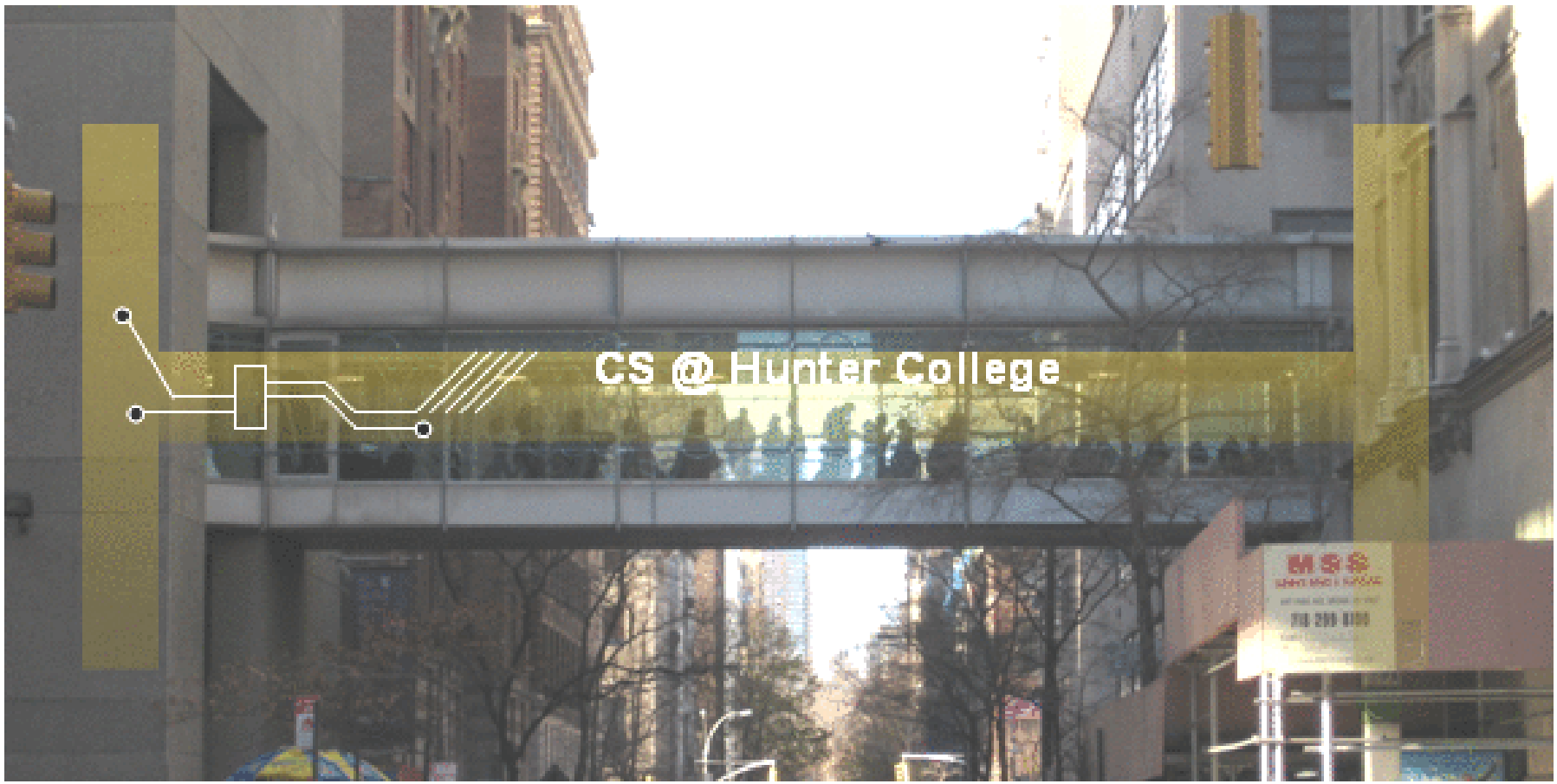
CSCI 127: Introduction to Computer Science



hunter.cuny.edu/csci

Today's Topics

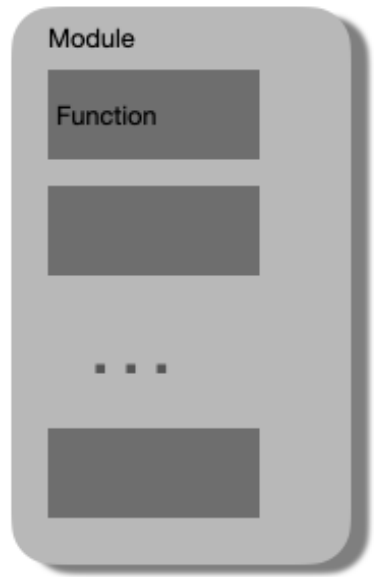
- **Introduction to Functions**
- Recap: Slicing & Images

