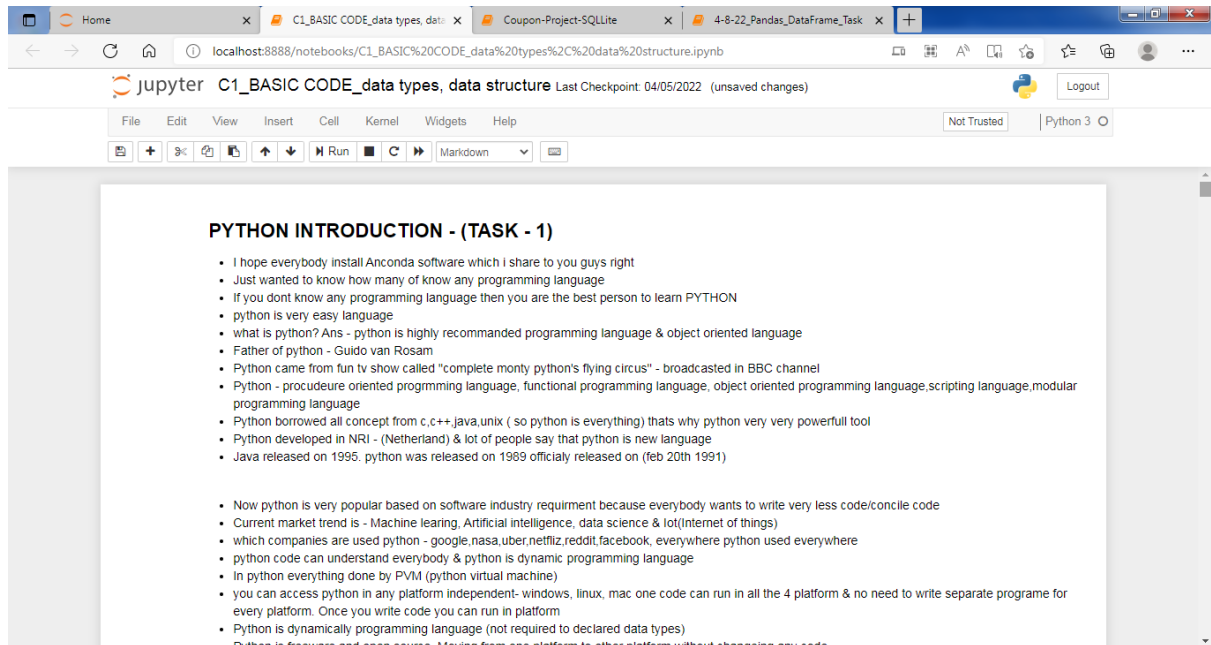


## Basic Code



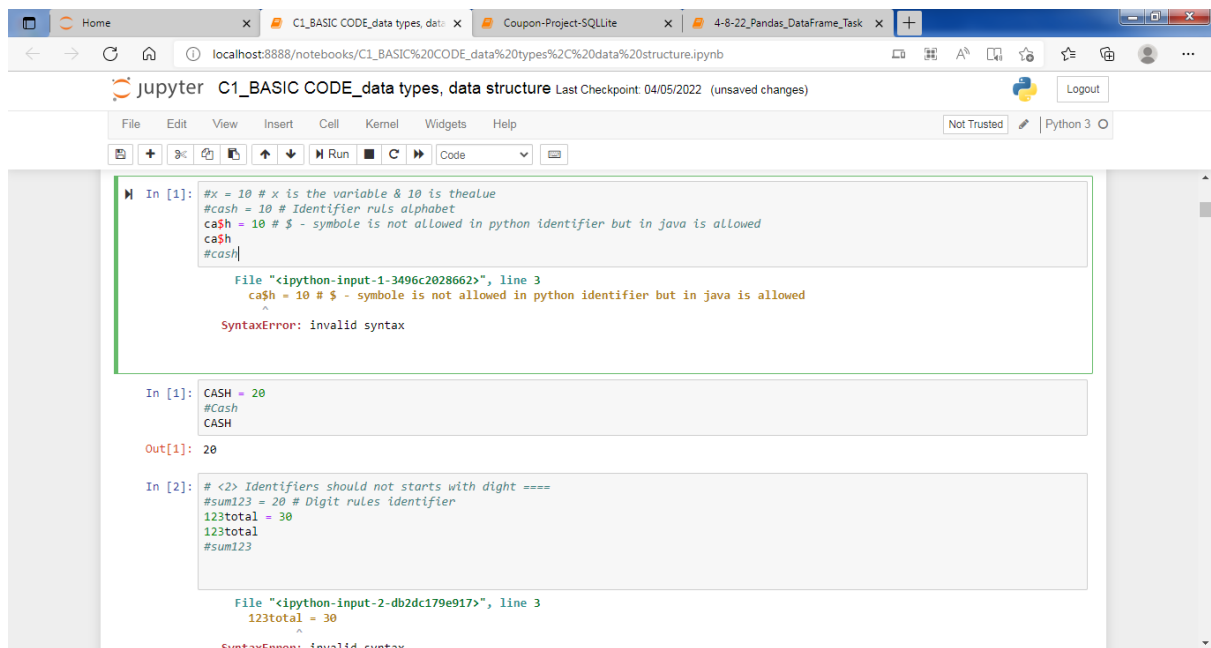
The screenshot shows a Jupyter Notebook interface with a browser window at the top. The notebook title is "C1\_BASIC CODE\_data types, data structure" and it shows "Last Checkpoint: 04/05/2022 (unsaved changes)". The interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help) and a toolbar with icons for file operations, running, and output. The main content area displays a document titled "PYTHON INTRODUCTION - (TASK - 1)" with two bulleted lists of information about Python.

### PYTHON INTRODUCTION - (TASK - 1)

- I hope everybody install Anaconda software which I share to you guys right
- Just wanted to know how many of you know any programming language
- If you don't know any programming language then you are the best person to learn PYTHON
- Python is very easy language
- What is Python? Ans - Python is highly recommended programming language & object-oriented language
- Father of Python - Guido van Rossum
- Python came from a fun TV show called "Complete Monty Python's Flying Circus" - broadcasted on BBC channel
- Python - procedure-oriented programming language, functional programming language, object-oriented programming language, scripting language, modular programming language
- Python borrowed all concepts from C, C++, Java, Unix (so Python is everything) that's why Python is very powerful tool
- Python developed in NRI - (Netherlands) & a lot of people say that Python is a new language
- Java released in 1995. Python was released in 1989 officially released on Feb 20th 1991

- Now Python is very popular based on software industry requirement because everybody wants to write very less code/concise code
- Current market trend is - Machine learning, Artificial intelligence, data science & IoT (Internet of Things)
- Which companies use Python - Google, NASA, Uber, Netflix, Reddit, Facebook, everywhere Python is used everywhere
- Python code can understand everybody & Python is a dynamic programming language
- In Python, everything is done by PVM (Python Virtual Machine)
- You can access Python on any platform independent - Windows, Linux, Mac. One code can run on all the 4 platforms & no need to write separate programs for every platform. Once you write code, you can run it on any platform
- Python is a dynamically typed programming language (not required to declare data types)
- Python is free and open source. Moving from one platform to another without changing any code



The screenshot shows the same Jupyter Notebook interface, but now displaying code execution errors. The first cell contains a Python script with several syntax errors, and the second cell shows the output of a valid code snippet.

```
In [1]: #x = 10 # x is the variable & 10 is the value
#cash = 10 # Identifier rules alphabet
ca$h = 10 # $ - symbol is not allowed in python identifier but in java is allowed
ca$h
#cash

File ~\ipython-input-1-3496c2028662, line 3
ca$h = 10 # $ - symbol is not allowed in python identifier but in java is allowed
^
SyntaxError: invalid syntax
```

```
In [1]: CASH = 20
#CASH
CASH

Out[1]: 20
```

```
In [2]: # <2> Identifiers should not start with digit ====
#sum123 = 20 # Digit rules identifier
123total = 30
123total
#sum123

File ~\ipython-input-2-db2dc179e917, line 3
123total = 30
^
SyntaxError: invalid syntax
```

Home x C1\_BASIC CODE\_data types, data structure x Coupon-Project-SQLite x 4-8-22\_Pandas\_DataFrame\_Task x +

localhost:8888/notebooks/C1\_BASIC%20CODE\_data%20types%20data%20structure.ipynb

jupyter C1\_BASIC CODE\_data types, data structure Last Checkpoint: 04/05/2022 (unsaved changes) Logout

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3

```
In [4]: TOTAL = 20
TOTAL
Out[4]: 20

In [6]: def = 4.6
def
File "<ipython-input-6-74b2d234418d>", line 1
def = 4.6
^
SyntaxError: invalid syntax

In [5]: if = 780
if
File "<ipython-input-5-f54330b147fc>", line 1
if = 780
^
SyntaxError: invalid syntax

In [ ]: DEF = 5.6
DEF

In [ ]: # <4> Keywords can not be assigned as identifier
```

Home x C1\_BASIC CODE\_data types, data structure x Coupon-Project-SQLite x 4-8-22\_Pandas\_DataFrame\_Task x +

localhost:8888/notebooks/C1\_BASIC%20CODE\_data%20types%20data%20structure.ipynb

jupyter C1\_BASIC CODE\_data types, data structure Last Checkpoint: 04/05/2022 (unsaved changes) Logout

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3

```
In [7]: DEF = 5.6
DEF
Out[7]: 5.6

In [8]: # <4> Keywords can not be assigned as identifier
#if = 10 # if is keyword
#DEF = 20 # def is keyword
for = 50
#DEF
#if
for
File "<ipython-input-8-0581332fa669>", line 4
for = 50
^
SyntaxError: invalid syntax

In [9]: def = 30
def
File "<ipython-input-9-d7d2371477df>", line 1
def = 30
^
SyntaxError: invalid syntax

In [ ]: if = 30
```

The screenshot shows a Jupyter Notebook titled "C1\_BASIC CODE\_data types, data structure". The notebook contains several code cells:

- In [10]:** A code cell containing:
 

```
if = 30
if
```

 This cell has executed and produced an error message:
 

```
File <ipython-input-10-23187ed1a6a1>, line 1
    if = 30
       ^
SyntaxError: invalid syntax
```
- In [11]:** A code cell containing:
 

```
IF = 56
IF
```

 The output of this cell is:
 

```
Out[11]: 56
```
- In [ ]:** A code cell containing:
 

```
if = 60
if
```
- In [12]:** A code cell containing:
 

```
# S> NO Length limit in python Identifier
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%XXXXXXXXXX = 10
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%XXXXXXXXXX
```

 The output of this cell is:
 

```
Out[12]: 10
```
- In [13]:** A code cell containing:
 

```
_abc_def_gef = 10
_abc_def_gef
```

 The output of this cell is:
 

```
Out[13]: 10
```

A green box highlights the bottom two code cells (In [13] and Out[13]). At the very bottom of the image, there is a small text snippet: "Q & A for valid / Invalid identifier - 1>123AMX 2>Amx123 3>m12al 4>abcdef\_g ef 5>def 6>else 7>ELSE".

Home x C1\_BASIC CODE\_data types, data Coupon-Project-SQLite x 4-8-22\_Pandas\_DataFrame\_Task x +

localhost:8888/notebooks/C1\_BASIC%20CODE\_data%20types%2C%20data%20structure.ipynb

jupyter C1\_BASIC CODE\_data types, data structure Last Checkpoint: 04/05/2022 (unsaved changes) Logout

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3

In [14]: `#a = True # hash is used for comment  
a = True  
a`

Out[14]: True

In [15]: `#b = None  
b = none  
b`

-----  
NameError Traceback (most recent call last)  
<ipython-input-15-09bbc72edfb6> in <module>()  
1 #b = None  
----> 2 b = none  
3 b  
  
NameError: name 'none' is not defined

In [16]: `c = False  
#c = false  
c`

Out[16]: False

In [ ]: `# How to remembr all keywords --- (Interview questions)  
# KEYWORD is the module run from IMPORT class  
import keyword`

Home x C1\_BASIC CODE\_data types, data Coupon-Project-SQLite x 4-8-22\_Pandas\_DataFrame\_Task x +

localhost:8888/notebooks/C1\_BASIC%20CODE\_data%20types%2C%20data%20structure.ipynb

jupyter C1\_BASIC CODE\_data types, data structure Last Checkpoint: 04/05/2022 (unsaved changes) Logout

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3

In [17]: `# How to remembr all keywords --- (Interview questions)  
# KEYWORD is the module run from IMPORT class  
import keyword  
keyword.kwlist`

Out[17]: ['False',  
'None',  
'True',  
'and',  
'as',  
'assert',  
'async',  
'await',  
'break',  
'class',  
'continue',  
'def',  
'del',  
'elif',  
'else',  
'except',  
'finally',  
'for',  
'from',  
'if',  
'import',  
'in',  
'is',  
'lambda',  
'nonlocal',  
'not',  
'or',  
'raise',  
'return',  
'try',  
'while',  
'with',  
'yield']

In [ ]: `# write a coding to create index of these keywords --- IMP ---  
import pandas as pd # pandas is the module to create a dataframe  
df = pd.DataFrame(keyword.kwlist)`

Home x C1\_BASIC CODE\_data types, data structure x Coupon-Project-SQLite x 4-8-22\_Pandas\_DataFrame\_Task x +

localhost:8888/notebooks/C1\_BASIC%20CODE\_data%20types%2C%20data%20structure.ipynb

jupyter C1\_BASIC CODE\_data types, data structure Last Checkpoint: 04/05/2022 (unsaved changes) Logout

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3

PYTHON DATA TYPES // (14) - INBUILT DATA TYPES -

1>int 2>float 3>complex 4>bool 5>str 6>bytes 7>bytearray 8>range 9>list 10>tuple 11>set 12>frozenset 13>dict 14>None

- python provides some inbuilt function like -- <1> print() <2> type() <3> id()
- int,float,complex,boolen is not represent object # Tricky question
- except these 4 everythig object # Tricky question

NOTE - [In python all 14 data types are object only] \*Thats why we called as python is object oriented program

'hello world'

In [19]: # What is other inbuilt datatype available except 14 datatypes-

```
a = 10
#print('a')      # To find the variable
#print(a)        # To find the value of variable
#type(a) # To find the data type
id(a)            # To find an address of an object
```

Out[19]: 8791475023904

In [20]: b = 20
id(b)

Out[20]: 8791475024224

In [ ]: c = 20

Home x C1\_BASIC CODE\_data types, data structure x Coupon-Project-SQLite x 4-8-22\_Pandas\_DataFrame\_Task x +

localhost:8888/notebooks/C1\_BASIC%20CODE\_data%20types%2C%20data%20structure.ipynb

jupyter C1\_BASIC CODE\_data types, data structure Last Checkpoint: 04/05/2022 (unsaved changes) Logout

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3

Out[19]: 8791475023904

In [20]: b = 20
id(b)

Out[20]: 8791475024224

In [22]: c = 20
id(c)

Out[22]: 8791475024224

In [21]: a = 10
b = 10
id(a)

Out[21]: 8791475023904

In [23]: a = 10
a
id(a)

Out[23]: 8791475023904

int datatypes -

- INT Datatypes - The No without decimal point are called as INTEGRAL DATATYPES \*int datatype how many ways represent values in 3ways -

Home x C1\_BASIC CODE\_data types, data structure x Coupon-Project-SQLite x 4-8-22\_Pandas\_DataFrame\_Task x +

localhost:8888/notebooks/C1\_BASIC%20CODE\_data%20types%2C%20data%20structure.ipynb

jupyter C1\_BASIC CODE\_data types, data structure Last Checkpoint: a minute ago (unsaved changes) Logout

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3

```
#a = 1111 # value is declared
b = 0b111 # Now pvm convert value to binary value
b
#a

Out[24]: 7

In [25]: c = 0b1111
c

Out[25]: 15

In [27]: b = 0b222
File "<ipython-input-27-4f96d237fbfc>", line 1
      b = 0b222
          ^
SyntaxError: invalid token

In [26]: b = 0b1111 # Now pvm convert value to binary value
b

Out[26]: 15

In [ ]: #3. Octal form(Base 8)
#a = 111 # Value is declared
b = 0o11 # Now pvm covert value to octal value
b
#a
```

Home x C1\_BASIC CODE\_data types, data structure x +

localhost:8888/notebooks/C1\_BASIC%20CODE\_data%20types%2C%20data%20structure.ipynb

jupyter C1\_BASIC CODE\_data types, data structure Last Checkpoint: 2 minutes ago (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3

```
TypeError: 'tuple' object does not support item assignment

In [72]: t1
Out[72]: (10, 'amx', True, 5.8, 10)

In [73]: t1[0] = 20 # tuple immutable ( not changable ) e.g - kyc / adhar
-----
TypeError                                Traceback (most recent call last)
<ipython-input-73-65c6390ef6e5> in <module>
----> 1 t1[0] = 20 # tuple immutable ( not changable ) e.g - kyc / adhar

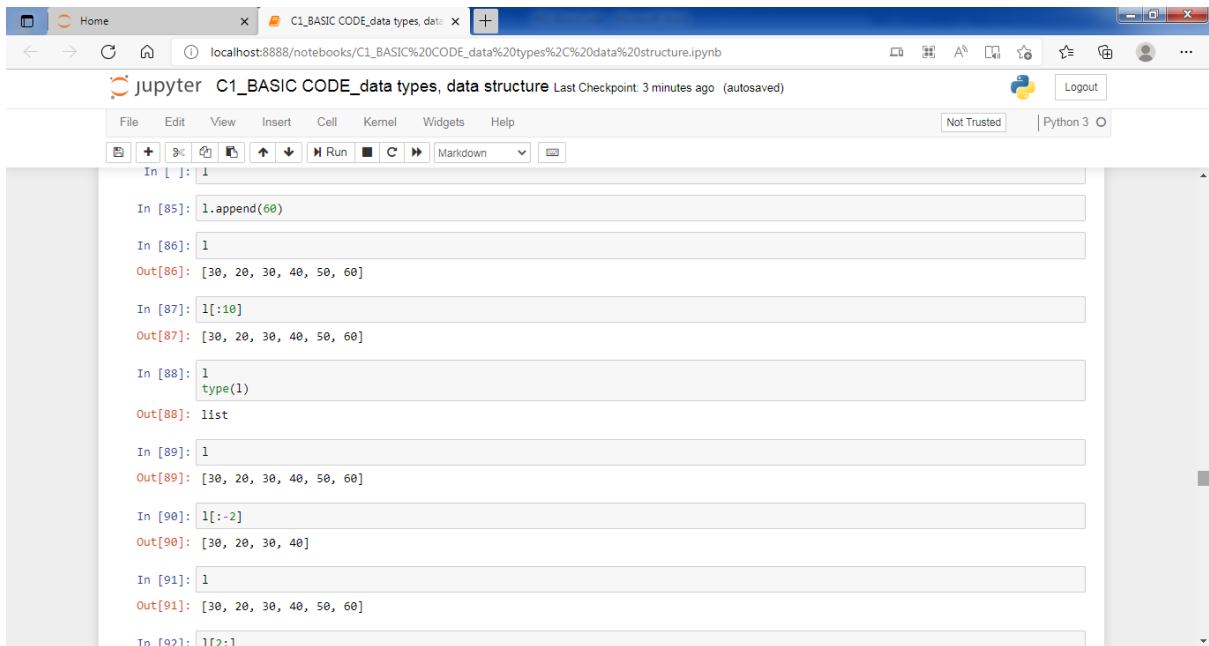
TypeError: 'tuple' object does not support item assignment

In [74]: t1
Out[74]: (10, 'amx', True, 5.8, 10)

In [75]: t1[0:3] # [[ IF YOU RIGHT SIDE THE CALCULATION WILL (Nth INDEX-1) (3-1) == UPTO 2ND INDEX ]]
Out[75]: (10, 'amx', True)

In [76]: t1[0:4]
Out[76]: (10, 'amx', True, 5.8)

In [77]: t1
```



The screenshot shows a Jupyter Notebook window titled "C1\_BASIC CODE\_data types, data structure". The browser address bar indicates the notebook is running on localhost:8888. The notebook interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help) and a toolbar with icons for file operations, running, and markdown. The code area contains the following cells:

```
In [ ]: l

In [85]: l.append(60)

In [86]: l
Out[86]: [30, 20, 30, 40, 50, 60]

In [87]: l[:10]
Out[87]: [30, 20, 30, 40, 50, 60]

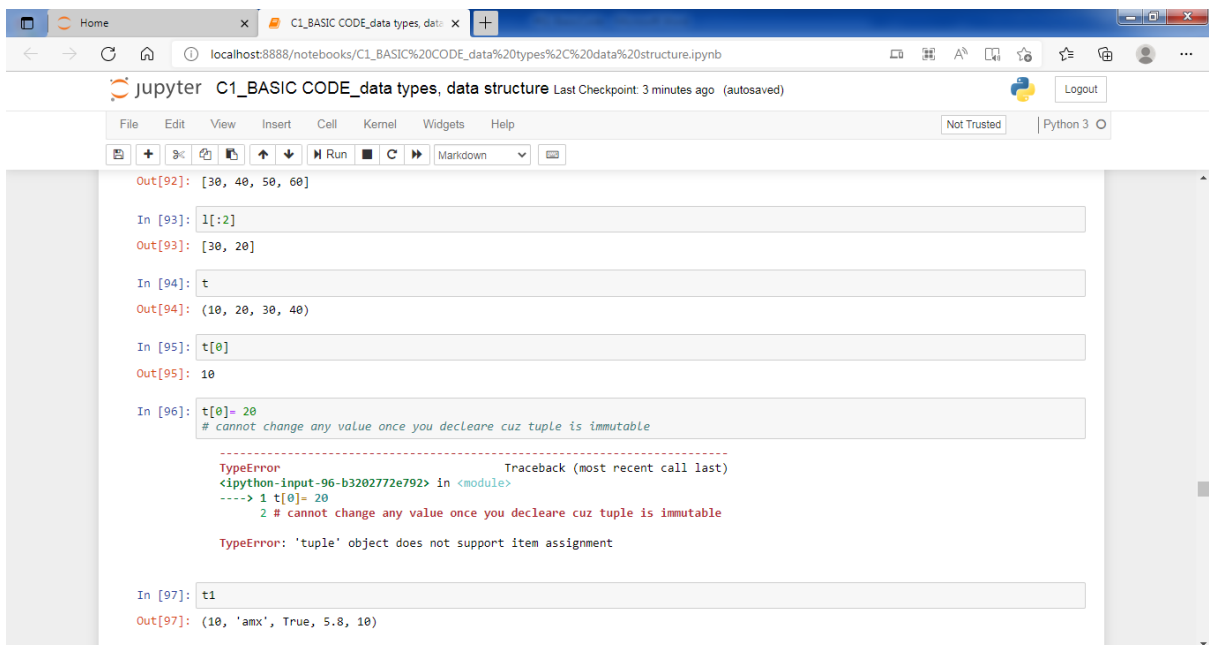
In [88]: l
type(l)
Out[88]: list

In [89]: l
Out[89]: [30, 20, 30, 40, 50, 60]

In [90]: l[:-2]
Out[90]: [30, 20, 30, 40]

In [91]: l
Out[91]: [30, 20, 30, 40, 50, 60]

In [92]: l[2:]
```



The screenshot shows the same Jupyter Notebook window, continuing from the previous state. The code area contains the following cells:

```
Out[92]: [30, 40, 50, 60]

In [93]: l[:2]
Out[93]: [30, 20]

In [94]: t
Out[94]: (10, 20, 30, 40)

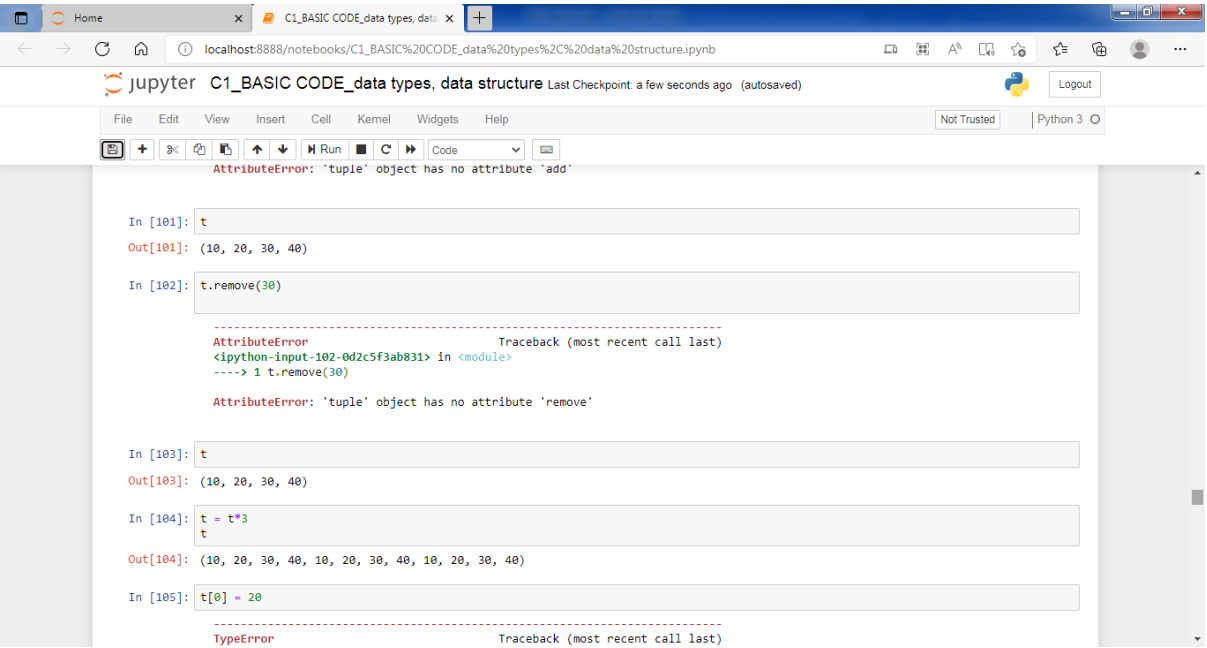
In [95]: t[0]
Out[95]: 10

In [96]: t[0]= 20
# cannot change any value once you declare cuz tuple is immutable

-----
TypeError                                 Traceback (most recent call last)
<ipython-input-96-b3202772e792> in <module>
----> 1 t[0]= 20
      2 # cannot change any value once you declare cuz tuple is immutable

TypeError: 'tuple' object does not support item assignment

In [97]: t1
Out[97]: (10, 'amx', True, 5.8, 10)
```





```
Home x C1_BASIC CODE_data types, data structure +
localhost:8888/notebooks/C1_BASIC%20CODE_data%20types%2C%20data%20structure.ipynb
jupyter C1_BASIC CODE_data types, data structure Last Checkpoint: a few seconds ago (autosaved)
File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3
In [103]: t
Out[103]: (10, 20, 30, 40)
In [104]: t = t*3
Out[104]: (10, 20, 30, 40, 10, 20, 30, 40, 10, 20, 30, 40)
In [105]: t[0] = 20
-----
TypeError                                Traceback (most recent call last)
<ipython-input-105-c4877d781a3b> in <module>
----> 1 t[0] = 20
TypeError: 'tuple' object does not support item assignment
In [106]: t1
Out[106]: (10, 'amx', True, 5.8, 10)
In [109]: t2 = t1 * 2 #in this case content has not changed but same t content repeted twice
In [110]: t2
Out[110]: (10, 'amx', True, 5.8, 10, 10, 'amx', True, 5.8, 10)
In [111]: t3 = (10,20,[2,6]) #is this valid one & u can declare List inside the tuple
```

```
Home x C1_BASIC CODE_data types, data structure +
localhost:8888/notebooks/C1_BASIC%20CODE_data%20types%2C%20data%20structure.ipynb
jupyter C1_BASIC CODE_data types, data structure Last Checkpoint: 24 minutes ago (autosaved)
File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3
In [62]: colors = "red", "blue", "green","yellow"
         colors
         rev = colors[::-1]
         rev
Out[62]: ('yellow', 'green', 'blue', 'red')
In [60]: colors = "red", "blue", "green","yellow"
         colors
         rev = colors[::-3]
         rev
Out[60]: ('yellow', 'red')
In [63]: colors = "red", "blue", "green","yellow"
         colors
         rev = colors[::-2]
         rev
Out[63]: ('yellow', 'blue')
In [114]: colors = "red", "green", "blue"
          colors
          rev = colors[::-1]
          rev
Out[114]: ('red', 'green')
In [115]: colors = "red", "green", "blue"
```

Home x C1\_BASIC CODE\_data types, data structure x +

localhost:8888/notebooks/C1\_BASIC%20CODE\_data%20types%2C%20data%20structure.ipynb

jupyter C1\_BASIC CODE\_data types, data structure Last Checkpoint: 25 minutes ago (unsaved changes) Logout

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3

In [117]: `rev = colors[::-2] # reversing order is allowed`  
`rev`

Out[117]: ('blue', 'red')

- in python which is the most common data types - range()
- range() datatypes represent a sequence of values
- always immutable
- range() datatypes we have multiple forms lets see one by one
- List insertion order is preserved but set insertion is not preserved

In [65]: `# FORM-1: range(10) -- represents values from 0 to 9 (python index start from 0)`  
`r = range(30)`  
`#type(r)`  
`r`

Out[65]: `range(0, 30)`

In [66]: `range(10.0, 11.5) # you cannot declare float argument`

```
-----
TypeError                                 Traceback (most recent call last)
<ipython-input-66-c1fde18be80a> in <module>()
----> 1 range(10.0, 11.5) # you cannot declare float argument

TypeError: 'float' object cannot be interpreted as an integer
-----
```

Home x C1\_BASIC CODE\_data types, data structure x +

localhost:8888/notebooks/C1\_BASIC%20CODE\_data%20types%2C%20data%20structure.ipynb

jupyter C1\_BASIC CODE\_data types, data structure Last Checkpoint: 28 minutes ago (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3

Out[117]: ('blue', 'red')

- in python which is the most common data types - range()
- range() datatypes represent a sequence of values
- always immutable
- range() datatypes we have multiple forms lets see one by one
- List insertion order is preserved but set insertion is not preserved

In [67]: `# FORM-1: range(10) -- represents values from 0 to 9 (python index start from 0)`  
`r = range(30)`  
`#type(r)`  
`r`

Out[67]: `range(0, 30)`

In [68]: `r1 = range(10)`  
`#type(r)`  
`r1`

Out[68]: `range(0, 10)`

In [66]: `range(10.0, 11.5) # you cannot declare float argument`

```
-----
TypeError                                 Traceback (most recent call last)
<ipython-input-66-c1fde18be80a> in <module>()
----> 1 range(10.0, 11.5) # you cannot declare float argument

TypeError: 'float' object cannot be interpreted as an integer
-----
```

Home x C1\_BASIC CODE\_data types, data structure x +

localhost:8888/notebooks/C1\_BASIC%20CODE\_data%20types%2C%20data%20structure.ipynb

jupyter C1\_BASIC CODE\_data types, data structure Last Checkpoint: 29 minutes ago (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3

```
19
20
21
22
23
24
25
26
27
28
29

In [70]: for i in r1: print(i)

0
1
2
3
4
5
6
7
8
9

In [126]: r[4]
Out[126]: 4

In [127]: r[0]
```

Home x C1\_BASIC CODE\_data types, data structure x +

localhost:8888/notebooks/C1\_BASIC%20CODE\_data%20types%2C%20data%20structure.ipynb

jupyter C1\_BASIC CODE\_data types, data structure Last Checkpoint: 30 minutes ago (unsaved changes) Logout

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3

```
In [70]: for i in r1: print(i)

0
1
2
3
4
5
6
7
8
9

In [71]: r[4]
Out[71]: 4

In [72]: r1[3]
Out[72]: 3

In [127]: r[0]
Out[127]: 0

In [73]: r[7]
Out[73]: 7

In [128]: r[4]
```

Home x C1\_BASIC CODE\_data types, data structure

localhost:8888/notebooks/C1\_BASIC%20CODE\_data%20types%2C%20data%20structure.ipynb

jupyter C1\_BASIC CODE\_data types, data structure Last Checkpoint: 33 minutes ago (unsaved changes) Logout

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3

set datatype

Difference between list & set --

- My requirement of the programme is -- Represent group of object as single entity where duplicates are allowed and order is important then we go for LIST
- I dont want duplicated, & order also not important then we will go for SET
- LIST are represent as [] && SET are represent as {}
- SET object does not support indexing or slicing
- Insertion order are not preserved, order & duplicates are not allowed
- Index concept are not allowed, heterogeneous object are allowed
- set is mutable
- add & remove method is use for SET but append method is used only for LIST

```
In [80]: s = {10,20,30,10,20,30} # is it allowed or not we will check
In [83]: sa = {10,20,30,10,50,50,50,40,40,20,30} # is it allowed or not we will check
In [78]: s #duplicates are not allowed
Out[78]: {10, 20, 30}
In [84]: sa
Out[84]: {10, 20, 30, 40, 50}
In [147]: s1 = {30,10,20,10,'NARESH',5.0,True}
s1
```

