

Today's class

- Naming things
- Functions
- “It doesn't work!”

Python names can be (nearly) anything

- Must start with a letter (but can *contain* numerals)
- Can't contain spaces
 - **myPicture** is okay but **my Picture** is not
- Be careful not to use command names as your own names
 - **print = 1** won't work
 - (Avoid names that appear in the editor pane of JES highlighted in blue or purple)
- Case matters
 - **MyPicture** is not the same as **myPicture** or **mypicture**
- Sensible names are sensible
 - E.g. **myPicture** is a good name for a picture, but not for a sound file.
 - **x** could be a good name for an x-coordinate in a picture, but probably not for anything else - it is too vague

JES Functions

- Many functions are pre-defined in JES for sound and picture manipulations
 - **pickAFile()**
 - **makePicture()**
 - **makeSound()**
 - **show()**
 - **play()**
- Some of these functions accept *input* values called parameters

```
theFile = pickAFile()
pic = makePicture(theFile)
```

Picture Functions

- **makePicture**(filename)
creates and returns a picture object, from the JPEG file at the filename
- **show**(picture)
displays a picture in a window
- We'll learn functions for manipulating pictures later, like **getColor**, **setColor**, and **repaint**

A recipe for showing picked picture files

```
def pickAndShow():  
    myfile = pickAFile()  
    mypicture = makePicture(myfile)  
    show(mypicture)
```

Function parameters allow flexibility

```
def playNamed(myfile):  
    mysound = makeSound(myfile)  
    play(mysound)
```

```
def showNamed(myfile):  
    mypict = makePicture(myfile)  
    show(mypict)
```

Q: What functions do you need?

Q: What (if any) should be their parameter(s)?

A: In general, have enough functions to do what you want, easily, understandably, and flexibly
(try for more generic, less specific functions)

What can go wrong when things look right?

- Did you use the *exact* same names (case, spelling)?
- All the lines in the block must be *indented*, and *indented the same amount*.
- Variables in the command area don't exist in your functions, and variables in your functions don't exist in the command area.
- The computer can't read your mind.
 - It will only do exactly what you tell it to do.

Programming is a craft

- You don't learn to write, paint, or ride a bike by attending biking lectures and watching others bike.
 - You learn to bike by *biking!*
- Programming is much the same.
 - You have to try it, make many mistakes, learn how to control the computer, learn how to *think* in Python.
- The six programs that you have to write in this class aren't enough!
 - Do programming on your own!
 - Play around with the class and book examples!