

Strings continu...

In [2]: `print(dir(str))`

```
['__add__', '__class__', '__contains__', '__delattr__', '__dir__', '__doc__',
 '__eq__', '__format__', '__ge__', '__getattribute__', '__getitem__', '__getnewa
rgs__', '__gt__', '__hash__', '__init__', '__init_subclass__', '__iter__', '__l
e__', '__len__', '__lt__', '__mod__', '__mul__', '__ne__', '__new__', '__reduce
__', '__reduce_ex__', '__repr__', '__rmod__', '__rmul__', '__setattr__', '__siz
eof__', '__str__', '__subclasshook__', 'capitalize', 'casefold', 'center', 'cou
nt', 'encode', 'endswith', 'expandtabs', 'find', 'format', 'format_map', 'inde
x', 'isalnum', 'isalpha', 'isascii', 'isdecimal', 'isdigit', 'isidentifier', 'i
slower', 'isnumeric', 'isprintable', 'isspace', 'istitle', 'isupper', 'join',
'ljust', 'lower', 'lstrip', 'maketrans', 'partition', 'replace', 'rfind', 'rind
ex', 'rjust', 'rpartition', 'rsplit', 'rstrip', 'split', 'splitlines', 'startsw
ith', 'strip', 'swapcase', 'title', 'translate', 'upper', 'zfill']
```

In [3]: `a = "123"`
`a`

Out[3]: '123'

In [5]: `a[1:]# [start:stop:step]`

Out[5]: '23'

In [6]: `a[::-1]#[start:stop:-1]`

Out[6]: '321'

In [7]: `a = "apssdc"`
`a.capitalize()`

Out[7]: 'Apssdc'

In [8]: `help(str.capitalize)# help(str.method)`

Help on method_descriptor:

```
capitalize(self, /)
    Return a capitalized version of the string.
```

More specifically, make the first character have upper case and the rest lo
 wer
 case.

In [10]: `help(str)`

...

List:

- []
- list is mutable

```
In [14]: a = ["alekhya",7.8,23]
a
```

```
Out[14]: ['alekhya', 7.8, 23]
```

```
In [15]: type(a)
```

```
Out[15]: list
```

```
In [16]: a[0]
```

```
Out[16]: 'alekhya'
```

```
In [17]: a[1:]
```

```
Out[17]: [7.8, 23]
```

```
In [18]: a[::-1]
```

```
Out[18]: [23, 7.8, 'alekhya']
```

```
In [19]: print(dir(list))
```

```
['__add__', '__class__', '__contains__', '__delattr__', '__delitem__', '__dir__  
_', '__doc__', '__eq__', '__format__', '__ge__', '__getattribute__', '__getitem__  
_', '__gt__', '__hash__', '__iadd__', '__imul__', '__init__', '__init_subclass__  
_', '__iter__', '__le__', '__len__', '__lt__', '__mul__', '__ne__', '__new__',  
'_reduce__', '__reduce_ex__', '__repr__', '__reversed__', '__rmul__', '__setattr__  
_', '__setitem__', '__sizeof__', '__str__', '__subclasshook__', 'append', 'clear',  
'copy', 'count', 'extend', 'index', 'insert', 'pop', 'remove', 'reverse', 'sort']
```

```
In [20]: a
```

```
Out[20]: ['alekhya', 7.8, 23]
```

```
In [21]: b = [89,90]
a.append(b)
```

```
In [22]: a
```

```
Out[22]: ['alekhya', 7.8, 23, [89, 90]]
```

```
In [23]: a.append(67)
```

In [24]: a

Out[24]: ['alekhya', 7.8, 23, [89, 90], 67]

In [25]: a.pop()

Out[25]: 67

In [26]: a

Out[26]: ['alekhya', 7.8, 23, [89, 90]]

In [27]: a.remove(7.8)

In [28]: a

Out[28]: ['alekhya', 23, [89, 90]]

In [29]: a.remove(a[0])

In [30]: a

Out[30]: [23, [89, 90]]

In [31]: a.remove(0)

```
-----  
ValueError                                Traceback (most recent call last)  
<ipython-input-31-ae1fd24e9d23> in <module>  
----> 1 a.remove(0)  
  
ValueError: list.remove(x): x not in list
```

In [32]: a.pop(0)

Out[32]: 23

In [33]: a

Out[33]: [[89, 90]]

In [34]: a[0]

Out[34]: [89, 90]

In [36]: a[0][1]

