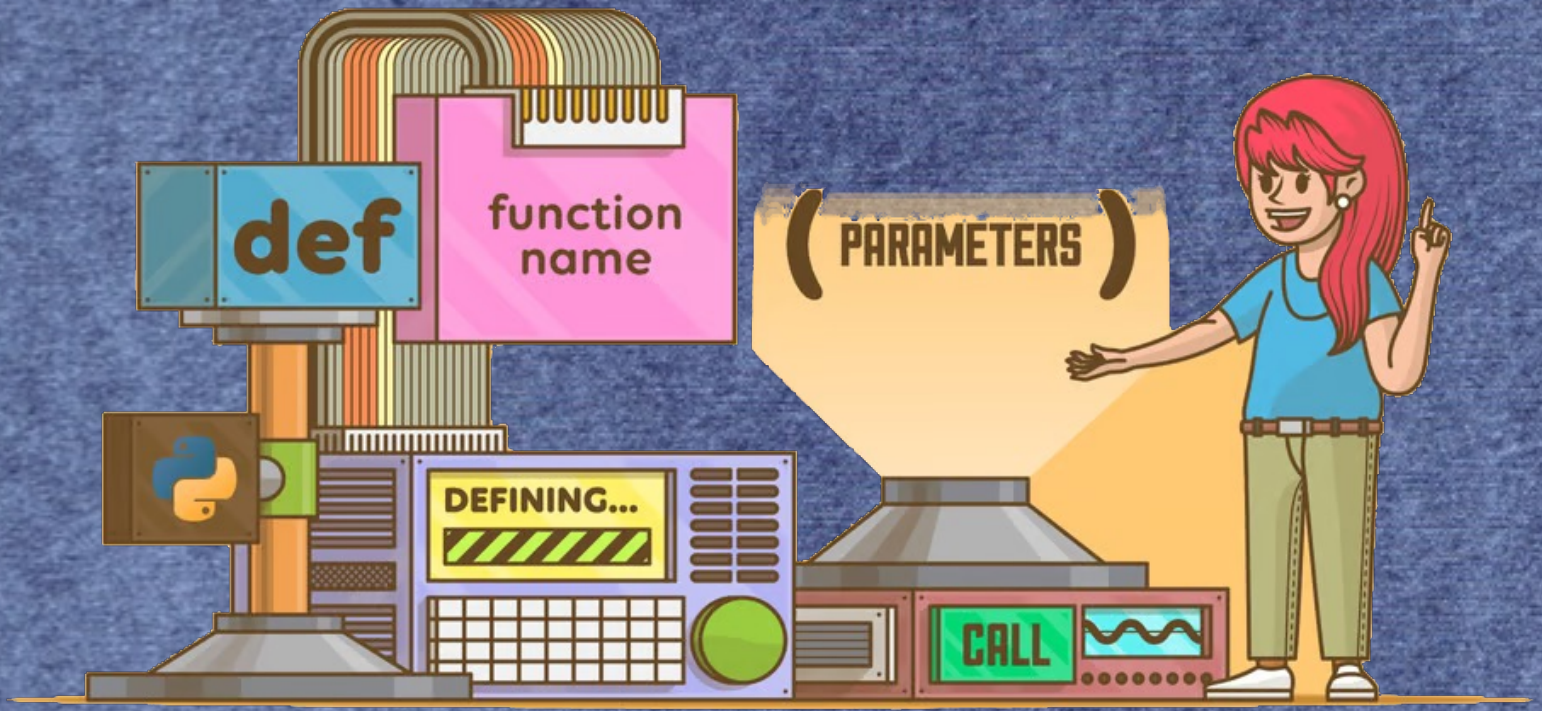
Functions



Functions

- What are functions?
- Syntax
- Calling
- Parameters
- Return values
- Variable Scope

Functions

- Named sequence of statements that belong together
- Primary purpose
  - organize programs into chunks
  - that match how we think about problem solution

Functions

- Compound Statement
  - –consists of one or several statements
  - –: used to indicate beginning
  - –indentation

```
def name( parameters ):
    statements
```

Function Syntax

```
def name( parameters ):
    statements
```

Function Example

```
def drawSquare():                    #function definition
    tess.forward(120)
    tess.left(90)
    tess.forward(120)
    tess.left(90)
    tess.forward(120)
    tess.left(90)
    tess.forward(120)
    tess.left(90)


drawSquare()                 # function call
```

Function Example

```
def drawSquare():                    #function definition
    for side in range(4):
                tess.forward(120)
                tess.left(90)


drawSquare()                         # function call
```

You Try It

- modify rotateSquare.py to use a function to draw one of the squares

```
def drawSquare():                    #function definition
    for side in range(4):
            tess.forward(120)
            tess.left(90)


drawSquare()                         # function call
```

Function Parameter Example

```
def drawSquare(sideLength):       #function definition
    for side in range(4):
                tess.forward(sideLength)
                tess.left(90)


tess.speed(0)

for square in range(36):
        drawSquare(150)                 # function call
        tess.color("blue")
        tess.left(5)
```

Function Parameter Example

```
import turtle
wn=turtle.Screen()

alex = turtle.Turtle()
tess = turtle.Turtle()

def drawSquare(turtle, sideLength):          # function definition
    for side in range(4):
                    turtle.forward(sideLength)
                    turtle.left(90)


drawSquare(alex, 150)                        # function call
drawSquare(tess, 300)
wn.exitonclick()
```

Function Return Example

```
def square(x):
        y = x * x
        return y

toSquare = 10

squareResult = square(toSquare)

print("The result of ", toSquare, " squared is ", squareResult)
```

You Try It

- write and test a function that calculates the area of a circle (πr^2) and returns the result

write and test a function that calculates the area of a circle (πr^2)

```
def circleArea(radius):
        area = 3.14 * radius * radius
        return area

radius = int(input('the radius is: '))

area = circleArea(radius)

print("The area of a circle of radius", radius, " is ", area)
```

Function Variable Scope

- variables and parameters are **local**
- functions can access **global** variables
  - but it is generally not good form

Function Variable Scope

```
def square():
        y = x * x
        return y

x = 10
z = square()

print(z)
print(x)
```

Function Variable Scope

```
def square(x):
        y = x * x
        x = 5
        return y

x = 10
z = square(x)

print(z)
print(x)
```

Function Variable Scope

- variables and parameters are **local**
- functions can access **global** variables
  - but it is generally not good form

Function Review

- What are functions?
- Syntax

```
def name( parameters ):
    statements
    return
```

- Calling
- Parameters
- Return values
- Variable Scope

Superbly Excellent!