After the course

- You know how to write programs sophisticated enough to do real work.
- Lots of real work requires you to use modules.
- Modules contain new functions.
- Modules also usually contain new data types!
- These are called classes.
- It’s important to know a little bit about classes to use modules.

Example: the tkinter module

- Used to define Graphical User Interfaces (one of many).
- GUI modules unfortunately always seem complicated to use.
- Let’s make a window.

Classes

- A class is like a new data type.
- Classes are most often added by modules. The tkinter module adds the Tk class (among others)
- An object in Python is any data item.

```python
root = tk.Tk()
```

- Creates an object which is an instance of class Tk, and puts it into variable root.
- Kind of like an “object of type Tk” — but with different jargon.

Data as objects

```python
name = ‘Nancy Drew’
root = tk.Tk()
```

- `name` is a variable, containing a string.
- `root` is a variable, containing a Tk, whatever that is.
- Strings and Tks are two kinds of objects; the class string is built-in to Python, but the class Tk was added by a module.

Instance

- There can be lots of objects of type string in a program, and similarly lots of objects of type Tk.
- Each object is an instance of its class.
Factory Functions

- Create instances of the class.
- Often these are the only functions in the module.
- Anything that works with an existing instance is given as a method.
- This is the difference between a function and a method — methods are functions that belong to specific kinds of objects.

```python
root.title("cute little window")
root.geometry("200x100")
root.mainloop()
```

Label objects

- Stuff that just sits in your window.

```python
label = tk.Label(root, text="Yowza!")
```

- Can also be pictures.

```python
flower = PhotoImage(file="Rhododendrum.gif")
pic = tk.Label(root, image=flower)
```

- Only guaranteed to handle certain image formats (gif, ppm, ppm), sadly.

Layout manager

- Figures out where everything goes in the window.
- We’ll just use the grid manager.

```python
label.grid()
pic.grid()
```

Data in an Object

- Objects store data as well as methods.
- tkinter lets the user have access to some (but not all) of the data stored in an object.

```python
print button.cget("text")
```

Style parameters

- Modify the style object

```python
bigFont = font.Font(family='Helvetica', size=26)
style = ttk.Style()
style.configure("TButton", font=bigFont)
style.configure("TLabel", font=bigFont)
```

Object Oriented Programming

- A philosophy about how to organize programs.
- Keep data organized into objects, along with methods to use or modify that data.
- Many languages, including Python, have a lot of syntactic sugar to make object oriented programming easier.
- Lots of modules are organized as collections of classes.
The future

- If you go on to use Python, you’ll use a lot of modules that have classes in them.
- We’ll make a user interface in Project 6.