In [1]:
```
#DICTIONARIES IN PYTHON

"""->A dictionary is a more general version of a list
->here is a list of dates  of particular  month
->days=[31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31]
->indices will be used to access the dates
    for example days[0]=31
->Here is a dictionary of the daysof the month of the year
    days={'january':31,'February':28,'March':31
    'May':31, 'June':30, 'July':31,
    'August':31, 'September':30,
    'October':31, 'November':30,}
    here we use ['january'] which are the days to access the days
"""
```

```
C'"->A dictionary is a more general version of a list \n->here is a list of dates  of
 particular  month\n->days=[31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31]\n->indices
 will be used to access the dates \n    for example days[0]=31\n->Here is a dictionary
 of the daysof the month of the year\n    days={\'january\':31,\'February\':28,\'Marc
 h\':31\n    \'May\':31, \'June\':30, \'July\':31,\n    \'August\':31, \'September\':3
 0,\n    \'October\':31, \'November\':30,}\n    here we use [\'january\'] which are th
 e days to access the days\n'
```

In [2]:
```
#diclaration of dictionaries
D={'A':100,'B':200}
print(D)
#use the keys to access the values in the dictionary
print(D['A'])#the first value
print(D['B'])#the second value
```

```
{'A': 100, 'B': 200}
100
200
```

In [3]:
```
#changing values in the dictionary
D={'A':100,'B':200}
print(D)#the unchanged dictionary
D['A']=90
D['B']=60
print("the dict with new values:",D)#the dict with changed values
```

```
{'A': 100, 'B': 200}
the dict with new values: {'A': 90, 'B': 60}
```

In [4]:
```
#adding a new entry
D={'A':100,'B':200}
print(D)
```

```
    #to add a new entry,specify the key and the value
    D['C']=300
    #A new key 'C' will bw created and have the value 300
    print("The dict with new entry:",D)
    #NOTE->THIS CANT HAPPEN WHEN USING A LIST
```

```
{'A': 100, 'B': 200}
The dict with new entry: {'A': 100, 'B': 200, 'C': 300}
```

In [5]:
```
#TO delete a value form the dictionary use the del[key] operator
D={'A':100,'B':200,'C':300}
del D['C']#This will delete the value at key 'C'
print(D)
```

```
{'A': 100, 'B': 200}
```

In [6]:
```
#USING dictionaries as actual dictinaries
prog=True
while(prog):
    d={'dog':'has a tail and woofs',
        'cat':'says meow',
        'mouse':'chased by cats'}
    entry = input("Enter your word:")
    if entry=='end':
        prog=False#program termination
    else:
        print(d[entry])#print the meaning of the word
```

```
Enter your word:end
```

In [7]:
```
#creating a dictionary with zip() and dict() function using list
keys =['navin','kiran','harsh']#this are the keys to the dict
values=['python','java','kotlin']#this are the valuesto the dict
data =dict (zip(keys,values))
#dict() and zip(key,value)function are used to merge keys and values
#hence creating a dictionary
data['monica']='C#'
print(data)
```

```
{'navin': 'python', 'kiran': 'java', 'harsh': 'kotlin', 'monica': 'C#'}
```

In [8]:
```
#NESTED DICTIONARIES
"""in the below dict prog i have created
a dict and a list inside a dict"""
prog={'JS':'atom','c#':'VS','python':['pycharm','sublime','Vim'],
```

```
         'java':{'JSE':'netbeans','JEE':'Eclipse'}}
    #to access a specific value with the key python
    print(prog['python'])
    print(prog['python'][0])
    print(prog['python'][1])
    print(prog['python'][2])
    #to access those for java
    print(prog['java'])
    print(prog['java']['JSE'])
    print(prog['java']['JEE'])
    print(prog)
```

```
['pycharm', 'sublime', 'Vim']
pycharm
sublime
Vim
{'JSE': 'netbeans', 'JEE': 'Eclipse'}
netbeans
Eclipse
{'JS': 'atom', 'c#': 'VS', 'python': ['pycharm', 'sublime', 'Vim'], 'java': {'JSE': 'n
etbeans', 'JEE': 'Eclipse'}}
```

