Welcome to Introduction to Python – Level 2

What do you need to get started?

1. You must have completed Level 1
2. Python (3.3.5): https://www.python.org/downloads/
3. PyCharm: http://www.jetbrains.com/pycharm/download/

https://github.com/anooptrivedi/workshops-python-level2

Class time: 2 hrs.

Wi-Fi: Free-WiFi

Who is helping you today?

Anoop Trivedi: Instructor

Aarthi Rao: Instructor

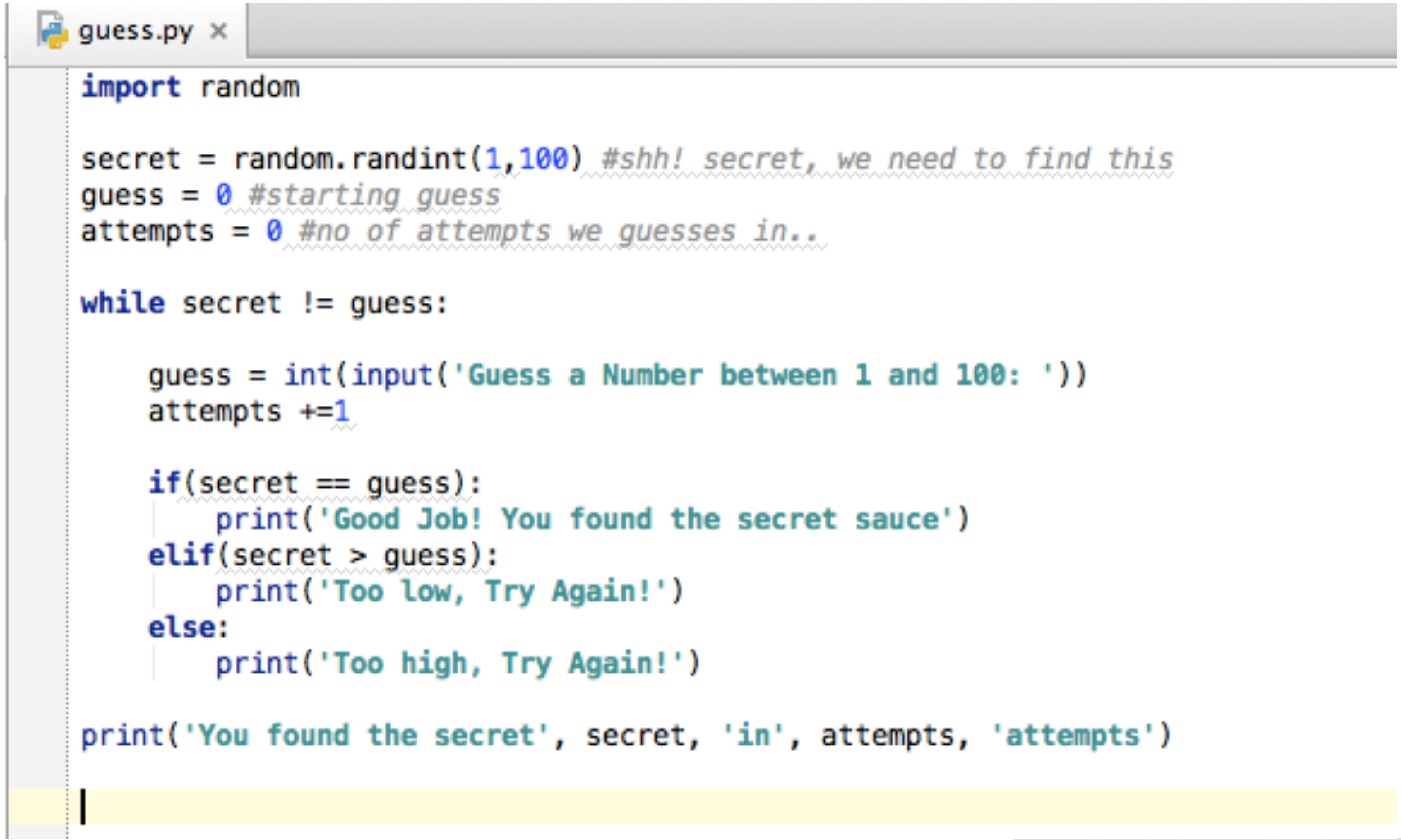
Pradeep Bhatter: Instructor

What are we going to do today ?

1. *Brush up from Lesson 1 – Guessing Game*
2. *Play with list, tuples, and dictionary*
3. *Basics of functions*
4. *Passing parameters to functions*
5. *Passing Tuples and Dictionary to functions*
6. *Introduction to Tkinter*
7. *Lets Play Hangman!*

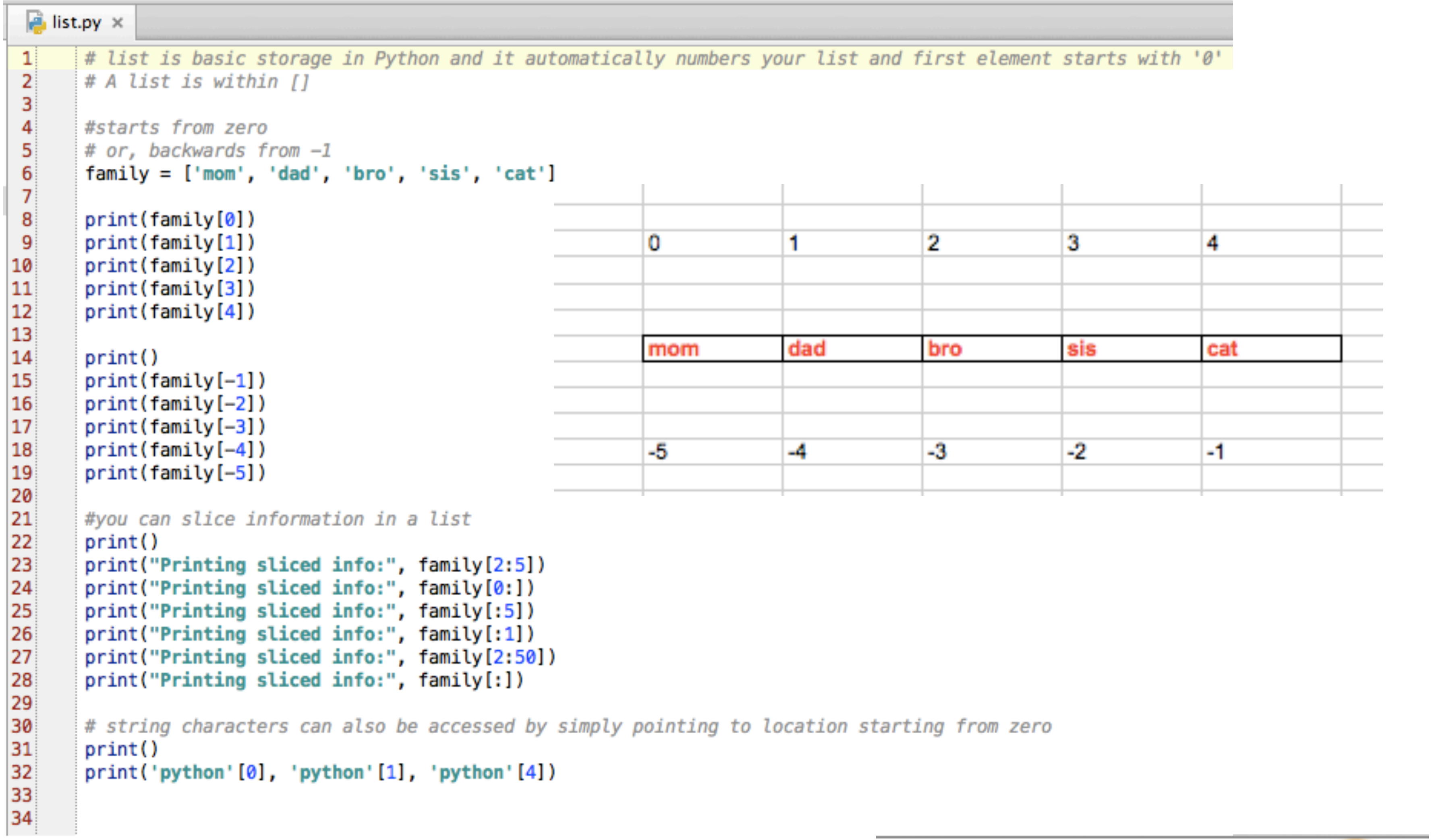
Mission: Find my secret number?

