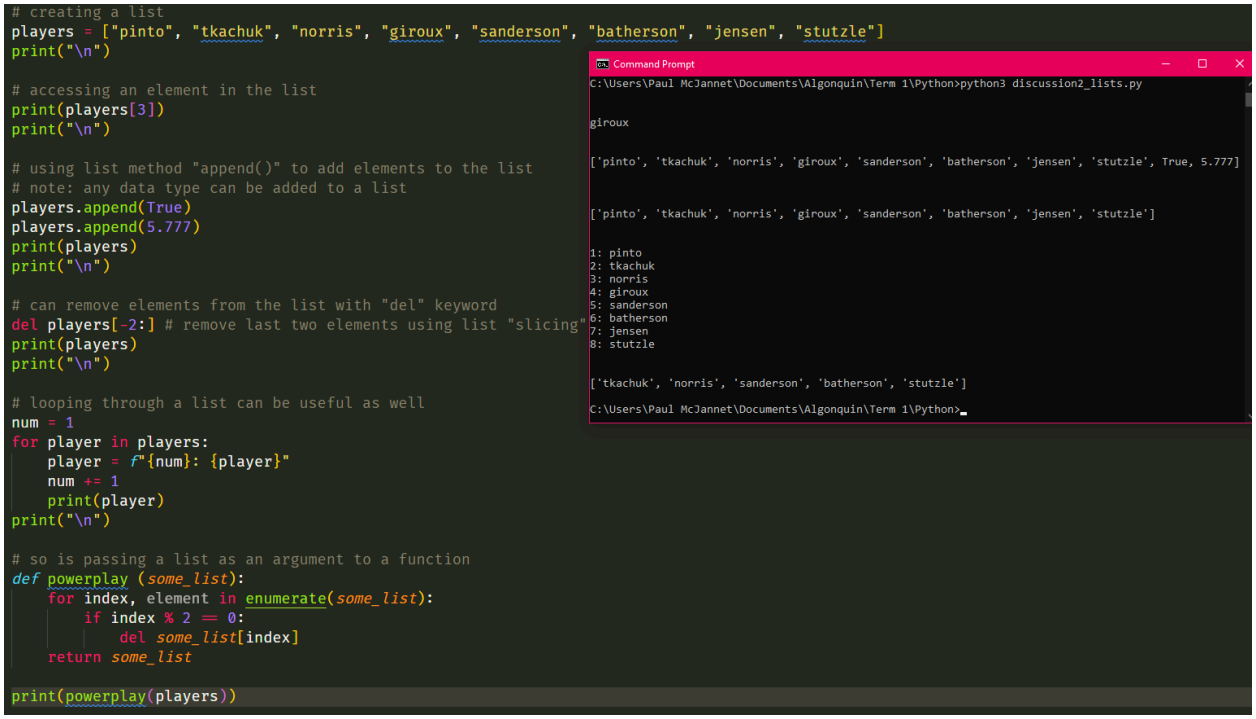
There are several different types of collections available in Python, but probably the two most common are lists and dictionaries. Both are objects that are used to house and organize data but each has its own unique rules and use cases.

Lists are very similar to JavaScript arrays: they are created with square brackets [], access elements inside using indexing, can contain elements of different types, and have several methods available to manipulate the data inside. For these reasons, lists are considered ordered and mutable. Here is an example of a list and the characteristics I described:



```python
# creating a list
players = ["pinto", "tkachuk", "norris", "giroux", "sanderson", "batherson", "jensen", "stutzle"]
print("\n")

# accessing an element in the list
print(players[3])
print("\n")

# using list method "append()" to add elements to the list
# note: any data type can be added to a list
players.append(True)
players.append(5.777)
print(players)
print("\n")

# can remove elements from the list with "del" keyword
del players[-2:] # remove last two elements using list "slicing"
print(players)
print("\n")

# looping through a list can be useful as well
num = 1
for player in players:
    player = f"{num}: {player}"
    num += 1
    print(player)
print("\n")

# so is passing a list as an argument to a function
def powerplay (some_list):
    for index, element in enumerate(some_list):
        if index % 2 == 0:
            del some_list[index]
    return some_list

print(powerplay(players))
```

```
Command Prompt
C:\Users\Paul McJannet\Documents\Algonquin\Term 1\Python>python3 discussion2_lists.py


giroux


['pinto', 'tkachuk', 'norris', 'giroux', 'sanderson', 'batherson', 'jensen', 'stutzle', True, 5.777]


['pinto', 'tkachuk', 'norris', 'giroux', 'sanderson', 'batherson', 'jensen', 'stutzle']


1: pinto
2: tkachuk
3: norris
4: giroux
5: sanderson
6: batherson
7: jensen
8: stutzle


['tkachuk', 'norris', 'sanderson', 'batherson', 'stutzle']

C:\Users\Paul McJannet\Documents\Algonquin\Term 1\Python>
```

Dictionaries are created using the squiggly brackets {} and store data in key:value pairs. Python dictionaries can have keys of any immutable type (they are not restricted to just strings), cannot have duplicate keys, and are accessed by key names instead of by index. There also exist several methods used for data manipulation and retrieval. Unlike lists, dictionaries are considered unordered since their elements are not indexed but referenced via keys. However, like lists, dictionaries are mutable since they can add and remove elements.

Here is an example of a dictionary and the characteristics I described:

```python
# creating a dictionary
# note: keys can be any immutable type such as strings, tuples or numbers
states = {"nevada": "nv", ("second",): "ny", 3: "mi"}
print(states)
print("\n")

# accessing an element in the dictionary by referring to its key
print(states[3]) # not index 3, but literally the key called 3!
print("\n")

# dictionaries are mutable -- updating elements uses simple assignment
states[3] = "michigan"
print(states)
print("\n")

# if a key doesn't already exist it will be added to the end by default
# (duplicate key names are not allowed)
states["florida"] = "fl"
print(states)
print("\n")

# simply check if something is in the dictionary
print("florida" in states) # True
print("\n")

# use "keys()" method to see all keys
states_keys = [key for key in states.keys()]
print(states_keys)
print("\n")

# use "values()" method to see all values
states_values = [value for value in states.values()]
print(states_values)
print("\n")

# use "items()" method to see all key:value pairs together
states_items = [item for item in states.items()]
print(states_items)
```

```
Command Prompt

C:\Users\Paul McJannet\Documents\Algonquin\Term 1\Python>python3 di
scussion2_dict.py
{'nevada': 'nv', ('second',): 'ny', 3: 'mi'}


mi


{'nevada': 'nv', ('second',): 'ny', 3: 'michigan'}


{'nevada': 'nv', ('second',): 'ny', 3: 'michigan', 'florida': 'fl'}



True


['nevada', ('second',), 3, 'florida']


['nv', 'ny', 'michigan', 'fl']


[('nevada', 'nv'), (('second',), 'ny'), (3, 'michigan'), ('florida'
, 'fl')]

C:\Users\Paul McJannet\Documents\Algonquin\Term 1\Python>
```

freecodecamp.org. (2020, August 27). *Intermediate Python Programming Course* [Video]. YouTube.

https://www.youtube.com/watch?v=HGOBQPFzWKo&t=68s

geeksforgeeks. (Last updated: 2024, September 6). *Python Dictionary Methods*. geeksforgeeks.

https://www.geeksforgeeks.org/python-dictionary-methods/

Microsoft. (2024). *_Visual Studio Code_* (Version 1.94.2) [Computer software].

https://code.visualstudio.com/

w3schools. (n.d.). *Python enumerate() Function.*

w3schools.https://www.w3schools.com/python/ref_func_enumerate.asp