Home x C1\_BASIC CODE\_data types, data structure

localhost:8888/notebooks/C1\_BASIC%20CODE\_data%20types%2C%20data%20structure.ipynb

jupyter C1\_BASIC CODE\_data types, data structure Last Checkpoint: 35 minutes ago (unsaved changes) Logout

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3

```
Out[84]: {10, 20, 30, 40, 50}
In [147]: s1 = {30,10,20,10,'NARESH',5.0,True}
s1
Out[147]: {10, 20, 30, 5.0, 'NARESH', True}
In [145]: s1[1:4]
-----
TypeError                                 Traceback (most recent call last)
<ipython-input-145-7f6c29df96fb> in <module>
----> 1 s1[1:4]
TypeError: 'set' object is not subscriptable
In [85]: s1[2:5]
-----
TypeError                                 Traceback (most recent call last)
<ipython-input-85-81143d7abbbb> in <module>()
----> 1 s1[2:5]
TypeError: 'set' object is not subscriptable
In [148]: s
Out[148]: {10, 20, 30}
```

Home x C1\_BASIC CODE\_data types, data structure Last Checkpoint: a few seconds ago (autosaved)

localhost:8888/notebooks/C1\_BASIC%20CODE\_data%20types%2C%20data%20structure.ipynb

jupyter C1\_BASIC CODE\_data types, data structure Last Checkpoint: a few seconds ago (autosaved)

File Edit View Insert Cell Kernel Widgets Help

Checkpoint created: 10:27:26 Not Trusted Python 3

```
In [154]: s.add(True) #mutable
s
Out[154]: {True, 10, 20, 30}

In [155]: s.add('b')
s
Out[155]: {10, 20, 30, True, 'b'}

In [156]: s.add('c')

In [89]: sa.add('70','70')
-----
TypeError                                Traceback (most recent call last)
<ipython-input-89-07590995516> in <module>()
----> 1 sa.add('70','70')

TypeError: add() takes exactly one argument (2 given)
```

```
In [157]: s
Out[157]: {10, 20, 30, True, 'b', 'c'}

In [158]: sa
Out[158]: {10, 20, 30, 40, 50, '70'}
```

Home x C1\_BASIC CODE\_data types, data structure Last Checkpoint: 2 minutes ago (unsaved changes)

localhost:8888/notebooks/C1\_BASIC%20CODE\_data%20types%2C%20data%20structure.ipynb

jupyter C1\_BASIC CODE\_data types, data structure Last Checkpoint: 2 minutes ago (unsaved changes)

File Edit View Insert Cell Kernel Widgets Help

Not Trusted Python 3

```
AttributeError: 'frozenset' object has no attribute 'remove'

**DICTIONARY DATATYPES (dict)

• oxford dictionary -- in oxford dictionary words along with meaning is there
• i can say that key:values
• in the case of list,tuple,set,range,set,frozenset we represent individually & all objects are individually object right guys.
• i want to represent as group of object as pair example -- (rollno name, fruits price, mobileneno: name)
• dict is very important, very special category compare to all category
• duplicate keys are not allowed but values can be duplicate
• dict are represent as {} : you can assign with given operator
• keys & values both can be heterogeneous
• No such type of rule that all keys are integer types & values are string type
• keys & values can any type of object

In [168]: d = {100:'amx', 200:'shiv', 300:'nan'}
d
Out[168]: {100: 'amx', 200: 'shiv', 300: 'nan'}

In [91]: dsa = {1:'Tam', 2:'IT', 3:'SA'}
dsa
Out[91]: {1: 'Tam', 2: 'IT', 3: 'SA'}

In [92]: type(d)
Out[92]: int
```

Home x C1\_BASIC CODE\_data types, data structure x

localhost:8888/notebooks/C1\_BASIC%20CODE\_data%20types%2C%20data%20structure.ipynb

jupyter C1\_BASIC CODE\_data types, data structure Last Checkpoint: a few seconds ago (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Checkpoint created: 10:31:50 Not Trusted Python 3

Run Code

```
-----> print(type(s))

NameError: name 's' is not defined

In [174]: # i created empty dict
          d2 = {} # i want to add some element, so dict is mutable

In [175]: d2[100] = 'Naresh' # this is the metod to add an object in dictionary & no add,remove function are applicable in dict

In [99]: dsa[1]
Out[99]: 'ShahArsh'

In [97]: dsa[1] = 'ShahArsh'

In [98]: dsa[1]
Out[98]: 'ShahArsh'

In [176]: d2[200] = 'amx'

In [177]: d2[300] = 'ABC'

In [178]: d2
Out[178]: {100: 'Naresh', 200: 'amx', 300: 'ABC'}
```