Computer has generated a secret number, can you hack it?



```python
import random

secret = random.randint(1,100) #shh! secret, we need to find this
guess = 0 #starting guess
attempts = 0 #no of attempts we guesses in..

while secret != guess:

    guess = int(input('Guess a Number between 1 and 100: '))
    attempts +=1

    if(secret == guess):
        print('Good Job! You found the secret sauce')
    elif(secret > guess):
        print('Too low, Try Again!')
    else:
        print('Too high, Try Again!')

print('You found the secret', secret, 'in', attempts, 'attempts')
```

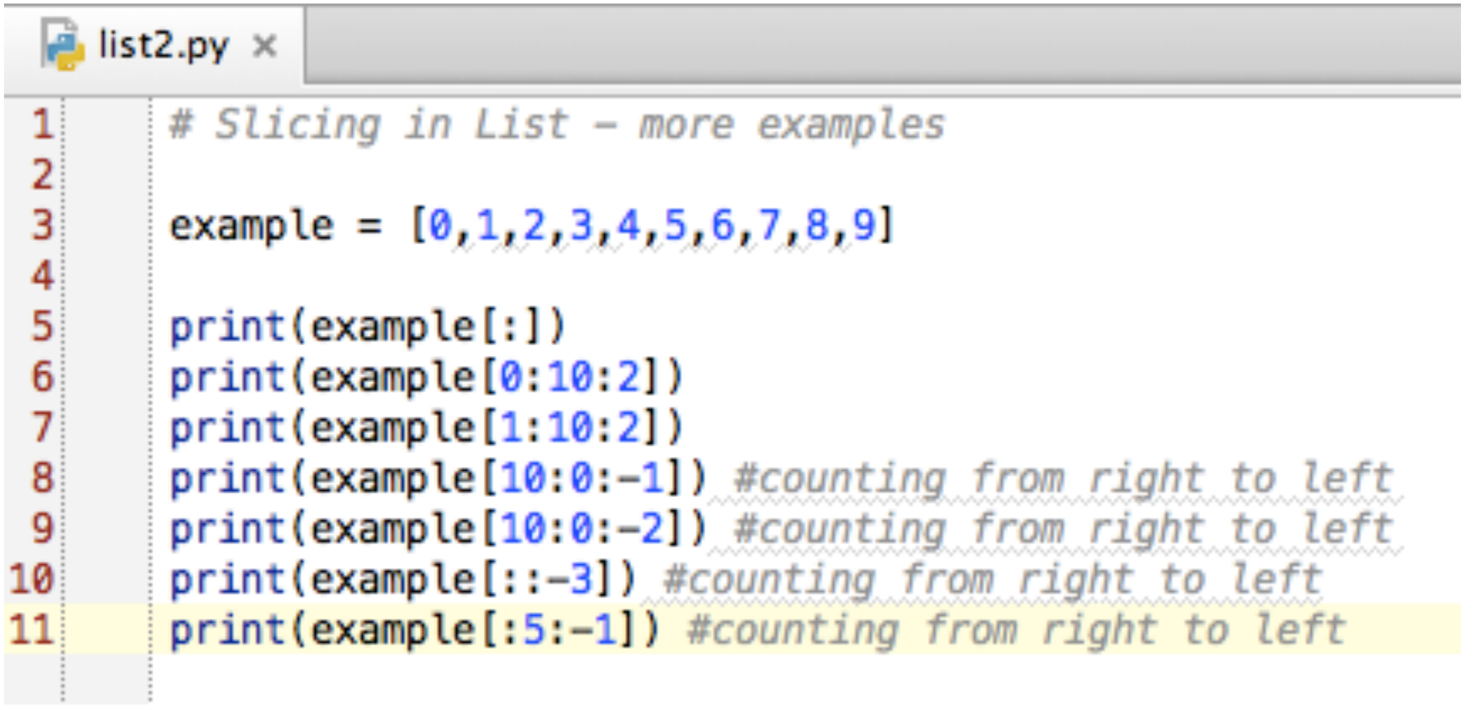
Slicing List:



```python
# list is basic storage in Python and it automatically numbers your list and first element starts with '0'
# A list is within []

#starts from zero
# or, backwards from -1
family = ['mom', 'dad', 'bro', 'sis', 'cat']

print(family[0])
print(family[1])
print(family[2])
print(family[3])
print(family[4])

print()
print(family[-1])
print(family[-2])
print(family[-3])
print(family[-4])
print(family[-5])

#you can slice information in a list
print()
print("Printing sliced info:", family[2:5])
print("Printing sliced info:", family[0:])
print("Printing sliced info:", family[:5])
print("Printing sliced info:", family[:1])
print("Printing sliced info:", family[2:50])
print("Printing sliced info:", family[:])

# string characters can also be accessed by simply pointing to location starting from zero
print()
print('python'[0], 'python'[1], 'python'[4])
```

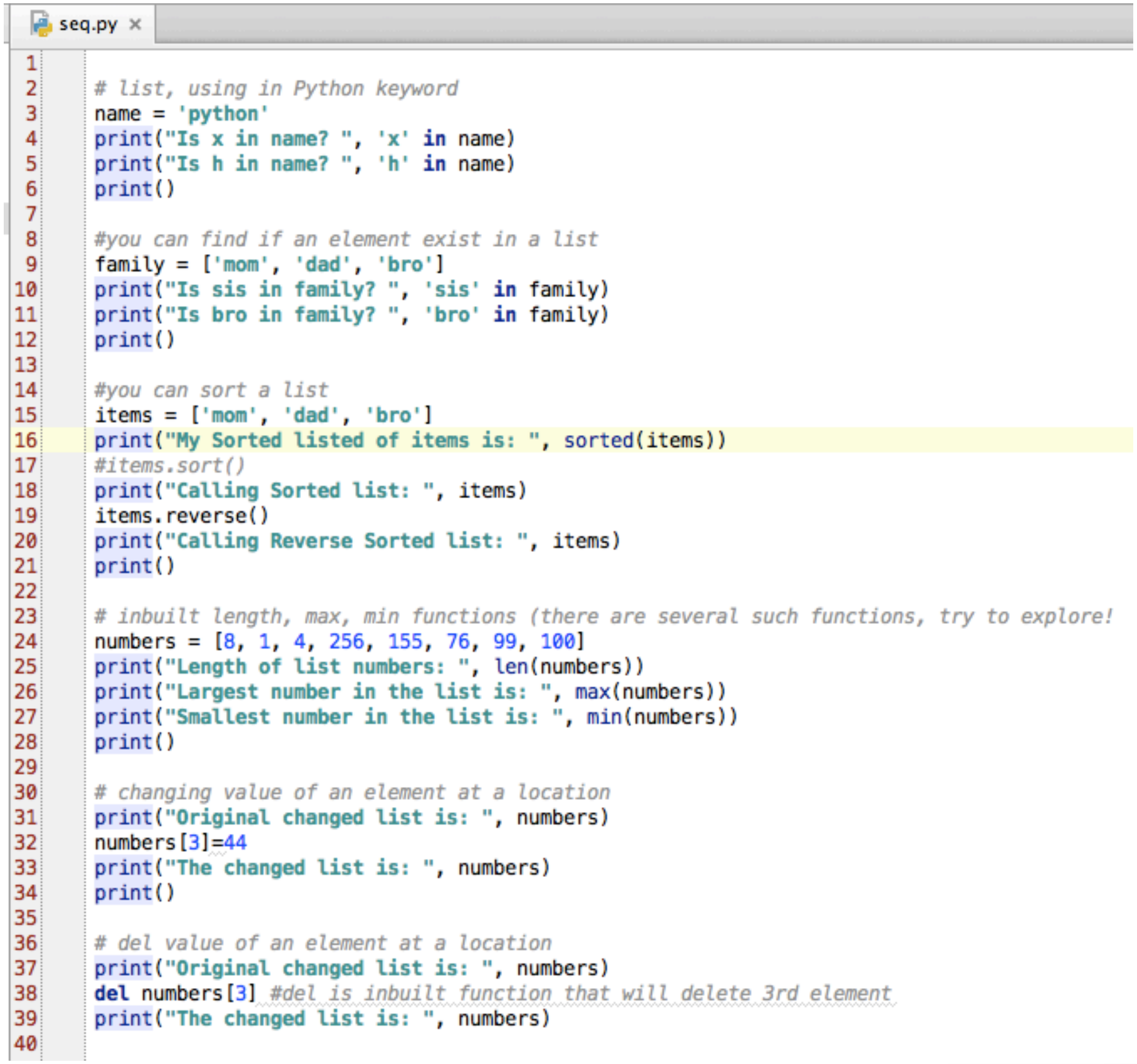
0	1	2	3	4
mom	dad	bro	sis	cat
-5	-4	-3	-2	-1

More ways of Slicing List:



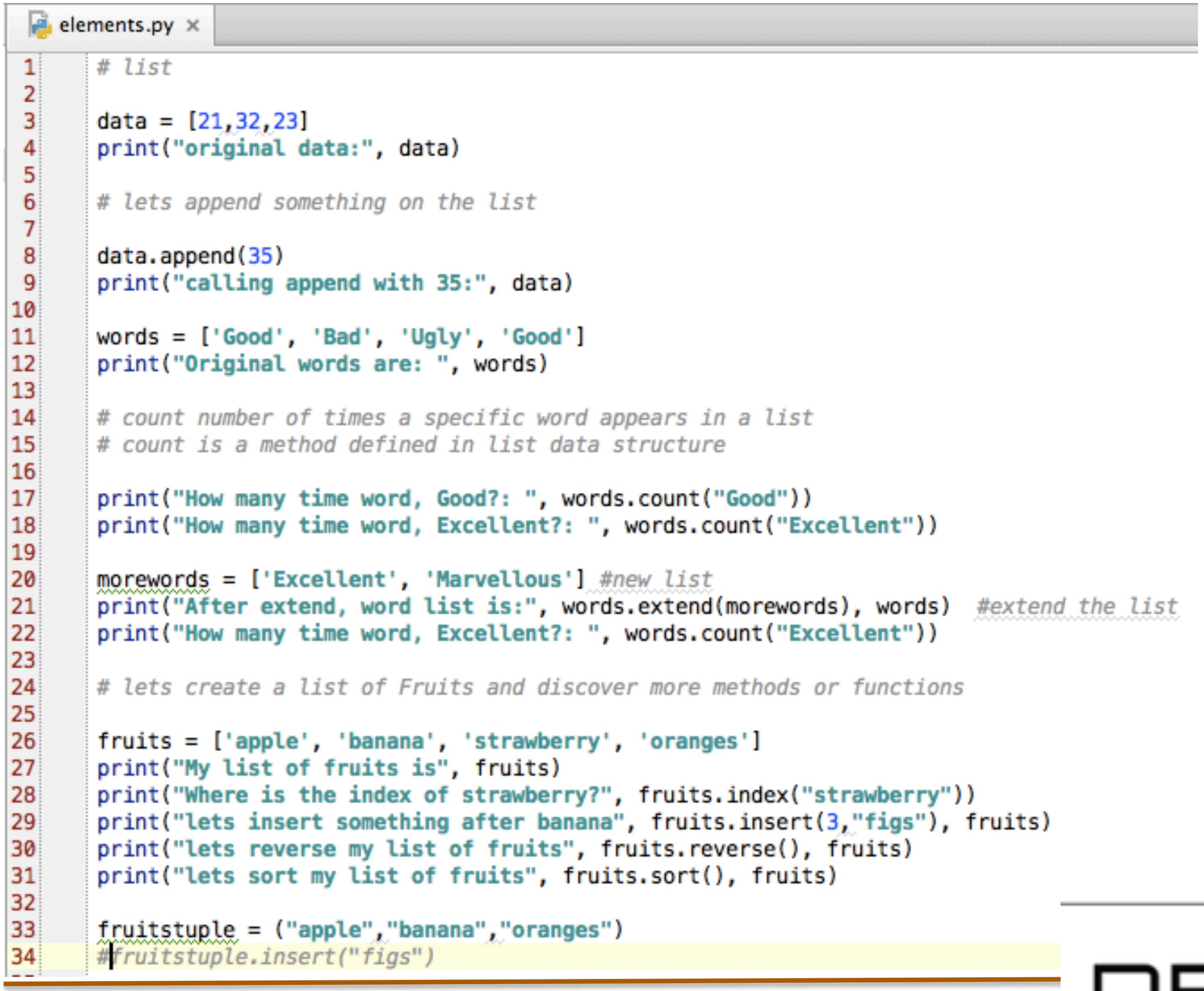
```python
# Slicing in List - more examples

example = [0,1,2,3,4,5,6,7,8,9]

print(example[:])
print(example[0:10:2])
print(example[1:10:2])
print(example[10:0:-1]) #counting from right to left
print(example[10:0:-2]) #counting from right to left
print(example[::-3]) #counting from right to left
print(example[:5:-1]) #counting from right to left
```

Calling in-built methods



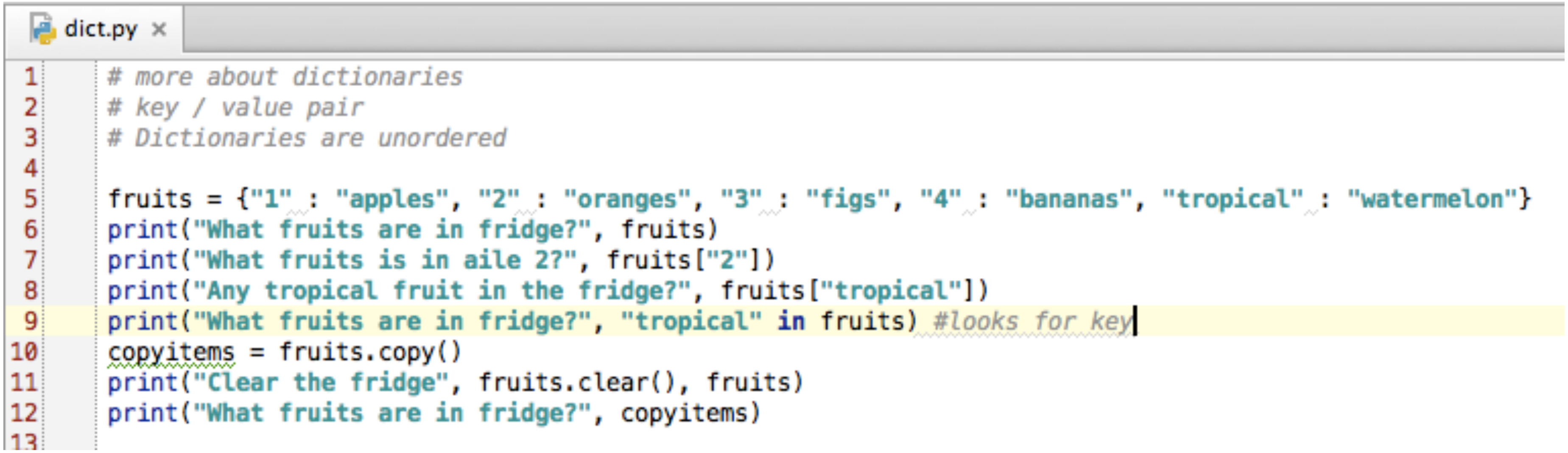
```

# list, using in Python keyword
name = 'python'
print("Is x in name? ", 'x' in name)
print("Is h in name? ", 'h' in name)
print()

#you can find if an element exist in a list
family = ['mom', 'dad', 'bro']
print("Is sis in family? ", 'sis' in family)
print("Is bro in family? ", 'bro' in family)
print()

#you can sort a list
items = ['mom', 'dad', 'bro']
print("My Sorted listed of items is: ", sorted(items))
#items.sort()
print("Calling Sorted list: ", items)
items.reverse()
print("Calling Reverse Sorted list: ", items)
print()

# inbuilt length, max, min functions (there are several such functions, try to explore!
numbers = [8, 1, 4, 256, 155, 76, 99, 100]
print("Length of list numbers: ", len(numbers))
print("Largest number in the list is: ", max(numbers))
print("Smallest number in the list is: ", min(numbers))
print()

# changing value of an element at a location
print("Original changed list is: ", numbers)
numbers[3]=44
print("The changed list is: ", numbers)
print()

# del value of an element at a location
print("Original changed list is: ", numbers)
del numbers[3] #del is inbuilt function that will delete 3rd element
print("The changed list is: ", numbers)

```

More about Lists and Tuples



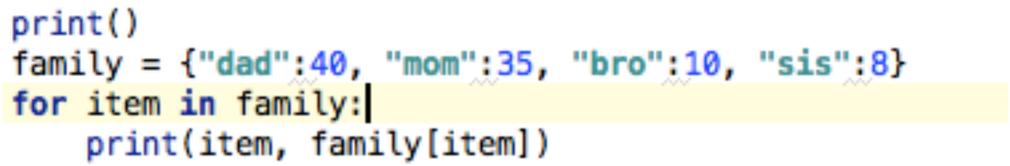
```
# list

data = [21,32,23]
print("original data:", data)

# lets append something on the list

data.append(35)
print("calling append with 35:", data)

words = ['Good', 'Bad', 'Ugly', 'Good']
print("Original words are: ", words)

# count number of times a specific word appears in a list
# count is a method defined in list data structure

print("How many time word, Good?: ", words.count("Good"))
print("How many time word, Excellent?: ", words.count("Excellent"))

morewords = ['Excellent', 'Marvellous'] #new list
print("After extend, word list is:", words.extend(morewords), words)  #extend the list
print("How many time word, Excellent?: ", words.count("Excellent"))

# lets create a list of Fruits and discover more methods or functions

fruits = ['apple', 'banana', 'strawberry', 'oranges']
print("My list of fruits is", fruits)
print("Where is the index of strawberry?", fruits.index("strawberry"))
print("lets insert something after banana", fruits.insert(3,"figs"), fruits)
print("lets reverse my list of fruits", fruits.reverse(), fruits)
print("lets sort my list of fruits", fruits.sort(), fruits)

fruitstuple = ("apple","banana","oranges")
#fruitstuple.insert("figs")
```

More about Dictionaries



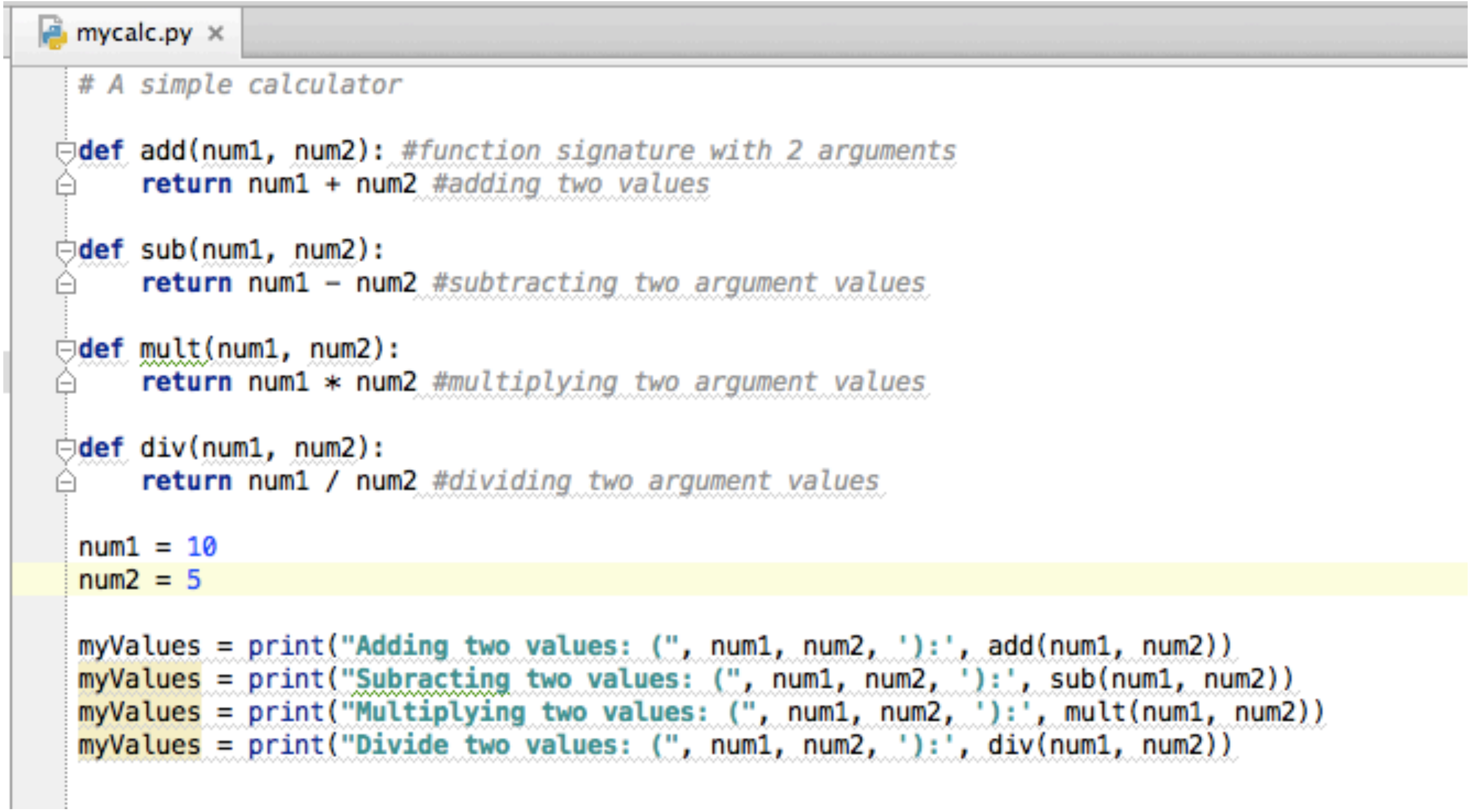
```python
# more about dictionaries
# key / value pair
# Dictionaries are unordered

fruits = {"1" : "apples", "2" : "oranges", "3" : "figs", "4" : "bananas", "tropical" : "watermelon"}
print("What fruits are in fridge?", fruits)
print("What fruits is in aile 2?", fruits["2"])
print("Any tropical fruit in the fridge?", fruits["tropical"])
print("What fruits are in fridge?", "tropical" in fruits) #looks for key
copyitems = fruits.copy()
print("Clear the fridge", fruits.clear(), fruits)
print("What fruits are in fridge?", copyitems)
```

