

Using functions in modules

- Modularity suggests grouping functions together that share common theme
- Place in a single .py file
- Use import command to access

Example

```
pi = 3.14159
```

```
def area(radius):  
    return pi*(radius**2)
```

```
def circumference(radius):  
    return 2*pi*radius
```

```
def sphereSurface(radius):  
    return 4.0*area(radius)
```

```
def sphereVolume(radius):  
    return (4.0/3.0)*pi*(radius**3)
```

Be sure to save this code in
a file called `circle.py`

Example

```
import circle
pi = 3.0
print pi
print circle.pi
print circle.area(3)
print
    circle.circumference(3)
```

- Will result in
3.0 # value from local env
3.14159 # value from file
28.27431 # uses values from
file
18.84954
- The . notation specifies
context from which to
read values

Example

```
from circle import *
```

```
pi = 0.0
```

```
print pi
```

```
print area(3)
```

```
print circumference(3)
```

- Will result in
0.0 # value from local scope
28.27431 # uses values from
file, because area not
bound locally, but inherits
from circle; however format
allows reference as if in
local scope

18.84954