Out[36]: 90

Tuple

- ()
- Tuple is immutable

```
In [49]: a = (1,2,3,90.9,"r",2)
```

```
In [50]: a
```

```
Out[50]: (1, 2, 3, 90.9, 'r', 2)
```

```
In [51]: type(a)
```

```
Out[51]: tuple
```

```
In [52]: a[0]
```

```
Out[52]: 1
```

```
In [53]: a[4]
```

```
Out[53]: 'r'
```

```
In [54]: print(dir(tuple))
```

```
['__add__', '__class__', '__contains__', '__delattr__', '__dir__', '__doc__',  
 '__eq__', '__format__', '__ge__', '__getattribute__', '__getitem__', '__getnewa  
rgs__', '__gt__', '__hash__', '__init__', '__init_subclass__', '__iter__', '__l  
e__', '__len__', '__lt__', '__mul__', '__ne__', '__new__', '__reduce__', '__red  
uce_ex__', '__repr__', '__rmul__', '__setattr__', '__sizeof__', '__str__', '__s  
ubclasshook__', 'count', 'index']
```

```
In [55]: a.index("r")
```

```
Out[55]: 4
```

```
In [56]: a
```

```
Out[56]: (1, 2, 3, 90.9, 'r', 2)
```

```
In [57]: a.index(3)
```

```
Out[57]: 2
```

```
In [58]: a.count(2)
```

```
Out[58]: 2
```

```
In [60]: help(tuple.count)
```

Help on method_descriptor:

```
count(self, value, /)
    Return number of occurrences of value.
```

Dictionary

- {}
- key:values--pair
- mutable
- we can access elemnts in the dictionary by using key

```
In [69]: d = {"fruits":["apples","banana","grapes"],"vegetables":["carrot","tomatoes","cabbage"]}
d
```

```
Out[69]: {'fruits': ['apples', 'banana', 'grapes'],
          'vegetables': ['carrot', 'tomatoes', 'cabbage']}
```

```
In [70]: d["fruits"]
```

```
Out[70]: ['apples', 'banana', 'grapes']
```

```
In [73]: d["fruits"][-1]
```

```
Out[73]: 'grapes'
```

```
In [74]: d["fruits"][2]
```

```
Out[74]: 'grapes'
```

```
In [75]: print(dir(dict))
```

```
['__class__', '__contains__', '__delattr__', '__delitem__', '__dir__', '__doc__',
 '__eq__', '__format__', '__ge__', '__getattribute__', '__getitem__', '__gt__',
 '__hash__', '__init__', '__init_subclass__', '__iter__', '__le__', '__len__',
 '__lt__', '__ne__', '__new__', '__reduce__', '__reduce_ex__', '__repr__',
 '__setattr__', '__setitem__', '__sizeof__', '__str__', '__subclasshook__', 'clear',
 'copy', 'fromkeys', 'get', 'items', 'keys', 'pop', 'popitem', 'setdefault',
 'update', 'values']
```

```
In [83]: a = dict()
b = {"fruits":"apl"}
```

```
In [85]: type(b)
         type(a)
```

```
Out[85]: dict
```

```
In [ ]: str()
        list()--[]
```

```
In [86]: d
```

```
Out[86]: {'fruits': ['apples', 'banana', 'grapes'],
          'vegetables': ['carrot', 'tomatoes', 'cabbage']}
```

```
In [87]: d.keys()
```

```
Out[87]: dict_keys(['fruits', 'vegetables'])
```

```
In [88]: d.values()
```

```
Out[88]: dict_values(['apples', 'banana', 'grapes'], ['carrot', 'tomatoes', 'cabbage'])
```

```
In [89]: d.items()
```

```
Out[89]: dict_items([('fruits', ['apples', 'banana', 'grapes']), ('vegetables', ['carrot', 'tomatoes', 'cabbage'])])
```

set

- {}
- unique

```
In [90]: a = {1,2,2,56,12}
```

```
In [91]: a
```

```
Out[91]: {1, 2, 12, 56}
```

```
In [92]: print(dir(set))
```

```
['__and__', '__class__', '__contains__', '__delattr__', '__dir__', '__doc__',  
 '__eq__', '__format__', '__ge__', '__getattribute__', '__gt__', '__hash__', '__  
iand__', '__init__', '__init_subclass__', '__ior__', '__isub__', '__iter__', '__  
_ixor__', '__le__', '__len__', '__lt__', '__ne__', '__new__', '__or__', '__rand  
__', '__reduce__', '__reduce_ex__', '__repr__', '__ror__', '__rsub__', '__rxor_  
__', '__setattr__', '__sizeof__', '__str__', '__sub__', '__subclasshook__', '__x  
or__', 'add', 'clear', 'copy', 'difference', 'difference_update', 'discard', 'i  
ntersection', 'intersection_update', 'isdisjoint', 'issubset', 'issuperset', 'p  
op', 'remove', 'symmetric_difference', 'symmetric_difference_update', 'union',  
'update']
```