Loop through a dictionary →



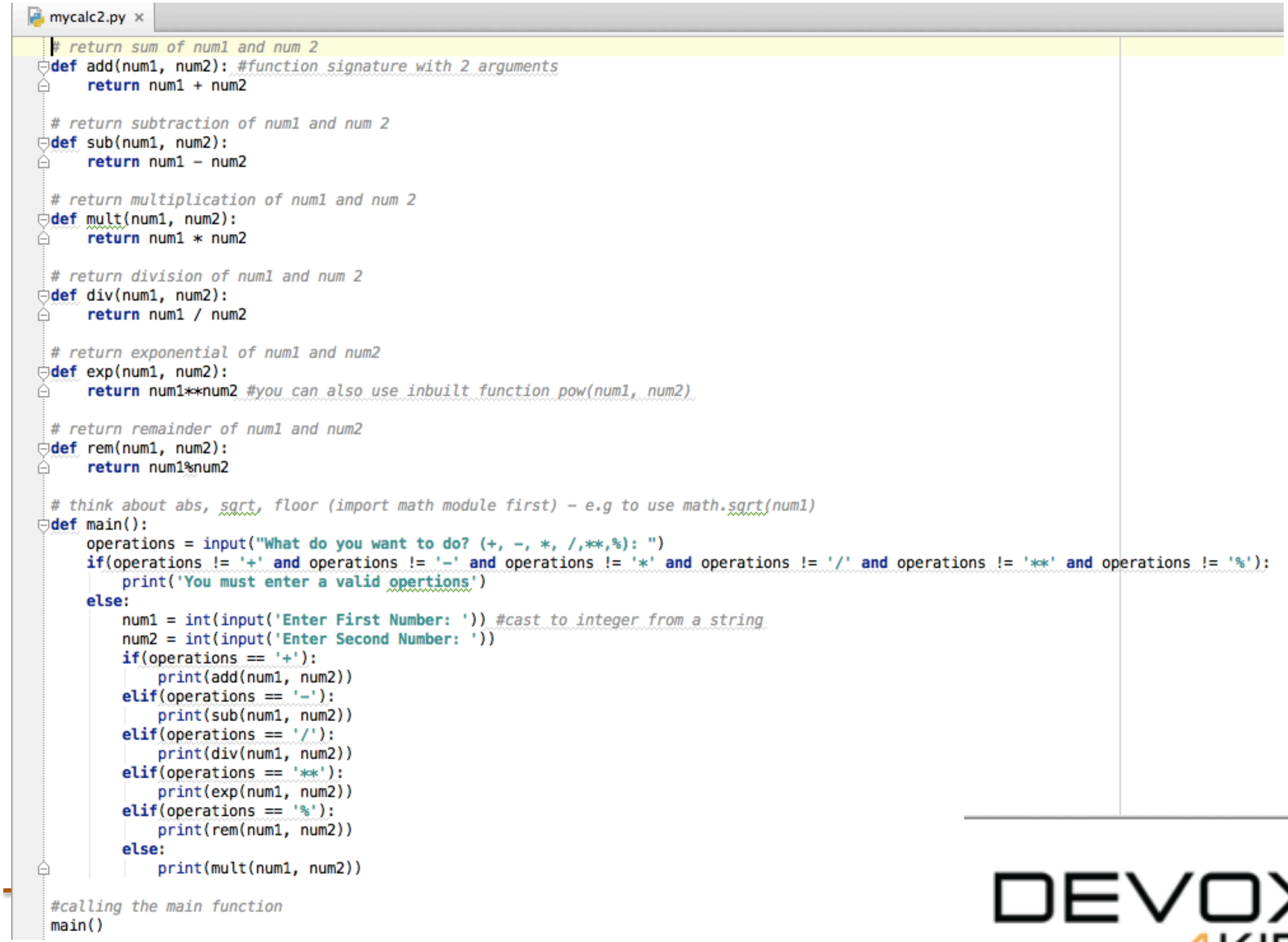
```python
print()
family = {"dad":40, "mom":35, "bro":10, "sis":8}
for item in family:
    print(item, family[item])
```

Introduction to Methods: Basic Calc



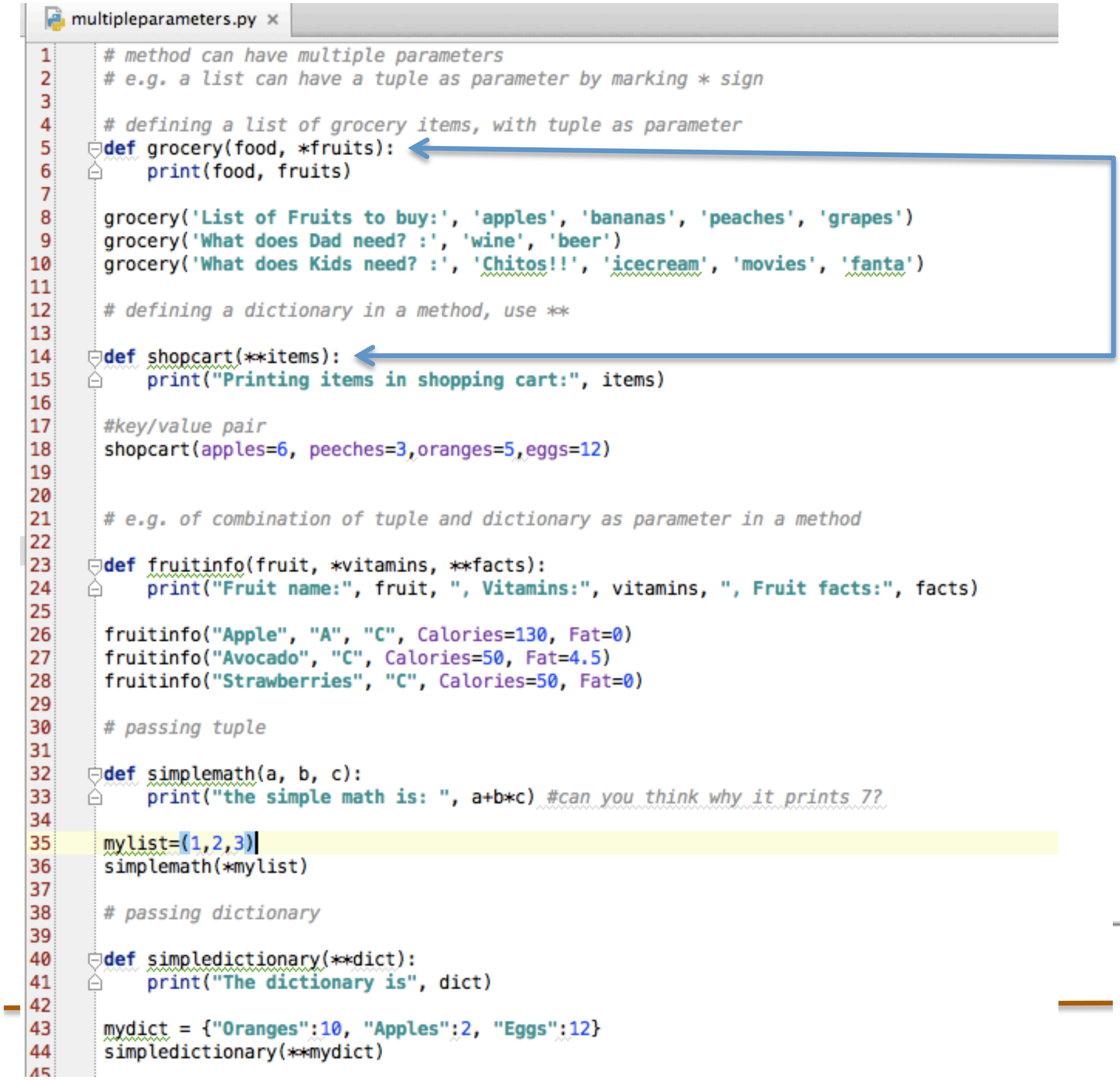
```python
# A simple calculator

def add(num1, num2): #function signature with 2 arguments
    return num1 + num2 #adding two values

def sub(num1, num2):
    return num1 - num2 #subtracting two argument values

def mult(num1, num2):
    return num1 * num2 #multiplying two argument values

def div(num1, num2):
    return num1 / num2 #dividing two argument values

num1 = 10
num2 = 5

myValues = print("Adding two values: (", num1, num2, '):', add(num1, num2))
myValues = print("Subracting two values: (", num1, num2, '):', sub(num1, num2))
myValues = print("Multiplying two values: (", num1, num2, '):', mult(num1, num2))
myValues = print("Divide two values: (", num1, num2, '):', div(num1, num2))
```

Advance Calculator:



```
# return sum of num1 and num 2
def add(num1, num2): #function signature with 2 arguments
    return num1 + num2

# return subtraction of num1 and num 2
def sub(num1, num2):
    return num1 - num2

# return multiplication of num1 and num 2
def mult(num1, num2):
    return num1 * num2

# return division of num1 and num 2
def div(num1, num2):
    return num1 / num2

# return exponential of num1 and num2
def exp(num1, num2):
    return num1**num2 #you can also use inbuilt function pow(num1, num2)

# return remainder of num1 and num2
def rem(num1, num2):
    return num1%num2

# think about abs, sqrt, floor (import math module first) - e.g to use math.sqrt(num1)
def main():
    operations = input("What do you want to do? (+, -, *, /,**,%): ")
    if(operations != '+' and operations != '-' and operations != '*' and operations != '/' and operations != '**' and operations != '%'):
        print('You must enter a valid opertions')
    else:
        num1 = int(input('Enter First Number: ')) #cast to integer from a string
        num2 = int(input('Enter Second Number: '))
        if(operations == '+'):
            print(add(num1, num2))
        elif(operations == '-'):
            print(sub(num1, num2))
        elif(operations == '/'):
            print(div(num1, num2))
        elif(operations == '**'):
            print(exp(num1, num2))
        elif(operations == '%'):
            print(rem(num1, num2))
        else:
            print(mult(num1, num2))

#calling the main function
main()
```

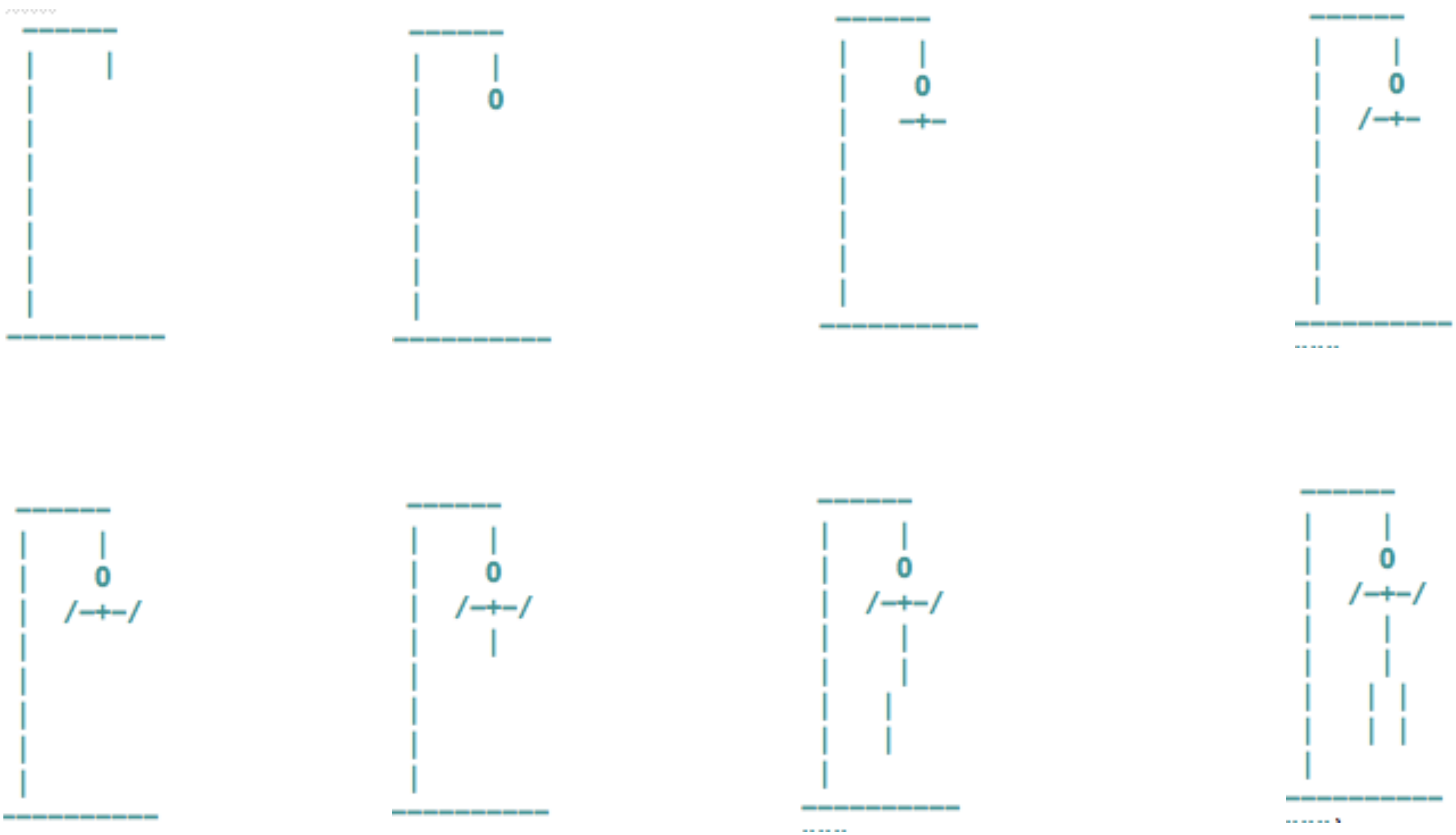
Passing Tuples and Dictionaries:



```python
# method can have multiple parameters
# e.g. a list can have a tuple as parameter by marking * sign

# defining a list of grocery items, with tuple as parameter
def grocery(food, *fruits):
    print(food, fruits)

grocery('List of Fruits to buy:', 'apples', 'bananas', 'peaches', 'grapes')
grocery('What does Dad need? :', 'wine', 'beer')
grocery('What does Kids need? :', 'Chitos!!', 'icecream', 'movies', 'fanta')

# defining a dictionary in a method, use **

def shopcart(**items):
    print("Printing items in shopping cart:", items)

#key/value pair
shopcart(apples=6, peeches=3,oranges=5,eggs=12)


# e.g. of combination of tuple and dictionary as parameter in a method

def fruitinfo(fruit, *vitamins, **facts):
    print("Fruit name:", fruit, ", Vitamins:", vitamins, ", Fruit facts:", facts)

fruitinfo("Apple", "A", "C", Calories=130, Fat=0)
fruitinfo("Avocado", "C", Calories=50, Fat=4.5)
fruitinfo("Strawberries", "C", Calories=50, Fat=0)

# passing tuple

def simplemath(a, b, c):
    print("the simple math is: ", a+b*c) #can you think why it prints 7?

mylist=(1,2,3)
simplemath(*mylist)

# passing dictionary

def simpledictionary(**dict):
    print("The dictionary is", dict)

mydict = {"Oranges":10, "Apples":2, "Eggs":12}
simpledictionary(**mydict)
```

