Functions

(continued)

Functions: a review

- Little pieces of code that we can write and invoke by name
  - Reusable
  - General
  - Do one thing

Functions: a review

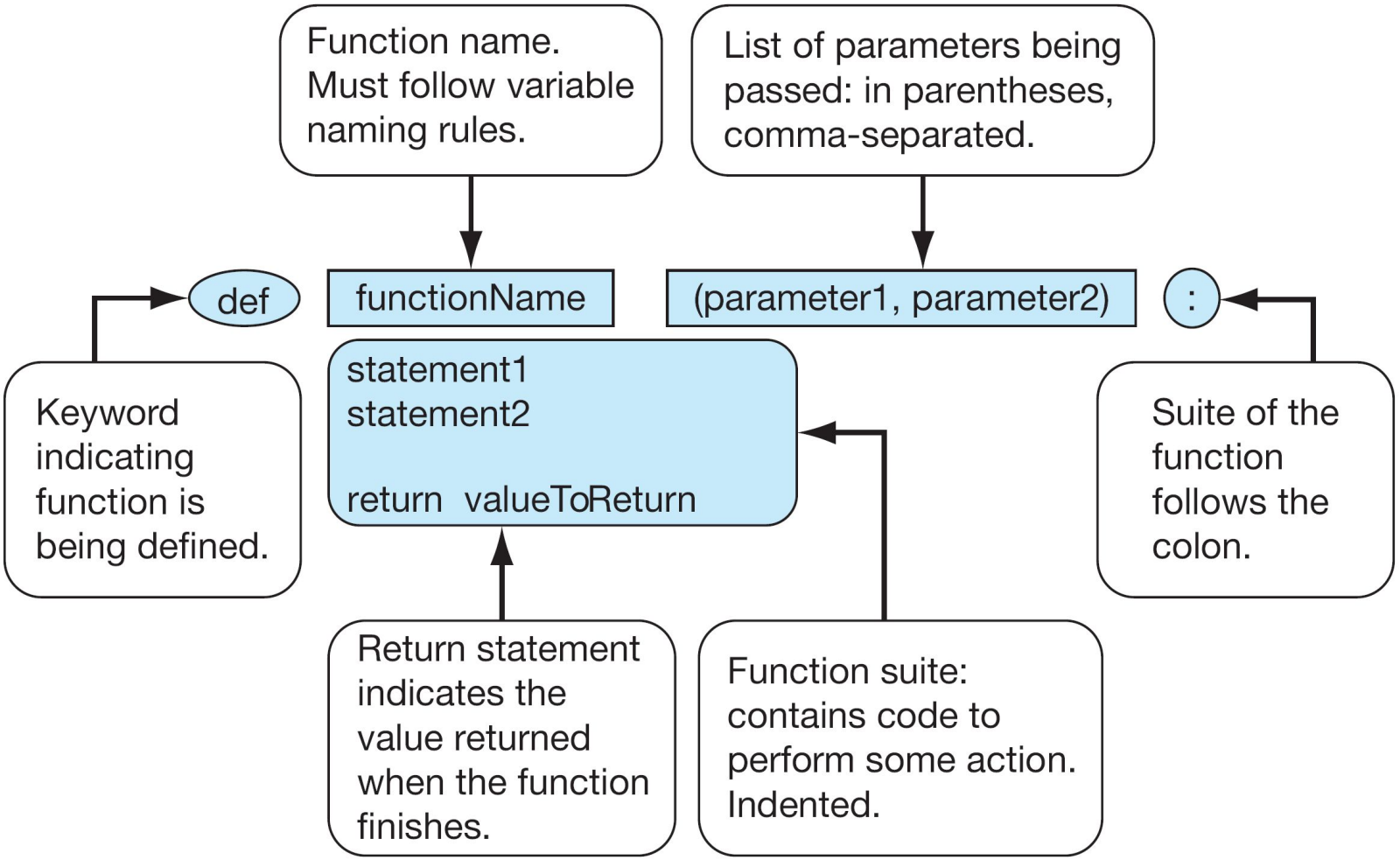
- Take in any number of parameters
  - Including no parameters!
- Possibly compute something
- Return a value
  - Can be None
  - If no return is specified, returns None

Two Parts to a Function

- ***Definition*** – creates the function
- ***Invocation*** – is the application of a function within a program

