Lists

In [2]:

```python
# declaring a list of numbers
nums = [5,10,15.2,20]
print(nums)
```

```
[5, 10, 15.2, 20]
```

In [3]:

```python
# accessing elements within a list
print(nums[1])          # will output the value at index 1
num = nums[2]           # saves index value 2 into num
print(num)              # prints value assigned to num
```

```
10
15.2
```

In [4]:

```python
# declaring a list of mixed data types
num = 4.3
data = [num,"word",True]   #Various data
print(data)
```

```
[4.3, 'word', True]
```

In [5]:

```python
# understanding list without lists
data = [5,"book",[34,"hello"],True]    #list can hold any type
print(data)
print(data[2])
```

```
[5, 'book', [34, 'hello'], True]
[34, 'hello']
```

In [6]:

```python
# using double bracket notation to access lists within lists
print(data[2][0])      #will output 34
inner_list = data[2]   # inner list
print(inner_list[1])
```

```
34
hello
```

In [7]:

```python
# changing values in a list through index
data = [5,10,15,20]
print(data)
data[0] = 100         #changing values at index 0
print(data)
```

```
[5, 10, 15, 20]
[100, 10, 15, 20]
```

In [10]:

```python
# understanding how lists are stored
a=[5,10]
b = a
print("a: {}\t b: {}".format(a,b))
print("Location a[0]: {}\t Location b[0]: {}".format(id(a[0]),id(b[0])))
a[0] = 20     #changing value at a[0] also changes value at b[0]
print("a: {}\t b: {}".format(a,b))
```

```
a: [5, 10]       b: [5, 10]
Location a[0]: 140715556153248   Location b[0]: 140715556153248
a: [20, 10]      b: [20, 10]
```

In [11]:

```python
# using [:] to copy a list
```

```
data=[5,10,15,20]
data_copy=data[:]     # a single colon copies the list
data[0] = 50
print("data: {}\t data_copy: {}".format(data,data_copy))
```

```
data: [50, 10, 15, 20]   data_copy: [5, 10, 15, 20]
```

In [12]:

```
"""
Exercise 1: Define a list of strings, where each string is a sport. Then output each
sport with the following line “I like to play {}”…
"""
sports=["Carrom","Football","Cricket","Chess","Tennis"]
print("I like to play {}.".format(sports[3]))
```

```
I like to play Chess.
```

In [18]:

```
"""
Exercise 2: For the following list, print out each item’s first letter. (output
should be ‘J’, ‘A’, ‘S’, ‘K’)
names = [‘John’, ‘Abraham’, ‘Sam’, ‘Kelly’]
"""
names = ['John', 'Abraham', 'Sam', 'Kelly']
print(" '{}','{}','{}','{}'".format(names[0][0],names[1][0],names[2][0],names[3][0],))
```

```
 'J','A','S','K'
```

For Loops

In [19]:

```
# for loop simple code
for num in range(5):
    print("Value: {}".format(num))
```

```
Value: 0
Value: 1
Value: 2
Value: 3
Value: 4
```

In [20]:

```
# providing start, stop and step for the range function
for num in range(2,10,2):
     print("Value: {}".format(num)) #will print all evens between 2 and 10
```

```
Value: 2
Value: 4
Value: 6
Value: 8
```

In [22]:

```
#printing all characters in a name using the 'in' keyword
name = 'Hrishikesh Devdikar'
for letter in name:
     print("Value: {}".format(letter))
```

```
Value: H
Value: r
Value: i
Value: s
Value: h
Value: i
Value: k
Value: e
Value: s
Value: h
```

```
Value:
Value: D
Value: e
Value: v
Value: d
Value: i
Value: k
Value: a
Value: r
```

In [23]:

```
# using the continue statement within a for loop
for num in range(5):
    if num == 3:
        continue    #goes back to the start of the loop rather than next statement
    print(num)
```

```
0
1
2
4
```

In [24]:

```
# breaking out of a loop using 'break' statement
for num in range(5):
    if num==3:
        break
    print(num)
```

```
0
1
2
```

In [26]:

```
# setting a placeholder using the 'pass' keyword
for i in range(5):
    # TODO: add code to print number
    pass
"""
It’s simply there so that we don’t have to write code within the loop just yet. It’s us
for framing out a program.
"""
```

In [28]:

```
"""
Exercise 3: Write a for loop that prints out all numbers from 1 to 100
that are divisible by three.
"""
for num in range(3,100,3):
    print(num)
```

```
3
6
9
12
15
18
21
24
27
30
33
36
39
42
45
48
```

```
51
54
57
60
63
66
69
72
75
78
81
84
87
90
93
96
99
```

In [32]:

```
"""
Exercise 4:Ask for user input, and write a for loop that will output all the
vowels within it.
"""
line= input("enter a statement:")
for vowels in 'aeiou':
    if vowels in line:
        print(vowels)
```

```
enter a statement:Hello World
e
o
```

While Loop

In [34]:

```
# simple while loop
health = 10
while health > 0:
    print(health)
    health -= 1      #forgetting this line will result in infite loop
```

```
10
9
8
7
6
5
4
3
2
1
```

In [35]:

```
# using two or more loops together (Nested loops)
for i in range(2):
    for j in range(3):
        print(i,j)
```

```
0 0
0 1
0 2
1 0
1 1
1 2
```

In [50]:

```
"""
Exercise 5:Write a while loop that continues to ask for user input and runs
until they type “quit”
"""

keep_going =True
while keep_going:
    word = input("Enter a word: ").lower()
    if word == "quit":
        keep_going =False
    else:
        keep_going = True
```

```
Enter a word: HARRY
Enter a word: POTTER
Enter a word: QUIT
```

In [2]:

```
"""
Exercise 6:Write a for loop within a while loop that will count from 0 to 5,but when it
"""
game_over = False
while not game_over:
    for i in range(5):
        if i ==3:
            game_over = True
            break
        print(i)
```

```
0
1
2
```

Working with Lists

In [3]:

```
# checking the number of items within a list
nums = [5,10,15]
length = len(nums)    #len() returns an integer
print(length)
```

```
3
```

In [6]:

```
# accessing specific items of a list with slices
print(nums[1:3])              # will output items in index 1 and 2
print(nums[:2])               # will output till index two is reached
print(nums[::2])              #step of two
print(nums[-2:])              #will print last two items in a list
```

```
[10, 15]
[5, 10]
[5, 15]
[10, 15]
```

In [7]:

```
# adding an item to the back of a list using append
nums = [10,20]
nums.append(5)
print(nums)
```

```
[10, 20, 5]
```

In [9]:

```
# adding a value to the beginning of the list
```

```
wor = ["ball","base"]
wor.insert(0,"glove")     # first number is the index, second is the value
```

In [16]:

```
# using pop to remove items and saving to a variable to use later
items=[5,"ball",True]
print(items.pop())    #by default removes the last item
removed_item = items.pop(0)
print(removed_item,"\n",items)
```

```
True
5
 ['ball']
```

In [18]:

```
# using the remove method with a try and except
sports = ["baseball", "soccer", "football", "chess"]
try:
    sports.remove("soccer")
except:
    print("That item does not exist")
print(sports)
```

```
['baseball', 'football', 'chess']
```

In [21]:

```
# using min,max and sum
nums=[5,3,9]
print(min(nums))
print(max(nums))
print(sum(nums))
```

```
3
9
17
```

In [22]:

```
# using sorted on lists for numerical and alphabetical data
nums=[5,8,0,2]
sorted_nums = sorted(nums)
print(nums,sorted_nums)
#does not alter the original list
```

```
[5, 8, 0, 2] [0, 2, 5, 8]
```

In [23]:

```
#sorting a list with .sort() in-place
nums=[5,0,8,2]
nums.sort()           #alters the original
print(nums)
```

```
[0, 2, 5, 8]
```

In [25]:

```
# using conditional statement on a list
names=["Jack", "Robert", "Mary"]
if "Mary" in names:
    print("Mary found")
if "Jimmy" not in names:
    print("Jimmy not found")
```

```
Mary found
Jimmy not found
```

In [26]:

```
# using conditionals to see if a list is empty
nums =[]
if not nums:        # could also say 'if nums ==[]'
    print("empty")
```

```
empty
```

In [27]:

```python
# using a for loop to print all items in a list
sports = ["Baseball","Chess","Football","Tennis"]
for sport in sports:
    print(sport)
```

```
Baseball
Chess
Football
Tennis
```

In [28]:

```python
# using the while loop to remove a certain value
names = ["Bob","Jack","Rob","Bob","Robert"]
while 'Bob' in names:
    names.remove('Bob')
print(names)
```

```
['Jack', 'Rob', 'Robert']
```

In [32]:

```python
"""
Exercise 7:  Remove all duplicates from the list below. Hint: Use the
.count() method. The output should be [‘Bob’, ‘Kenny’, ‘Amanda’]
"""
names = ['Bob', 'Kenny', 'Amanda', 'Bob', 'Kenny']
for name in names:
    counted = names.count(name)
    if counted > 1:
        new_list = names.remove(name)
print(names)
```

```
['Amanda', 'Bob', 'Kenny']
```

In [37]:

```python
"""
Exercise 8:   Use a while loop to continually ask the user to input a word, until
they type “quit ”. Once they type a word in, add it to a list. Once they quit the
loop, use a for loop to output all items within the list.
"""
keep_running = True
word=[]
while keep_running:
    ask = input("Enter a word").lower()
    word.append(ask)
    if ask == "quit":
        keep_running =False
for words in word:
    print(words)
```

```
Enter a wordnever
Enter a wordgive
Enter a wordup
Enter a worddont
Enter a wordquit
never
give
up
dont
quit
```

Project : Creating Hangman

In [ ]:

```
"""
the steps needed to
play the game:
1. Select a word to play with.
2. Ask user for input.
3. Check if guess is correct.
  a. If it is, show the letter in the proper place.
  b. If it isn’t, lose a life.
4. Continue steps 2 and 3 until one of the following occurs:
  a. The user guesses the word correctly.
  b. The user loses all their lives.

"""
```

In [82]:

```
# import additional functions
from random import choice
from IPython.display import clear_output

# declare game variables
words = ['tree', 'basket', 'chair', 'paper', 'python']
word = choice(words)
guessed, lives, game_over = [], 7, False

# create a list of underscores to the length of the word
guesses = ['_ '] * len(word)

# create main game loop
while not game_over:
    # output game information
    hidden_word = ''.join(guesses)
    print('Your guessed letters: {}'.format(guessed))
    print('Word to guess: {}'.format(hidden_word))
    print('Lives: {}'.format(lives))

    ans = input('Type quit or guess a letter: ').lower()

    clear_output()     # clear all previous output

    if ans == 'quit':
        print('Thanks for playing.')
        game_over = True
    elif ans in word and ans not in guessed:
        print('You guessed correctly!')

        # create a loop to change underscore to proper letter
        for i in range(len(word)):
            if word[i] == ans:
                guesses[i] = ans
    elif ans in guessed:
        print('You already guessed that. Try again.')
    else:                  # otherwise lose life
        lives -= 1
        print('Incorrect, you lost a life.')

    if ans not in guessed:
        guessed.append(ans)    # add guess to guessed list

    if lives <= 0:
        print('You lost all your lives, you lost!')
        game_over = True
    elif word == ''.join(guesses):
        print('Congratulations, you guessed it correctly!')
        game_over = True
```

```
You guessed correctly!
Congratulations, you guessed it correctly!
```

Challenges

In [83]:

```
"""
Use a for loop to build a pyramid of x’s. It should be
modular so that if you loop to 5 or 50, it still creates evenly spaced
rows. Hint: Multiply the string “x” by the row. For example, if you
loop to the range of 4, it should produce the following result:
>>> x
>>> x x
>>> x x x
"""
n = int(input("enter the number of times x needs to be printed: "))
for i in range(n+1):
      print(' ' * (n - i) + ' x' * i)
```

```
enter the number of times x needs to be printed: 5

     x
    x x
   x x x
  x x x x
 x x x x x
```

In [84]:

```
"""
Output Names: Write a loop that will iterate over a list of items
and only output items which have letters inside of a string. Take
the following list, for example, only “John” and “Amanda” should
be output:
>>> names = ['John', ' ', 'Amanda', 5]
"""
names = ["john"," ","amanda",5]
for i in names:
    if (type(i)==str):
        if i.strip() != '':
            print(i)
```

```
john
amanda
```

In [81]:

```
"""
 Given a list of temperatures that are in Celsius, write
a loop that iterates over the list and outputs the temperature converted
into Fahrenheit. Hint: The conversion is “F = (9/5) ∗ C + 32”:
>>> temps = [32, 12, 44, 29]

"""


temps = [32, 12, 44, 29]
faren=[]
for t in temps:
    f=(9/5)*t+32
    faren.append(f)
print(faren)
```

```
"""
alternative:
temps = [32, 12, 44, 29]

for i in range(len(temps)):
    temps[i] = (9/5) * temps[i] + 32

print(temps)
"""
```

```
[89.6, 53.6, 111.2, 84.2]
```