Functions

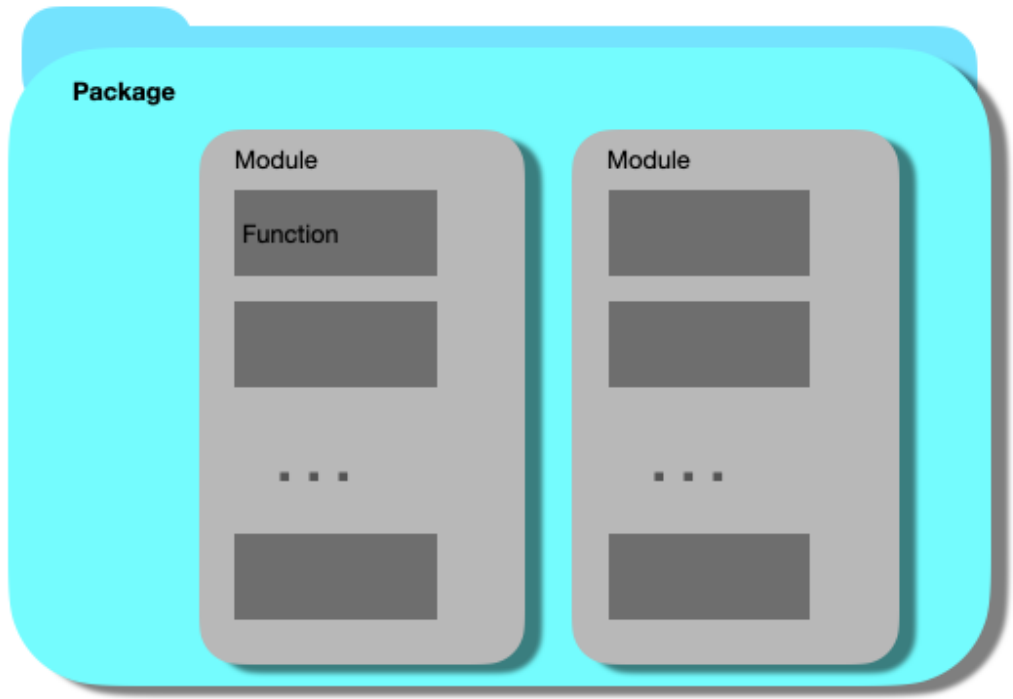
```
#Name:  your name here
#Date:  October 2017
#This program, uses functions,
#     says hello to the world!

def main():
     print("Hello, World!")

if __name__ == "__main__":
     main()
```

- Functions are a way to break code into pieces, that can be easily reused.
- Many languages require that all code must be organized with functions.
- The opening function is often called `main()`
- Naming conventions same as variables
- You **call** or **invoke** a function by typing its name, followed by any inputs, surrounded by parenthesis: Example: `print("Hello", "World")`
- Can write, or **define** your own functions, which are stored, until invoked or called.

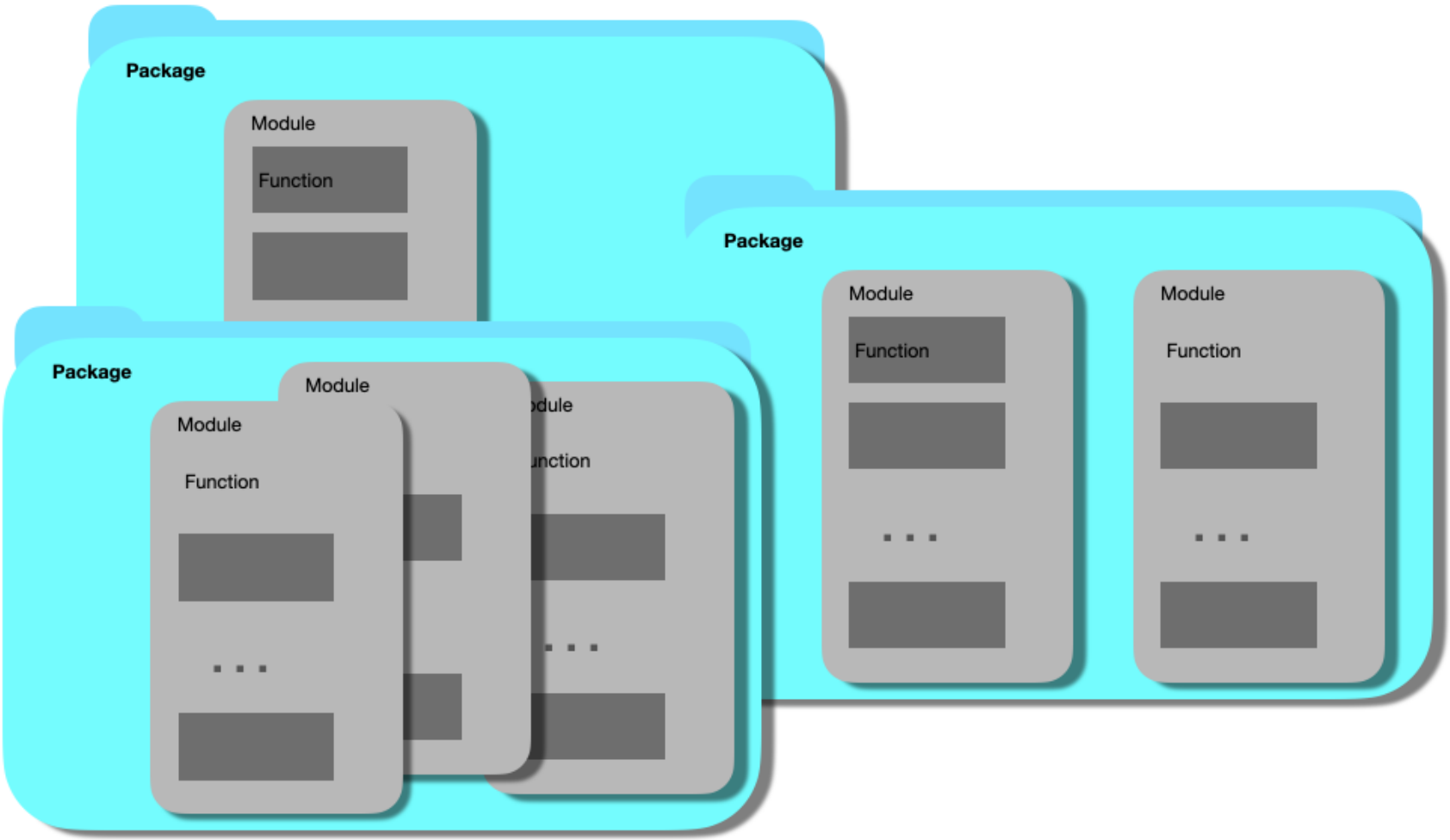
Modules and packages



Modules and packages



Modules and packages



Challenge: *Predict what the code will do*

```
def totalWithTax(food, tip):
    tax = 0.1 * food
    return(food + tax + tip)

lunch = float(input("Enter lunch total: "))
l_tip = float(input("Enter lunch tip: " ))
l_total = totalWithTax(lunch, l_tip)
print("Lunch total is", l_total)
```

Challenge: *Predict what the code will do*

```
def totalWithTax(food, tip):
    tax = 0.1 * food
    return(food + tax + tip)

dinner = float(input("Enter dinner total: "))
d_tip = float(input("Enter dinner tip: " ))
d_total = totalWithTax(dinner, d_tip)
print("Dinner total is", d_total)
```

Scope

```
def eight():
    x = 5+3
    print(x)

def nine():
    x = "nine"
    print(x)
```

- You can have multiple functions.
- Each function defines the **scope** of its local variables
- A variable defined inside a function is **local**, i.e. defined only inside that function.

Input Parameters & Return Values

```
def totalWithTax(food,tip):
    total = 0
    tax = 0.0875
    total = food + food * tax
    total = total + tip
    return(total)

lunch = float(input('Enter lunch total: '))
lTip = float(input('Enter lunch tip:' ))
lTotal = totalWithTax(lunch, lTip)
print('Lunch total is', lTotal)

dinner= float(input('Enter dinner total: '))
dTip = float(input('Enter dinner tip:' ))
dTotal = totalWithTax(dinner, dTip)
print('Dinner total is', dTotal)
```

- Functions can have **input parameters**.
- Surrounded by parentheses, both in the function definition, and in the function call (invocation).
- The "placeholders" in the function definition: **formal parameters**.
- The ones in the function call: **actual parameters**

Input Parameters & Return Values

```
def totalWithTax(food,tip):
    total = 0
    tax = 0.0875
    total = food + food * tax
    total = total + tip
    return(total)

lunch = float(input('Enter lunch total: '))
lTip = float(input('Enter lunch tip:' ))
lTotal = totalWithTax(lunch, lTip)
print('Lunch total is', lTotal)

dinner= float(input('Enter dinner total: '))
dTip = float(input('Enter dinner tip:' ))
dTotal = totalWithTax(dinner, dTip)
print('Dinner total is', dTotal)
```

Formal Parameters

Actual Parameters

- Functions can have **input parameters**.
- Surrounded by parentheses, both in the function definition, and in the function call (invocation).
- The "placeholders" in the function definition: **formal parameters**.
- The ones in the function call: **actual parameters**.
- Functions can also **return values** to where it was called.

GitHub

Octocat

- Used to collaborate on and share code, documents, etc.
- Supporting Open-Source Software: original source code is made freely available and may be redistributed and modified.
- More formally: `git` is a version control protocol for tracking changes and versions of documents.
- GitHub provides hosting for repositories (**'repos'**) of code.
- Also a convenient place to host websites (e.g. `huntercsci127.github.io`).

Challenge: *Predict what the code will do:*

```
def helper(meg, jo):
  s = ""
  for j in range(meg):
    print(j, ":", jo[j])
    if j % 2 == 0:
      s = s + jo[j]
      print("Building s:", s)
  return s
```

Lecture slip

Link to lecture slip example

Recap: Functions

```
#Name:  your name here
#Date:  October 2017
#This program, uses functions,
#     says hello to the world!

def main():
     print("Hello, World!")

if __name__ == "__main__":
     main()
```

- Functions are a way to break code into pieces that can be easily reused.
- You **call** or **invoke** a function by typing its name, followed by any inputs, surrounded by parenthesis: Example: `print("Hello", "World")`
- Can write, or **define** your own functions, which are stored, until invoked or called.

Today's Topics

- Introduction to Functions
- **Recap: Slicing & Images**

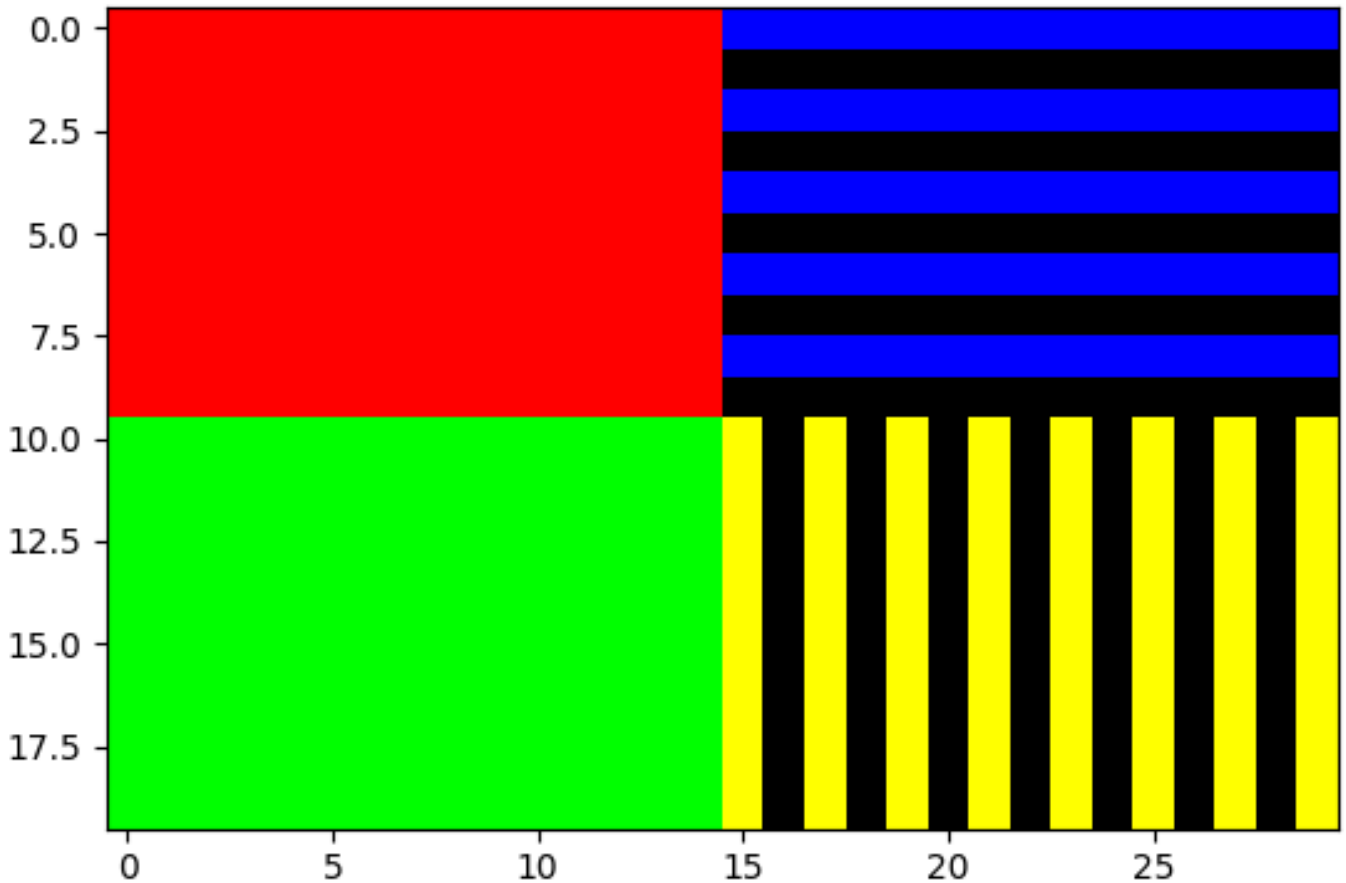
Images and Arrays

```
import matplotlib.pyplot as plt
import numpy as np
height= 20
width = 30

#An image is an array with height, width, and depth
#The height and width can be any integers but
# the depth is always 3 for the red, green, and blue channels
img = np.zeros((height, width, 3))
img[:height//2, :width//2, 0] = 1 #upper left corner
```

Images and Arrays (cont.)

```
img[height//2:, :width//2, 1] = 1 #lower left corner

img[:height//2:2, width//2:, 2] = 1 #upper right corner

img[height//2:, width//2::2, :2] = 1 #lower right corner

plt.imshow(img)
plt.show()
```

output for the above program

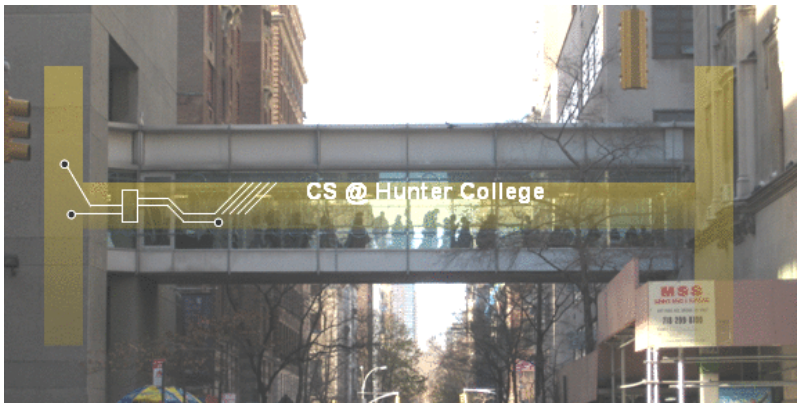


Review: Cropping Images

```
import matplotlib.pyplot as plt
import numpy as np
img = plt.imread('csBridge')
plt.imshow(img)
plt.show()
height = img.shape[0]
width = img.shape[1]
img2 = img[:height//2, :width//2]
plt.imshow(img2)
plt.show()
```

Review: Cropping Images

Weekly Reminders!



Before next lecture, don't forget to:

- Work on this week's Online Lab
- Schedule an appointment to take the Quiz in lab 1001G Hunter North
- If you haven't already, schedule an appointment to take the Code Review (**one every week**) in lab 1001G Hunter North
- Submit this week's programming assignments
- If you need help, schedule an appointment for Tutoring in lab 1001G 11:30am-5:30pm

Lecture Slips & Writing Boards

- Hand your lecture slip to a UTA.
- Return writing boards as you leave.