

## Build-in Functions #2

- **filter()** - filter objects from any iterable based on function you give

```
>>>
>>> L = [3, 6, 7, 9, 12, 14, 19, 21]
>>>
>>> def even(x):
        if x % 2 == 0:
            return True
        else:
            return False

>>> filter(even, L)
<filter object at 0x7f8ce252e8e0>
>>> f = filter(even, L)
>>>
>>> for i in f:
        print(i)

6
12
14
```

- **float()** - Convert datatype into float type
- **format()** - same as string formatting
- **frozenset()** - Takes any literal and convert it into immutable set
- **globals()** - gives all the global variables declared inside a python program

```
>>> f = 12.54634
>>> format(f, 'E')
'1.254634E+01'
>>>
>>> L = [1,2,3,4,5]
>>> fz = frozenset(L)
>>>
>>> fz
frozenset({1, 2, 3, 4, 5})
>>> globals()
{'__name__': '__main__', '__doc__': None, '__package__': None, '__loader__': <class '_frozen_importlib.BuiltinImporter'>, '__spec__': None, '__annotations__': {}, '__builtins__': <module 'builtins' (built-in)>, 'f': 12.54634, 'L': [1, 2, 3, 4, 5], 'fz': frozenset({1, 2, 3, 4, 5})}
>>>
>>> locals()
{'__name__': '__main__', '__doc__': None, '__package__': None, '__loader__': <class '_frozen_importlib.BuiltinImporter'>, '__spec__': None, '__annotations__': {}, '__builtins__': <module 'builtins' (built-in)>, 'f': 12.54634, 'L': [1, 2, 3, 4, 5], 'fz': frozenset({1, 2, 3, 4, 5})}
>>>
```

- `hasattr()` - verifying if you are having so and so attribute or not
- `hash()` - it'll show the hash value of the particular object you called
- `help()` - gives details about any method or class
- `hex()` - base conversion

```
>>>
>>> s1 = 'abcde'
>>>
>>> hasattr(s1, 'lower')
True
>>> hasattr(s1, 'isdigit')
True
>>> hasattr(s1, 'total')
SyntaxError: EOL while scanning string literal
>>> hasattr(s1, 'total')
False
>>> hash(s1)
-2359171732725835016
>>> n = 10
>>> hash(n)
10
>>> f = 12.345
>>> hash(f)
795515838178725900
>>>
```

```
>>>
>>> s1 = 'abcde'
>>>
>>> help(s1.lower)
Help on built-in function lower:

lower() method of builtins.str instance
    Return a copy of the string converted to lowercase.

>>> help(s1.isdigit)
Help on built-in function isdigit:

isdigit() method of builtins.str instance
    Return True if the string is a digit string, False otherwise.

    A string is a digit string if all characters in the string are digits and there
    is at least one character in the string.
```

- `isinstance()` - check if an object is an instance of particular class
- `issubclass()` - tells if a class is a subclass or not
- `iter()` - helps us iterate over every element in a sequence used along with `next()`

```
>>>
>>> s1 = 'abcde'
>>> n = 10
>>> f = 12.34
>>>
>>> isinstance(s1, str)
True
>>> isinstance(n, int)
True
>>> isinstance(n, float)
False
>>> False
False
>>>
>>> L = [10, 'john', 15.76, 'James']
>>> itr = iter(L)
>>> next(itr)
10
>>> next(itr)
'john'
>>> next(itr)
15.76
>>> next(itr)
'James'
>>>
```