In [9]:
```
#copy data to another dictionary using the copy()function
key=['jan','feb','march']
val=[31,29,30]
data = dict(zip(key,val))#creating my dictionary with dict function
data_two=data.copy()
print("This is the copy of dict data:",data_two)
print(data)
#we can also check if a value is in a dictionary
entry=input("Enter the entry:")#gets the key entered
if entry in data:
    print("The value is:",data[entry])
else:
    print("Does not exist")
```

```
This is the copy of dict data: {'jan': 31, 'feb': 29, 'march': 30}
{'jan': 31, 'feb': 29, 'march': 30}
Enter the entry:31
Does not exist
```

In [10]:
```
#accessing keys and values uisng methods
#To access the key value pairs you would use the .item() method
key=['jan','feb','march']
val=[31,29,30]
data = dict(zip(key,val))#creating my dictionary with dict function
get_data=data.items()#returns the dictionary in a list if tuple pairs
print(get_data)#(key,value )->format returned
```

```
dict_items([('jan', 31), ('feb', 29), ('march', 30)])
```

In [11]:
```
#geting all values using the values() function
key=['jan','feb','march']
val=[31,29,30]
data = dict(zip(key,val))#creating my dictionary with dict function
v=data.values()#returns the  list of values in the dictionary
print(v)
```

```
dict_values([31, 29, 30])
```

In [12]:
```
#CONCANTENATION CAN NOT BE DONE ON DICTIONARIES
d={'1':1,'2':2}
e={'3':3,'4':4}
print(d+e)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-12-e306fd92de60> in <module>
      2 d={'1':1,'2':2}
      3 e={'3':3,'4':4}
----> 4 print(d+e)

TypeError: unsupported operand type(s) for +: 'dict' and 'dict'
```

In [ ]:
```
#sorting with dictionaries
#bubble sort has been implemented
d={'a':1,'b':2,'c':3}
for i in d:
    for j in d:
        if d[i]>d[j]:
            temp=d[i]
            d[i]=d[j]
            d[j]=temp
print(d.values())
```

In [ ]:
```
#the update method is used to upadate the dictionary
d={'a':1,'b':2,'c':3}
print(d.items())
d.update({'d':4})
#syntax:dict.update({key:value})
print(d.items())
```

In [ ]:
```
#dictionary concantenation
dic1={1:10, 2:20}
dic2={3:30, 4:40}
dic3={5:50,6:60}
```

```
dic4={}
for i in (dic1,dic2,dic3):
    dic4.update(i)

print(dic4.items())
```

In [ ]:
```
#clear function
d={'a':1,'b':2,'c':3}
d.clear()
print(d)
```

In [ ]:
```
#pop function
d={'a':1,'b':2,'c':3}
print(d.pop('a'))
#syntax dict.pop(key)
```

In [ ]:
```
#user inputs into the dictionary
dic={}
for i in range(1,7):
    #initialize an index to the user input
    dic[i]=input("enter name and hobby")
print(dic.items())
```

In [ ]:

In [3]:
```
"""Write a program that repeatedly asks the
user to enter
product names and prices.
Store all of these
in a dictionary
whose keys are the product names and whose
values are the prices.
Whentheuserisdoneenteringproductsandprices,
allowthemtorepeatedlyenteraproduct
name and print the corresponding price or a message
if the product is not in the dictionary."""
#Key are the products
#values are the prices
def store():
    dic={}
    lis=[]#the list will store the dictionary keys
    #create an empty dictionary
    for i in range(1,3):
        key=input("enter product:")
        lis.append(key)
    print(lis)
```

In [ ]:

In [ ]: