```python
# Find faces in picture
# https://github.com/ageitgey/face_recognition/blob/master/examples/find_faces_in_picture.py

from PIL import Image
import face_recognition

# Load the jpg file into a numpy array
image = face_recognition.load_image_file("office.jpg")

# Find all the faces in the image using the default HOG-based model.
# This method is fairly accurate, but not as accurate as the CNN model and not GPU accelerated.
# See also: find_faces_in_picture_cnn.py
face_locations = face_recognition.face_locations(image)

for face_location in face_locations:

    # Print the location of each face in this image
    top, right, bottom, left = face_location

    # You can access the actual face itself like this:
    face_image = image[top:bottom, left:right]
    pil_image = Image.fromarray(face_image)
    pil_image.show()
```

```python
# Identify and draw box on David
# https://github.com/ageitgey/face_recognition/blob/master/examples/identify_and_draw_boxes_on_faces.py

import face_recognition
import numpy as np
from PIL import Image, ImageDraw

# Load a sample picture and learn how to recognize it.
known_image = face_recognition.load_image_file("toby.jpg")
encoding = face_recognition.face_encodings(known_image)[0]

# Load an image with unknown faces
unknown_image = face_recognition.load_image_file("office.jpg")

# Find all the faces and face encodings in the unknown image
face_locations = face_recognition.face_locations(unknown_image)
face_encodings = face_recognition.face_encodings(unknown_image, face_locations)

# Convert the image to a PIL-format image so that we can draw on top of it with the Pillow library
# See http://pillow.readthedocs.io/ for more about PIL/Pillow
pil_image = Image.fromarray(unknown_image)

# Create a Pillow ImageDraw Draw instance to draw with
draw = ImageDraw.Draw(pil_image)

# Loop through each face found in the unknown image
for (top, right, bottom, left), face_encoding in zip(face_locations, face_encodings):

    # See if the face is a match for the known face(s)
    matches = face_recognition.compare_faces([encoding], face_encoding)

    # Use the known face with the smallest distance to the new face
    face_distances = face_recognition.face_distance([encoding], face_encoding)
    best_match_index = np.argmin(face_distances)
    if matches[best_match_index]:

        # Draw a box around the face using the Pillow module
        draw.rectangle(((left - 20, top - 20), (right + 20, bottom + 20)), outline=(0, 255, 0), width=20)

# Remove the drawing library from memory as per the Pillow docs
del draw

```

```
43  # Display the resulting image
44  pil_image.show()
```

```python
# Demonstrates a function with a positional argument

print("hello, world")
```

```python
# Demonstrates concatenation of strings

name = input("What's your name? ")
print("hello, " + name)
```

```python
# Demonstrates a function with two positional arguments

name = input("What's your name? ")
print("hello,", name)
```

```python
# Demonstrates a format string

name = input("What's your name? ")
print(f"hello, {name}")
```

```python
# Demonstrates str functions

name = input("What's your name? ")
first, last = name.split(" ")
print(f"hello, {first}")
```

```
# Demonstrates addition

x = 1
y = 2

z = x + y

print(z)
```

```python
# Demonstrates (unintended) concatenation of strings

# Prompt user for two integers
x = input("What's x? ")
y = input("What's y? ")

# Print sum
z = x + y
print(z)
```

```python
# Demonstrates conversion from str to int

x = input("What's x? ")
x = int(x)
y = input("What's y? ")
y = int(y)

z = x + y

print(z)
```

```
# Demonstrates nesting of function calls

x = int(input("What's x? "))
y = int(input("What's y? "))

z = x + y

print(z)
```

```python
# Demonstrates conversion of str to float

x = float(input("What's x? "))
y = float(input("What's y? "))

z = x + y

print(z)
```

```
# Demonstrates fewer variables

x = float(input("What's x? "))
y = float(input("What's y? "))

print(round(x + y))
```

```python
# Demonstrates floating-point imprecision (e.g., 1.1 + 2.2)

x = float(input("What's x? "))
y = float(input("What's y? "))

z = x + y

print(f"{z:.50f}")
```

```python
# Demonstrates floating-point imprecision (e.g., 1 / 3)

x = float(input("What's x? "))
y = float(input("What's y? "))

z = x / y

print(f"{z:.50f}")
```

```
# Demonstrates multiple (identical) function calls

print("meow")
print("meow")
print("meow")
```

```python
# Demonstrates a for loop, using range

for i in range(3):
    print("meow")
```

```python
# Demonstrates defininig a function

def meow():
    print("meow")


for i in range(3):
    meow()
```

```python
# Says hello

import pyttsx3

engine = pyttsx3.init()
engine.say("hello, world")
engine.runAndWait()
```

```python
# Says hello

import pyttsx3

engine = pyttsx3.init()
name = input("What's your name? ")
engine.say(f"hello, {name}")
engine.runAndWait()
```