Passing tuples and dictionaries as parameters

Lets Build & Play Hangman!



```
 ------
 |    |
 |
 |
 |
 |
 |
 |
 |
----------
```

```
 ------
 |    |
 |    0
 |
 |
 |
 |
 |
 |
----------
```

```
 ------
 |    |
 |    0
 |   -+-
 |
 |
 |
 |
 |
----------
```

```
 ------
 |    |
 |    0
 |  /-+-
 |
 |
 |
 |
 |
----------
```

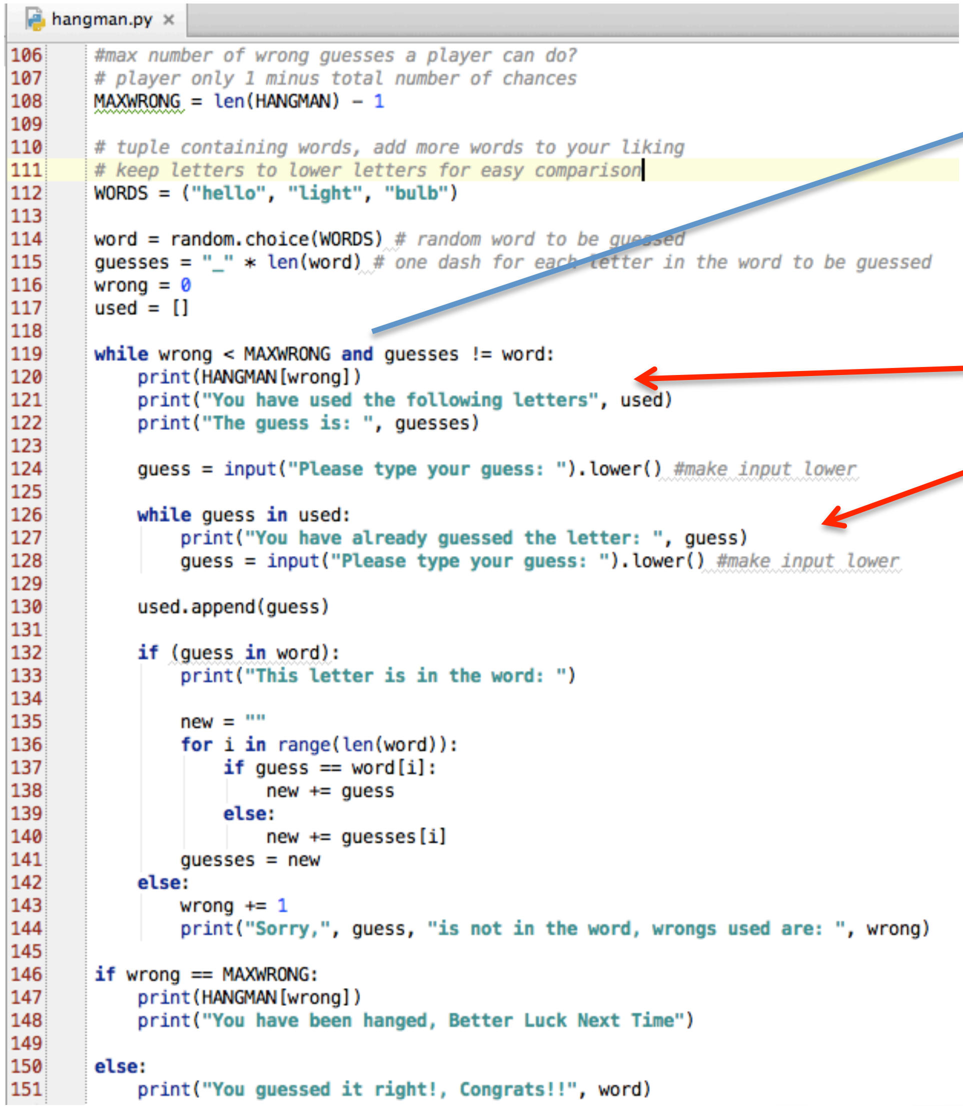
```
 ------
 |    |
 |    0
 |  /-+-/
 |
 |
 |
 |
 |
----------
```

```
 ------
 |    |
 |    0
 |  /-+-/
 |    |
 |
 |
 |
 |
----------
```

```
 ------
 |    |
 |    0
 |  /-+-/
 |    |
 |    |
 |   |
 |   |
 |
----------
```

```
 ------
 |    |
 |    0
 |  /-+-/
 |    |
 |    |
 |   | |
 |   | |
 |
----------
```

Hangman!