- A function must be ***defined*** before it is invoked

Function Definition

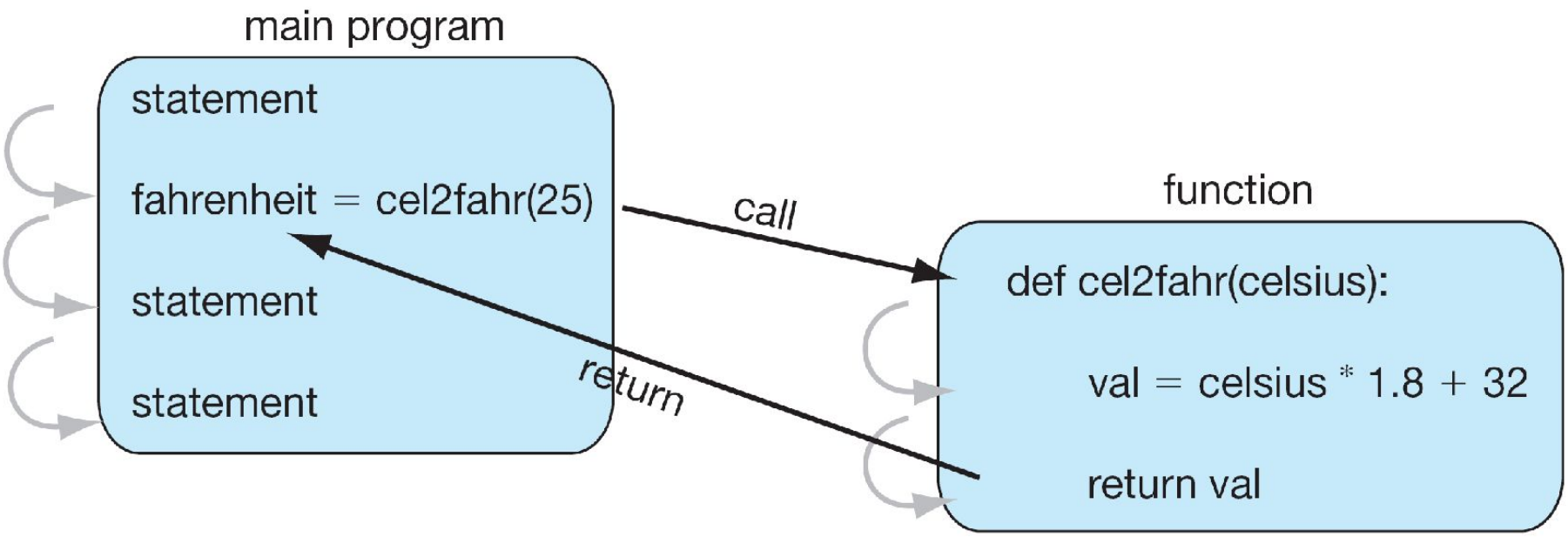


Function Definition + Invocation

```
def celsius_2_fahrenheit (c):
        """ Convert Celsius to Fahrenheit."""
         return c * 1.8 + 32.0

c_temp = int(input("temperature to convert: "))
f = celsius_2_Fahrenheit(c_temp)
```

Flow of control



Function that takes in no parameters

```
def answerToEverything():
    return 42
```

- This function takes in no parameters
- It does no computation
- Just returns a number

Function that returns nothing

- Let's take a look at print()

```
>>> result = print("hi")
>>> print(result)
None
>>>
```

None

- None is a special value in Python that represents nothing
  - The first letter of None must be capitalized – it will turn orange (in IDLE)
- Use it when you have nothing to return
  - Like if one of the parameters was invalid

Future Programming Assignments

- I will ask you to write python files with only functions in them
- That's ok – you can call functions directly from IDLE
  - Step 1: Run your python file (F5)
  - At interpreter, type your function name with parameters

Code Comments

- Still do what you are already doing
- Before each function, list
  - Function name
  - Inputs – inputs parameters to your function and their types
  - Output – what your function returns and its type
- Inside the function using a docstring (triple quotes)
  - Description – what your function does

Example Program

- Let's say that we are writing a computer-computer rock/paper/scissors game with 4 functions
  - Function getWeapon(): no inputs, gets random weapon from "r", "p", "s", returns "r","p","s"
  - Function isTie(one,two): inputs weapons from both players, returns true if tie and false otherwise
  - Function playerOneWins(one,two): inputs weapons from both players, returns true if player one wins and false otherwise

Example Program

- Let's say that we are writing a computer-computer rock/paper/scissors game with 4 functions
  - Function rps(numberOfGames): input is the number of games (int), plays rock paper scissors with two computer players, and returns the result of the number of player1 wins, player2 wins, and ties in a string