

# Python\_m\_Special(Magic\_or\_Dunder)\_[Day\_19](DATA MINDS)

September 18, 2023

MAGIC - : ANY OPERATION THAT WE DONE INSIDE THE PYTHON LIKE ADDITION,SUBSTRACTION,CONCATINATION ETC LIKE ANY KIND OF OPERATION SO BY DEFAULT IT CALLS A FUNCTION INTERNALLY THAT ALREADY MADE BY SOMEONE ELSE AND THAT'S CALLED MAGIC METHOD OR FUNCTION

IN VERY SIMPLE TERMS - : WHENEVER WE USE AN OPERATION IN PYTHON LIKE STRING OR INTEGER CONCATINATION OR ANY KIND OF ADDITION , SUBSTRACTION OPERATION SOMETHING LIKE THAT THAN BY DEFAULT INTERNALLY IT CALLS A FUNCTION THAT CREATED BY SOMEONE ELSE SO THIS IS CALLED AS "MAGIC"

DUNDER -: DUNDER IS NOTHING BUT A UDERSCFORE,UNDERSORE BEFORE A FUNCTION OR AFTER A FUNCTION LIKE AS WE USE IN BEFORE AND AFTER OF "INIT" FUNCTION SO OBOIOUSLY INIT IS ALSO A DUNDER

IMPORTANT - : DIR ( ) FUNCTION - WHENEVER WE WANT TO SEE THE NO OF FUNCTIONS WE SIMPLY PUT THE LIST , TUPLE , STR ,INT INSIDE THE DIR( ) FUNCTIONS THAT SHOWS ALL THE FUNCTIONS RELATED TO THAT PARTICULAR DATATYPE

```
[1]: #"dir" stands for Directory function

dir(int)

#these all functions are used whenever we call the int function inside the dir
```

```
[1]: ['__abs__',
      '__add__',
      '__and__',
      '__bool__',
      '__ceil__',
      '__class__',
      '__delattr__',
      '__dir__',
      '__divmod__',
      '__doc__',
      '__eq__',
      '__float__',
      '__floor__',
      '__floordiv__',
```

```
'__format__',
'__ge__',
'__getattr__',
'__getnewargs__',
'__gt__',
'__hash__',
'__index__',
'__init__',
'__init_subclass__',
'__int__',
'__invert__',
'__le__',
'__lshift__',
'__lt__',
'__mod__',
'__mul__',
'__ne__',
'__neg__',
'__new__',
'__or__',
'__pos__',
'__pow__',
'__radd__',
'__rand__',
'__rdivmod__',
'__reduce__',
'__reduce_ex__',
'__repr__',
'__rfloordiv__',
'__rlshift__',
'__rmod__',
'__rmul__',
'__ror__',
'__round__',
'__rpow__',
'__rrshift__',
'__rshift__',
'__rsub__',
'__rtruediv__',
'__rxor__',
'__setattr__',
'__sizeof__',
'__str__',
'__sub__',
'__subclasshook__',
'__truediv__',
'__trunc__',
```

```

'__xor__',
'as_integer_ratio',
'bit_count',
'bit_length',
'conjugate',
'denominator',
'from_bytes',
'imag',
'numerator',
'real',
'to_bytes']

```

[2]: *#whenever we use str function so all these things are going to be done inside*  
*↪ the function*

```
dir(str)
```

[2]:

```

['__add__',
'__class__',
'__contains__',
'__delattr__',
'__dir__',
'__doc__',
'__eq__',
'__format__',
'__ge__',
'__getattr__',
'__getitem__',
'__getnewargs__',
'__gt__',
'__hash__',
'__init__',
'__init_subclass__',
'__iter__',
'__le__',
'__len__',
'__lt__',
'__mod__',
'__mul__',
'__ne__',
'__new__',
'__reduce__',
'__reduce_ex__',
'__repr__',
'__rmod__',
'__rmul__',
'__setattr__',

```

'\_\_sizeof\_\_',  
'\_\_str\_\_',  
'\_\_subclasshook\_\_',  
'capitalize',  
'casefold',  
'center',  
'count',  
'encode',  
'endswith',  
'expandtabs',  
'find',  
'format',  
'format\_map',  
'index',  
'isalnum',  
'isalpha',  
'isascii',  
'isdecimal',  
'isdigit',  
'isidentifier',  
'islower',  
'isnumeric',  
'isprintable',  
'isspace',  
'istitle',  
'isupper',  
'join',  
'ljust',  
'lower',  
'lstrip',  
'maketrans',  
'partition',  
'removeprefix',  
'removesuffix',  
'replace',  
'rfind',  
'rindex',  
'rjust',  
'rpartition',  
'rsplit',  
'rstrip',  
'split',  
'splitlines',  
'startswith',  
'strip',  
'swapcase',  
'title',

```
'translate',  
'upper',  
'zfill']
```

NOTE - BASICALLY THE PREVIOUS CODE WE WRITE INSIDE THE SHELLS ARE CALLED MAGIC FUNCTION OR DUNDER FUNCTION

1 ) We take example of addition

```
[3]: #a = variable
```

```
a=100
```

```
[4]: #we simply add 5 in variable "a"
```

```
a+5
```

```
[4]: 105
```

```
[6]: #we done same thing here , we add 5 in variable "a" but the differnce is that  
      ↳ here we achieve the same thing with the help of "MAGIC" and the double  
      ↳ underscore that we have use before or after the the add that's called  
      ↳ "DUNDER"
```

```
a.__add__(5)
```

```
[6]: 105
```

2 ) We take example of substraction

```
[7]: #b=variable
```

```
b=50
```

```
[8]: #simple way of substraction
```

```
b-15
```

```
[8]: 35
```

```
[10]: #This way of writing code is called the "MAGIC" that same as perevious code but  
      ↳ the difference is only the syntax , we use double underscore thatt's called  
      ↳ Dunder and writing this ways is called "magic"
```

```
b.__sub__(15)
```

```
[10]: 35
```

3 ) We take example of multiplication

```
[11]: c=8
```

```
[12]: c*5
```

```
[12]: 40
```

```
[13]: #By using Magic or Dunder  
c.__mul__(5)
```

```
[13]: 40
```

```
[25]: class Data_science:  
      def __init__(self):  
          self.mob_no=846792462046
```

```
[26]: Ds=Data_science()
```

```
[27]: Ds.mob_no
```

```
[27]: 846792462046
```

```
[41]: class Intro:  
      def __init__(self):  
          self.name="Virat Tiwari"
```

```
[42]: Detail=Intro()
```

```
[43]: Detail.name
```

```
[43]: 'Virat Tiwari'
```

```
[49]: class Intro:  
  
      def __new__(cls):  
          print("This is my NEW MAGIC")  
  
      def __init__(self):  
          print("This is my OLD INIT")  
  
          self.name="Virat Tiwari"
```

```
#NOTE - IN THIS FUNCTION WE CREATE MAGIC FUNCTION WITH THE NAME OF __new__ SO  
↳ WE SEE THAT MAGIC FUNCTION ALWAYS EXECUTE BEFORE THE THE INIT FUNCTION  
↳ WHENEVER WE CALL THESE TWO FUNCTIONS TOGETHER  
#BUT IN GENERAL WE NEVER DIRECTLY CALL THEM(MAGIC FUNCTION)
```

```
[50]: Detail=Intro()
```

This is my NEW MAGIC

IMPORTANT NOTE -: IN PYTHON , OBJECT OR INSTANCE IS GENERALLY CREATED BY THE MAGIC FUNCTION BUT WE THINK IT CREATED BY THE INIT FUNCTION , IN VERY SIMPLE TERMS MAGIC FUNCTION CREATED THE OBJECT OF CLASS BEFORE THE INIT FUNCTION IN PYTHON PROGRAMMING LANGUAGE

```
[54]: class Detail:  
  
    def __init__(self):  
  
        self.name="Virat Tiwari"
```

```
[57]: Stu=Detail()
```

```
[58]: Stu.name
```

```
[58]: 'Virat Tiwari'
```

```
[59]: type(Stu)
```

```
[59]: __main__.Detail
```

```
[62]: Stu
```

```
[62]: <__main__.Detail at 0x7fd1b3490460>
```

```
[63]: # ("0x7fd1b3490460") - This is called hexadecimal code
```

NOTE - : WITH THE HELP OF MAGIC WE CAN WRITE SOMETHING FOR OUR OBJECT AND THERE IS SOME OTHER NAMES OR SYNONYMS OF OBJECTS AS WELL LIKE INSTANCE OR VARIABLE