Modules: Takeaways

Syntax

LOADING FUNCTIONS AND VARIABLES FRFOM MODULES

- To import an entire module:

```
importmath
```

- To use a function after importing the entire module:

```
importmath

root= math.sqrt(99)

flr = math.floor(89.9)
```

- To import an entire module using an alias:

```
importmathas m

root= m.sqrt(33)
```

- To import all objects from a module:

```
frommath import*

root= sqrt(1001)
```

- To import a specific function from a module:

```
frommath importsqrt

frommath importfloor

root= sqrt(99)

flr = floor(89.9)
```

- To import a specific variable from a module:

```
importmath

a = math.sqrt(math.pi)

b = math.ceil(math.pi)

c = math.floor(math.pi)
```

USING THE CSV MODULE

- To open a file using the **csv** module:

```
importcsv

f = open("nfl.csv"r')

csvreader= csv.reader(f)

nfl = list(csvreader)
```

Concepts

- A **module** is a collection of functions and variables that have been bundled together in a single file. Modules help us:
    - Organize our code by separating related functions and objects into their own modules.
    - Gain new functionality by using code written by others.
- The **namespace** is a dictionary that contains all the names we can refer to. Whenever we load a module, we're loading all it's associated function into the namespace.

Resources

- Documentation on modules in Python
- Documentation on the import system in Python