Functions

- 2 types
 - builtin functions
 - user defined functions

```
In [93]: import builtins
```

```
In [94]: dir(builtins)
```

```
Out[94]: ['ArithmeticError',  
          'AssertionError',  
          'AttributeError',  
          'BaseException',  
          'BlockingIOError',  
          'BrokenPipeError',  
          'BufferError',  
          'BytesWarning',  
          'ChildProcessError',  
          'ConnectionAbortedError',  
          'ConnectionError',  
          'ConnectionRefusedError',  
          'ConnectionResetError',  
          'DeprecationWarning',  
          'EOFError',  
          'Ellipsis',  
          'EnvironmentError',  
          'Exception',  
          'False',  
          'FileExistsError',  
          'FileNotFoundError',  
          'FloatingPointError',  
          'FutureWarning',  
          'GeneratorExit',  
          'IOError',  
          'ImportError',  
          'ImportWarning',  
          'IndentationError',  
          'IndexError',  
          'InterruptedError',  
          'IsADirectoryError',  
          'KeyError',  
          'KeyboardInterrupt',  
          'LookupError',  
          'MemoryError',  
          'ModuleNotFoundError',  
          'NameError',  
          'None',  
          'NotADirectoryError',  
          'NotImplemented',  
          'NotImplementedError',  
          'OSError',  
          'OverflowError',  
          'PendingDeprecationWarning',  
          'PermissionError',  
          'ProcessLookupError',  
          'RecursionError',  
          'ReferenceError',  
          'ResourceWarning',  
          'RuntimeError',  
          'RuntimeWarning',  
          'StopAsyncIteration',  
          'StopIteration',  
          'SyntaxError',
```



```
'SyntaxWarning',
'SystemError',
'SystemExit',
'TabError',
'TimeoutError',
'True',
'TypeError',
'UnboundLocalError',
'UnicodeDecodeError',
'UnicodeEncodeError',
'UnicodeError',
'UnicodeTranslateError',
'UnicodeWarning',
'UserWarning',
'ValueError',
'Warning',
'WindowsError',
'ZeroDivisionError',
'__IPYTHON__',
'__build_class__',
'__debug__',
'__doc__',
'__import__',
'__loader__',
'__name__',
'__package__',
'__spec__',
'abs',
'all',
'any',
'ascii',
'bin',
'bool',
'breakpoint',
'bytearray',
'bytes',
'callable',
'chr',
'classmethod',
'compile',
'complex',
'copyright',
'credits',
'delattr',
'dict',
'dir',
'display',
'divmod',
'enumerate',
'eval',
'exec',
'filter',
'float',
'format',
'frozenset',
'get_ipython',
'getattr',
```

```
'globals',  
'hasattr',  
'hash',  
'help',  
'hex',  
'id',  
'input',  
'int',  
'isinstance',  
'issubclass',  
'iter',  
'len',  
'license',  
'list',  
'locals',  
'map',  
'max',  
'memoryview',  
'min',  
'next',  
'object',  
'oct',  
'open',  
'ord',  
'pow',  
'print',  
'property',  
'range',  
'repr',  
'reversed',  
'round',  
'set',  
'setattr',  
'slice',  
'sorted',  
'staticmethod',  
'str',  
'sum',  
'super',  
'tuple',  
'type',  
'vars',  
'zip']
```

In [96]: `help(sum)`

Help on built-in function sum in module builtins:

```
sum(iterable, start=0, /)
```

Return the sum of a 'start' value (default: 0) plus an iterable of numbers

When the iterable is empty, return the start value.

This function is intended specifically for use with numeric values and may reject non-numeric types.

```
In [97]: a = 10  
        b = 20  
        sum((a,b))
```

Out[97]: 30

```
In [98]: sum((1,2))
```

Out[98]: 3

```
In [99]: ord('a')
```

Out[99]: 97

```
In [100]: chr(97)
```

Out[100]: 'a'

```
In [106]: def add(a,b):  
          c = a+b  
          return c
```

```
In [107]: add(10,20)
```

Out[107]: 30

```
In [108]: def add():  
          c = 7+9  
          return c  
add()
```

Out[108]: 16

```
In [110]: # positional arguments -- pass in an order  
  
def student(marks,name,rollnumber):  
    return(marks,name,rollnumber)  
student(56,"alekhya","13p7654")
```

Out[110]: (56, 'alekhya', '13p7654')

```
In [111]: # keyword arguments
#key = value

def student(marks,name):
    return(marks,name)
student(marks=89,name="apssdc")
```

Out[111]: (89, 'apssdc')

```
In [112]: # default arguments

def default(marks,name="apssdc"):
    return(marks,name)
default(78)
```

Out[112]: (78, 'apssdc')

```
In [113]: default(89,"alekhya")
```

Out[113]: (89, 'alekhya')

```
In [116]: # variable length argument
def variable_length(*x):
    return x
variable_length(1,2,3,4.8,5,"alekhya")
```

Out[116]: (1, 2, 3, 4.8, 5, 'alekhya')

```
In [121]: # keyword length arguments
def person(**kwlen):
    return kwlen
person(name="alekhya",marks=89,mailid = "alekhya.g@apssdc.in")
```

Out[121]: {'name': 'alekhya', 'marks': 89, 'mailid': 'alekhya.g@apssdc.in'}

Modules and Packages

- module is a single python file created for specific task
- package is a bunch of modules available on a single folder with specific task

```
In [122]: import programming

# moduleName.methodname
```

```
In [125]: programming.add(35,4)
```

Out[125]: 39

```
In [124]: programming.sub(5,4)
```

```
Out[124]: 1
```

```
In [127]: import pack.basic1
```

```
In [128]: from pack import basic1
```

```
In [129]: from pack import *
```

```
In [130]: from pack import basic1 as ba
```

```
In [131]: help(ba)
```

Help on module pack.basic1 in pack:

NAME

pack.basic1

FUNCTIONS

even(a)

FILE

c:\users\alekhya\desktop\machine learning(17-05-2021)\day3\pack\basic1.py

```
In [132]: ba.even(7)
```

```
Out[132]: False
```

Comprehensions

```
In [134]: # append 1-100 numbers in a list
l = []
for i in range(1,101):
    l.append(i)

print(l)
```

```
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22,
23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42,
43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62,
63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82,
83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100]
```

List Comprehension

- syntax:

- `newlist = [expression for iter in iterable if condition==TRUE]`

In [135]: `l = [i for i in range(1,101)]`

In [137]: `print(l)`

```
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22,
23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42,
43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62,
63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82,
83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100]
```

dictionary Comprehension

- syntax:
 - `newdict = {key:value for iter in iterable if condition==TRUE}`

In [139]: `d = {i:i**2 for i in range(1,101)}`
`print(d)`

```
{1: 1, 2: 4, 3: 9, 4: 16, 5: 25, 6: 36, 7: 49, 8: 64, 9: 81, 10: 100, 11: 121,
12: 144, 13: 169, 14: 196, 15: 225, 16: 256, 17: 289, 18: 324, 19: 361, 20: 400,
21: 441, 22: 484, 23: 529, 24: 576, 25: 625, 26: 676, 27: 729, 28: 784, 29: 841,
30: 900, 31: 961, 32: 1024, 33: 1089, 34: 1156, 35: 1225, 36: 1296, 37: 1369,
38: 1444, 39: 1521, 40: 1600, 41: 1681, 42: 1764, 43: 1849, 44: 1936, 45: 2025,
46: 2116, 47: 2209, 48: 2304, 49: 2401, 50: 2500, 51: 2601, 52: 2704, 53: 2809,
54: 2916, 55: 3025, 56: 3136, 57: 3249, 58: 3364, 59: 3481, 60: 3600, 61: 3721,
62: 3844, 63: 3969, 64: 4096, 65: 4225, 66: 4356, 67: 4489, 68: 4624, 69: 4761,
70: 4900, 71: 5041, 72: 5184, 73: 5329, 74: 5476, 75: 5625, 76: 5776, 77: 5929,
78: 6084, 79: 6241, 80: 6400, 81: 6561, 82: 6724, 83: 6889, 84: 7056, 85: 7225,
86: 7396, 87: 7569, 88: 7744, 89: 7921, 90: 8100, 91: 8281, 92: 8464, 93: 8649,
94: 8836, 95: 9025, 96: 9216, 97: 9409, 98: 9604, 99: 9801, 100: 10000}
```

In []: