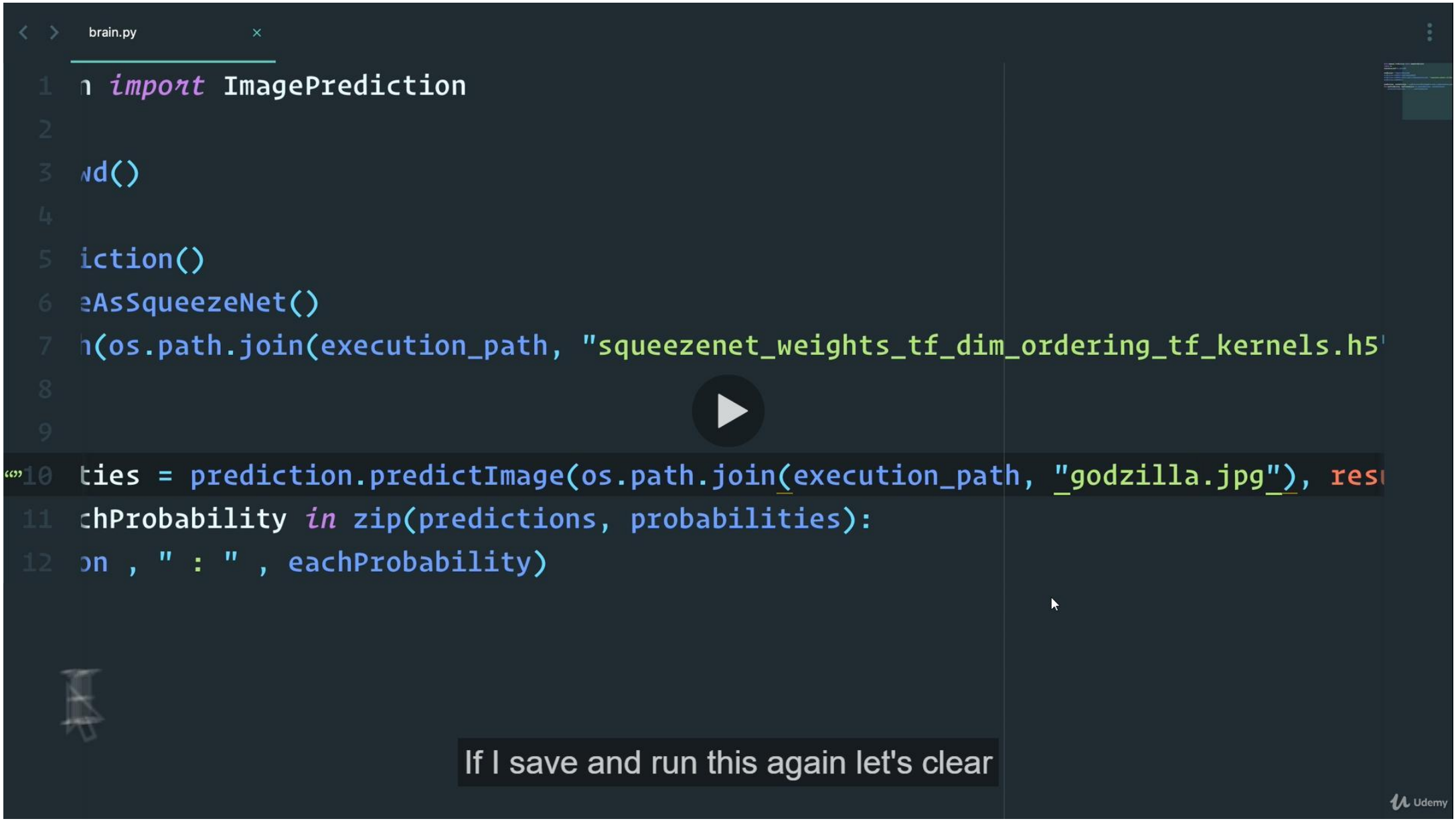
302. Exercise: ReallySmartBrain

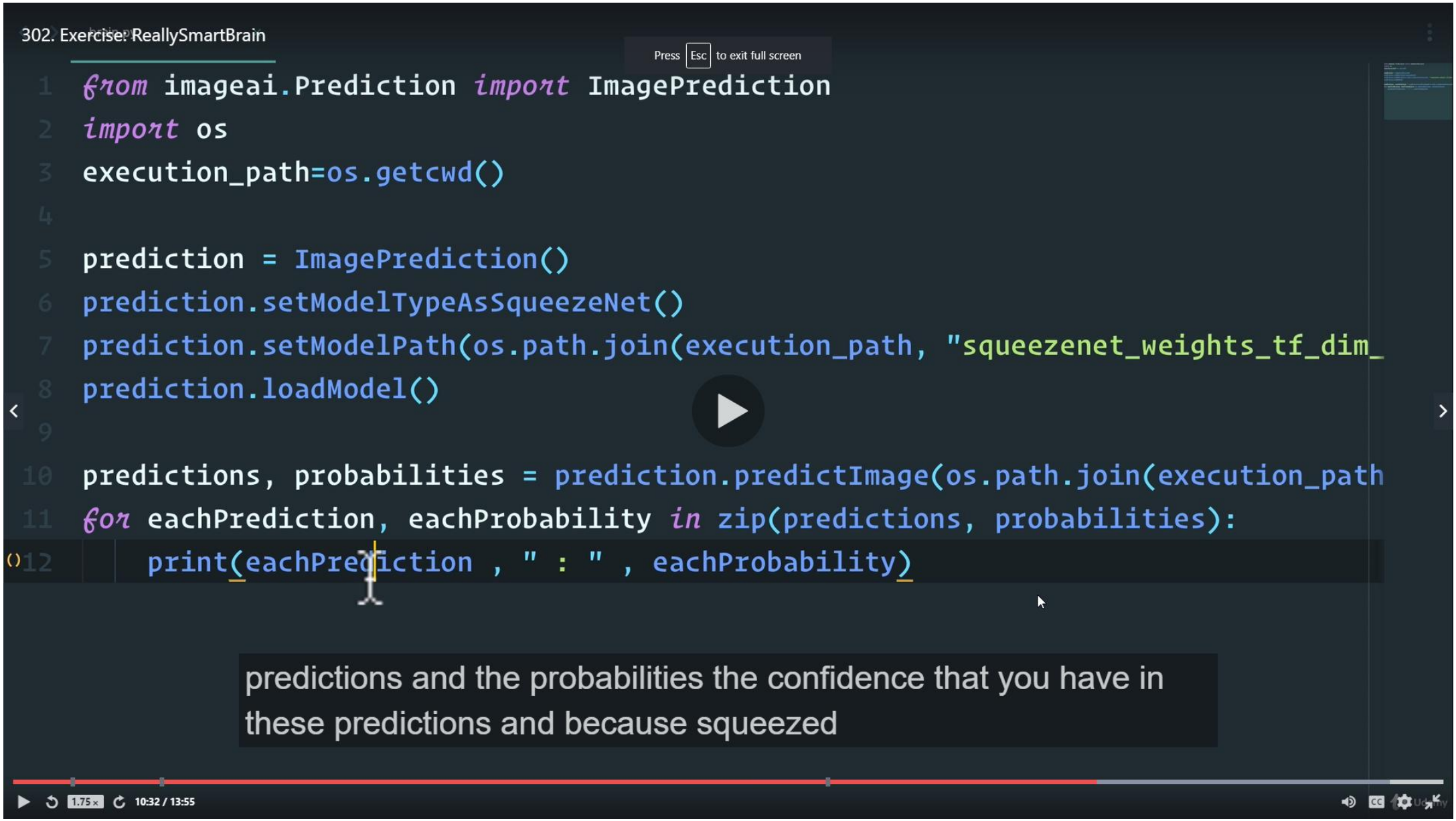
```
  np_resource = np.dtype([("resource", np.ubyte, 1)])
WARNING: Logging before flag parsing goes to stderr.
W0821 13:53:07.670420 4440729024 deprecation.py:506] F
rom /Library/Frameworks/Python.framework/Versions/3.7/
lib/python3.7/site-packages/tensorflow/python/ops/init
_ops.py:1251: calling VarianceScaling.__init__ (from t
ensorflow.python.ops.init_ops) with dtype is deprecate
d and will be removed in a future version.
Instructions for updating:
Call initializer instance with the dtype argument inst
ead of passing it to the constructor
2019-08-21 13:53:08.247321: I tensorflow/core/platform
/cpu_feature_guard.cc:142] Your CPU supports instructi
ons that this TensorFlow binary was not compiled to us
e: AVX2 FMA
American_alligator  :  46.12872302532196
common_iguana  :  18.396607041358948
curly-coated_retriever  :  11.290409415960312
Irish_water_spaniel  :  5.9547595679759 98
Kerry_blue_terrier  :  5.18788322 8065583
 ~/D/ReallySmartBrain
```

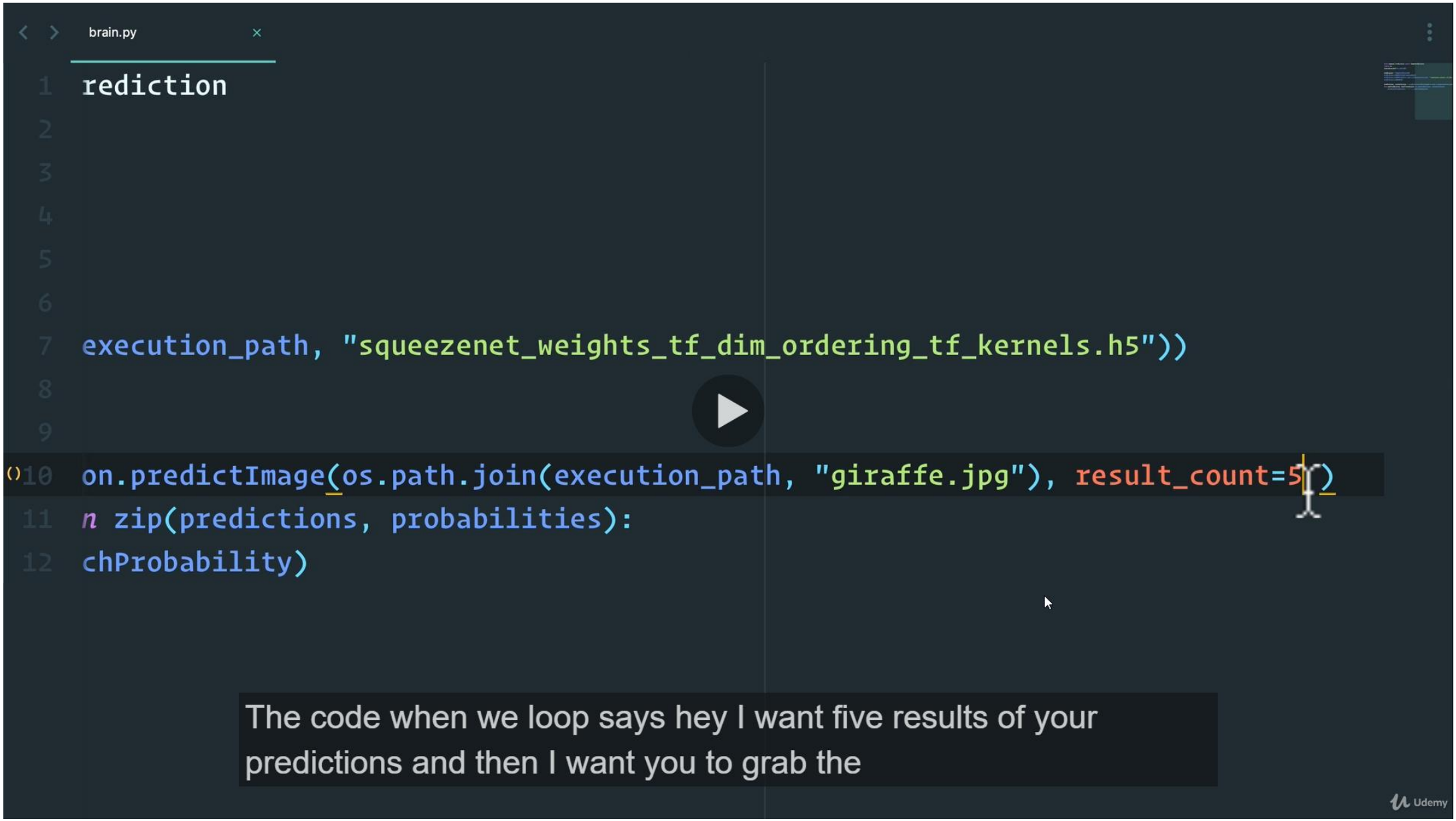


That's not too bad actually.

1.75× 11:07 / 13:55


brain.py
n import ImagePrediction
ıd()
iction()
eAsSqueezeNet()
h(os.path.join(execution_path, "squeezenet_weights_tf_dim_ordering_tf_kernels.h5
ties = prediction.predictImage(os.path.join(execution_path, "godzilla.jpg"), res
chProbability in zip(predictions, probabilities):
on , " : " , eachProbability)
If I save and run this again let's clear
Udemy


302. Exercise: ReallySmartBrain
Press Esc to exit full screen
from imageai.Prediction import ImagePrediction
import os
execution_path=os.getcwd()
prediction = ImagePrediction()
prediction.setModelTypeAsSqueezeNet()
prediction.setModelPath(os.path.join(execution_path, "squeezenet_weights_tf_dim_
prediction.loadModel()
predictions, probabilities = prediction.predictImage(os.path.join(execution_path
for eachPrediction, eachProbability in zip(predictions, probabilities):
print(eachPrediction , " : " , eachProbability)
predictions and the probabilities the confidence that you have in these predictions and because squeezed
1.75×
10:32 / 13:55


brain.py
rediction
execution_path, "squeezenet_weights_tf_dim_ordering_tf_kernels.h5"))
on.predictImage(os.path.join(execution_path, "giraffe.jpg"), result_count=5)
n zip(predictions, probabilities):
chProbability)
The code when we loop says hey I want five results of your predictions and then I want you to grab the
Udemy