Python Modules

- A file in Python is called a *module*
  - Modules allow us to store code and run it later
  - Modules also allow us to *reuse* code in different projects

- To use code from a module we *import* it

  ```
  import math
  ```

  - This makes the name of the module a meaningful symbol in our current environment
    - The name "math" is added to the table of symbols and references that module
    - So we can't have some other variable or function named "math" or one of them gets clobbered

Python Modules

- The *math* module has a square root function called *sqrt*
  - It takes one parameter (a number) and returns the square root of that number

- Once math is imported, we can refer to "the sqrt function that belongs to the math module" using the . operator

```
import math
x = math.sqrt(16)
```

Python Modules

▸ Alternatively, we could import just one particular function from math

```
from math import sqrt
x = sqrt(16)
```

 ▸ In this case "math" is not a meaningful symbol

 ▸ "math.sqrt" is not a meaningful symbol

 ▸ "sqrt" is, because that's the only thing we imported

▸ Alternatively, you could import everything from a module, but that is generally unwise

```
from math import *
```