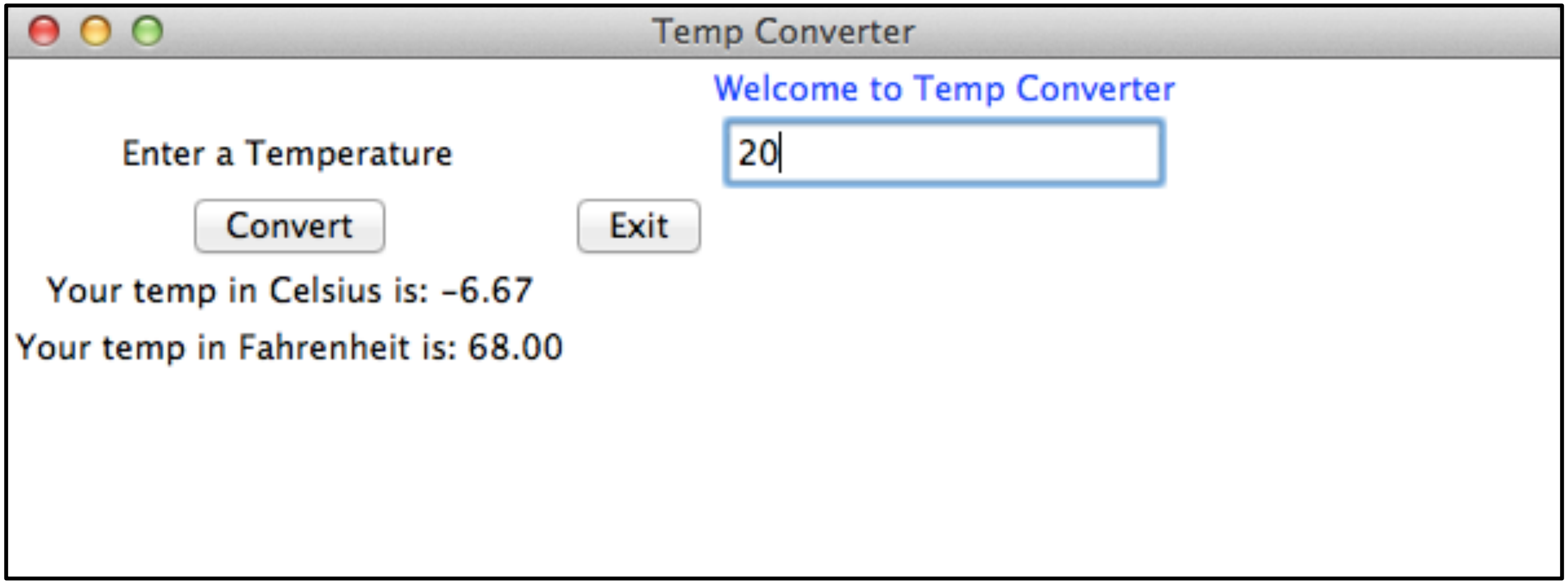


```python
#max number of wrong guesses a player can do?
# player only 1 minus total number of chances
MAXWRONG = len(HANGMAN) - 1

# tuple containing words, add more words to your liking
# keep letters to lower letters for easy comparison
WORDS = ("hello", "light", "bulb")

word = random.choice(WORDS) # random word to be guessed
guesses = "_" * len(word) # one dash for each letter in the word to be guessed
wrong = 0
used = []

while wrong < MAXWRONG and guesses != word:
    print(HANGMAN[wrong])
    print("You have used the following letters", used)
    print("The guess is: ", guesses)

    guess = input("Please type your guess: ").lower() #make input lower

    while guess in used:
        print("You have already guessed the letter: ", guess)
        guess = input("Please type your guess: ").lower() #make input lower

    used.append(guess)

    if (guess in word):
        print("This letter is in the word: ")

        new = ""
        for i in range(len(word)):
            if guess == word[i]:
                new += guess
            else:
                new += guesses[i]
        guesses = new
    else:
        wrong += 1
        print("Sorry,", guess, "is not in the word, wrongs used are: ", wrong)

if wrong == MAXWRONG:
    print(HANGMAN[wrong])
    print("You have been hanged, Better Luck Next Time")

else:
    print("You guessed it right!, Congrats!!", word)
```

‘and’ example

Nested statements

Introduction to Tkinter – Temp Converter App



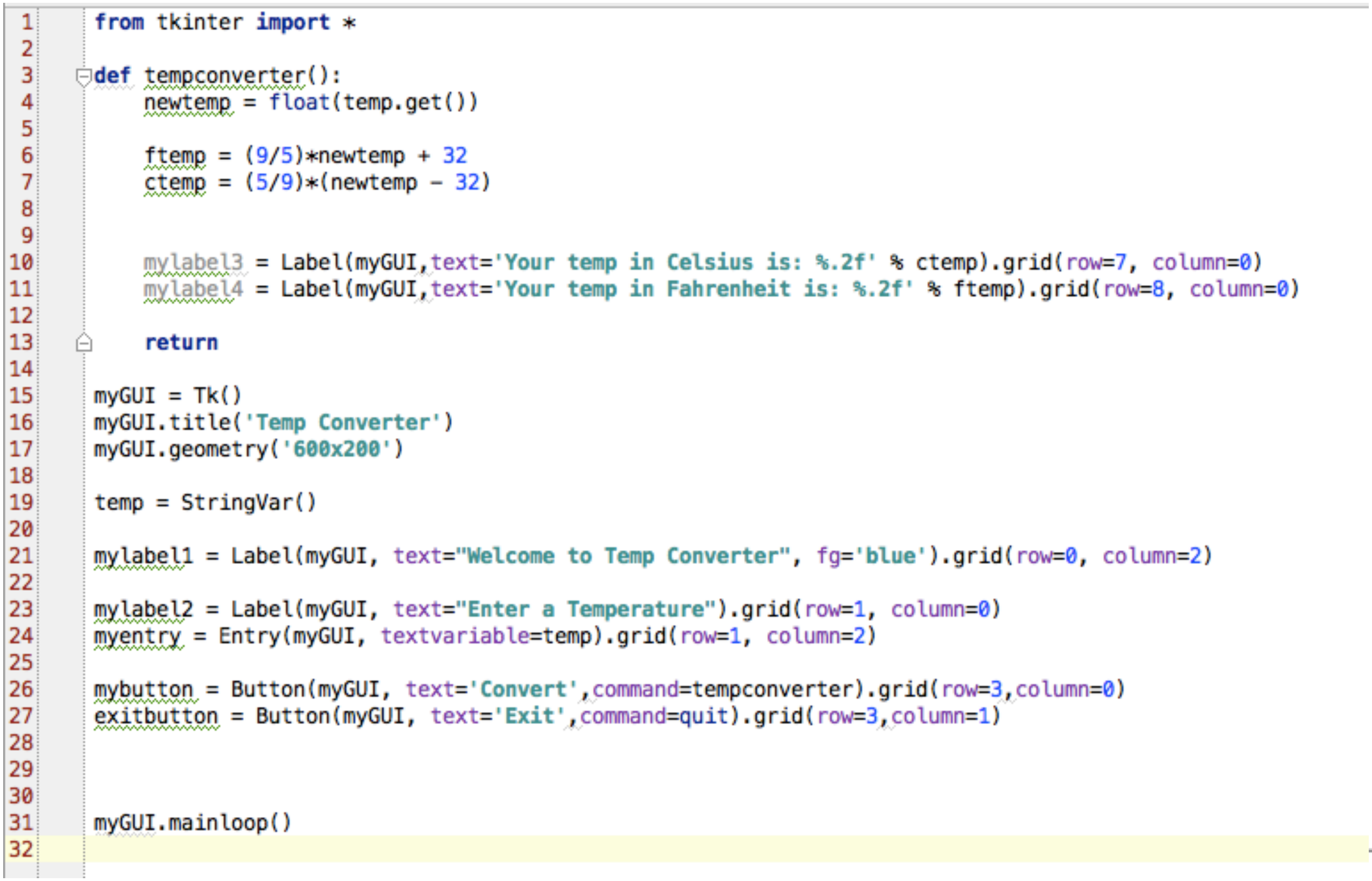
Welcome to Temp Converter

Enter a Temperature 20

Your temp in Celsius is: -6.67

Your temp in Fahrenheit is: 68.00

Temp Converter App



```
from tkinter import *

def tempconverter():
    newtemp = float(temp.get())

    ftemp = (9/5)*newtemp + 32
    ctemp = (5/9)*(newtemp - 32)


    mylabel3 = Label(myGUI,text='Your temp in Celsius is: %.2f' % ctemp).grid(row=7, column=0)
    mylabel4 = Label(myGUI,text='Your temp in Fahrenheit is: %.2f' % ftemp).grid(row=8, column=0)

    return

myGUI = Tk()
myGUI.title('Temp Converter')
myGUI.geometry('600x200')

temp = StringVar()

mylabel1 = Label(myGUI, text="Welcome to Temp Converter", fg='blue').grid(row=0, column=2)

mylabel2 = Label(myGUI, text="Enter a Temperature").grid(row=1, column=0)
myentry = Entry(myGUI, textvariable=temp).grid(row=1, column=2)

mybutton = Button(myGUI, text='Convert',command=tempconverter).grid(row=3,column=0)
exitbutton = Button(myGUI, text='Exit',command=quit).grid(row=3,column=1)



myGUI.mainloop()

```