Untitled

March 30, 2023

0.1 SETS

[4]:
```
## sets are represented by {} curly brackets.
```

[5]:
```
s={}
```

[6]:
```
type(s)
```

[6]:
```
dict
```

[7]:
```
## but in above its showing as dict
```

[16]:
```
s1={1,2,3,4,5,6,"punith",True}
```

[17]:
```
type(s1)
```

[17]:
```
set
```

[18]:
```
## now its showing as sets
## in presence of linear elements inside curly brackets then it is treated as␣
 ↪sets.
```

[22]:
```
## even sets is collection are also used to store elements like lst tuples
```

[20]:
```
s2={1,1,12,3,3,3,3,4,5,5,5,55,523,34}
```

[21]:
```
s2
```

[21]:
```
{1, 3, 4, 5, 12, 34, 55, 523}
```

[25]:
```
## sets removes duplicates from datasets. helps to find unique elements
```

[29]:
```
## sets can be converted into the list and tuples
```

[36]:
```
list(s2)
```

[36]:
```
[1, 34, 3, 4, 5, 523, 12, 55]
```

[31]:
```
tuple(s2)
```

[31]:
```
(1, 34, 3, 4, 5, 523, 12, 55)
```

[38]:
```
## convert list to sets and tuples into sets
```

[37]:
```
set(s2)
```

[37]:
```
{1, 3, 4, 5, 12, 34, 55, 523}
```

[39]:
```
t1=(1,2,3,4,5,6,6,5,8,7,1,2,3)
```

[40]:
```
set(t1)
```

[40]:
```
{1, 2, 3, 4, 5, 6, 7, 8}
```

[41]:
```
s4={1,2,3,4,[1,2,3,4]}
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[41], line 1
----> 1 s4={1,2,3,4,[1,2,3,4]}

TypeError: unhashable type: 'list'
```

[43]:
```
## sets cannot hold list  inside but it allows to store tuples
```

[45]:
```
s3={1,2,3,4,(1,2,3,4)}
```

[48]:
```
## it allows to hold the tuple inside the sets because tuples are immutable
```

[49]:
```
s6={"Punith","punith", 1,2,3,4}
```

[50]:
```
s6
```

[50]:
```
{1, 2, 3, 4, 'Punith', 'punith'}
```

[51]:
```
s7={"punith","punith",1,2,3,4,51,1,5}
```

[52]:
```
s7
```

[52]:
```
{1, 2, 3, 4, 5, 51, 'punith'}
```

[53]:
```
## it is case sensitive , the above example shows the function of case sensitive
```

[54]:
```
s8={"punith","red","green",1,2,3,4}
```

[56]:
```
s8[0]
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[56], line 1
----> 1 s8[0]

TypeError: 'set' object is not subscriptable
```

[57]:
```
## sets cannot do indexing like tuples and lst
```

[58]:
```
s8[::-1]
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[58], line 1
----> 1 s8[::-1]

TypeError: 'set' object is not subscriptable
```

[60]:
```
## sets do not allow to reverse and also slicing
```

[61]:
```
## but sets can be itterrated
```

[62]:
```
s8
```

[62]:
```
{1, 2, 3, 4, 'green', 'punith', 'red'}
```

[66]:
```
for i in s8:
    print(i)
```

```
1
2
3
4
green
red
punith
```

[67]:
```
for i in s8:
    print(i,type(i))
```

```
1 <class 'int'>
2 <class 'int'>
3 <class 'int'>
4 <class 'int'>
```

```
green <class 'str'>
red <class 'str'>
punith <class 'str'>
```

[ ]:

[68]:
```
## sets have many operations
```

[69]:
```
s8
```

[69]:
```
{1, 2, 3, 4, 'green', 'punith', 'red'}
```

[70]:
```
s8.add(67)
```

[71]:
```
s8
```

[71]:
```
{1, 2, 3, 4, 67, 'green', 'punith', 'red'}
```

[72]:
```
s8.add(3)
```

[73]:
```
s8
```

[73]:
```
{1, 2, 3, 4, 67, 'green', 'punith', 'red'}
```

[74]:
```
## the number 3 is already present in a sets so it wont repeat again and␣
↪follows uniqueness
```

[75]:
```
len(s8)
```

[75]:
```
8
```

[76]:
```
s8.pop()
```

[76]:
```
1
```

[77]:
```
## pop used to remove the elements
```

[ ]:

[78]:
```
s8.clear()
```

[79]:
```
s8
```

[79]:
```
set()
```

[80]:
```
## removed all the elements
```

[ ]:

[81]:
```
s9={1,2,3,4}
s10={1,2,3,5}
```

[83]:
```
s9.difference(s10)
```

[83]: {4}

[ ]:

[ ]:

[ ]:

[ ]: