clear() Removes all the elements from the dictionary

In [1]:
```
#The clear() method removes all the elements from a dictionary.
car = {
  "brand": "Ford",
  "model": "Mustang",
  "year": 1964
}

car.clear()

print(car)
```

```
{}
```

copy() Returns a copy of the dictionary

In [2]:
```
#The copy() method returns a copy of the specified dictionary.
car = {
  "brand": "Ford",
  "model": "Mustang",
  "year": 1964
}

x = car.copy()

print(x)
```

```
{'brand': 'Ford', 'model': 'Mustang', 'year': 1964}
```

fromkeys() Returns a dictionary with the specified keys and value

In [3]:
```
#fromkeys()     Returns a dictionary with the specified keys and value
x = ('key1', 'key2', 'key3')
y = 0

thisdict = dict.fromkeys(x, y)

print(thisdict)
```

```
{'key1': 0, 'key2': 0, 'key3': 0}
```

get() Returns the value of the specified key

In [4]:
```
#The get() method returns the value of the item with the specified key.
car = {
  "brand": "Ford",
  "model": "Mustang",
  "year": 1964
}

x = car.get("model")

print(x)
```

```
Mustang
```

items() Returns a list containing a tuple for each key value pair

In [ ]:
```
car = {
  "brand": "Ford",
  "model": "Mustang",
  "year": 1964
}

x = car.items()

print(x)
```

Definition and Usage

The items() method returns a view object. The view object contains the key-value pairs of the dictionary, as tuples in a list. The view object will reflect any changes done to the dictionary, see example below.

In [6]:
```
car = {
  "brand": "Ford",
  "model": "Mustang",
  "year": 1964
}

x = car.items()
```

```
print(x)
```

```
dict_items([('brand', 'Ford'), ('model', 'Mustang'), ('year', 1964)])
```

keys() Returns a list containing the dictionary's keys

In [7]:
```
#The keys() method returns a view object. The view object contains the keys of the dictionary, as a list.

#The view object will reflect any changes done to the dictionary, see example below.


car = {
  "brand": "Ford",
  "model": "Mustang",
  "year": 1964
}

x = car.keys()

print(x)
```

```
dict_keys(['brand', 'model', 'year'])
```

pop() Removes the element with the specified key

In [8]:
```
#The pop() method removes the specified item from the dictionary.
#The value of the removed item is the return value of the pop() method, see example below.
car = {
  "brand": "Ford",
  "model": "Mustang",
  "year": 1964
}

car.pop("model")

print(car)
```

```
{'brand': 'Ford', 'year': 1964}
```

popitem() Removes the last inserted key-value pair

In [9]:
```
#setdefault()   Returns the value of the specified key. If the key does not exist: insert the key, with the specified value
car = {
  "brand": "Ford",
  "model": "Mustang",
  "year": 1964
}

car.popitem()

print(car)
```

```
{'brand': 'Ford', 'model': 'Mustang'}
```

update() Updates the dictionary with the specified key-value pairs

In [10]:
```
car = {
  "brand": "Ford",
  "model": "Mustang",
  "year": 1964
}

car.update({"color": "White"})

print(car)
```

```
{'brand': 'Ford', 'model': 'Mustang', 'year': 1964, 'color': 'White'}
```

values() Returns a list of all the values in the dictionary

In [11]:
```
#The values() method returns a view object. The view object contains the values of the dictionary, as a list.
#The view object will reflect any changes done to the dictionary, see example below.


car = {
  "brand": "Ford",
  "model": "Mustang",
  "year": 1964
}

x = car.values()

print(x)
```

```
dict_values(['Ford', 'Mustang', 1964])
```