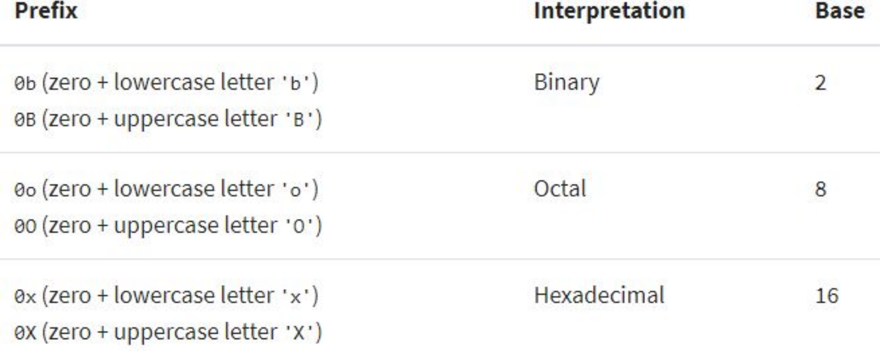
Python

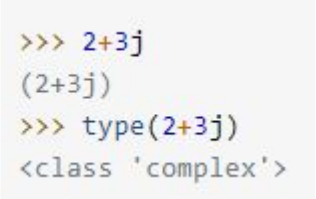
Code Girls 2020-21

Basic Data Types

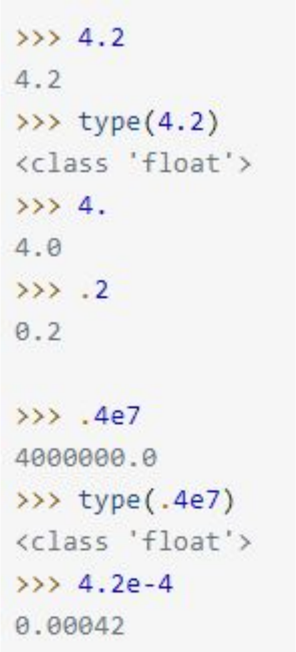
- int
  - You can also do other bases!
- float (values with a decimal point)
  - 4.2, 4.0, 0.2,
- complex
- str
- bool
  - True or False



Prefix	Interpretation	Base
0b (zero + lowercase letter 'b') 0B (zero + uppercase letter 'B')	Binary	2
0o (zero + lowercase letter 'o') 0O (zero + uppercase letter 'O')	Octal	8
0x (zero + lowercase letter 'x') 0X (zero + uppercase letter 'X')	Hexadecimal	16



```
>>> 2+3j
(2+3j)
>>> type(2+3j)
<class 'complex'>
```

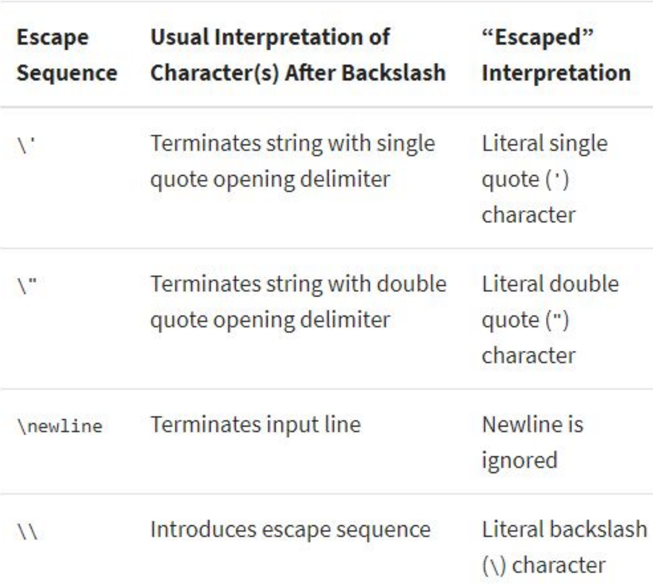


```
>>> 4.2
4.2
>>> type(4.2)
<class 'float'>
>>> 4.
4.0
>>> .2
0.2

>>> .4e7
4000000.0
>>> type(.4e7)
<class 'float'>
>>> 4.2e-4
0.00042
```

str

- Single or double quotes
    - If you want either type of quote within use the other type on the outside
- Escape sequences
    - Use a backslash (\)
    - Suppress the special interpretation that certain characters are usually given within a string
    - Apply special interpretation to characters in a string which would normally be taken literally
        - Commonly used: \n, \t

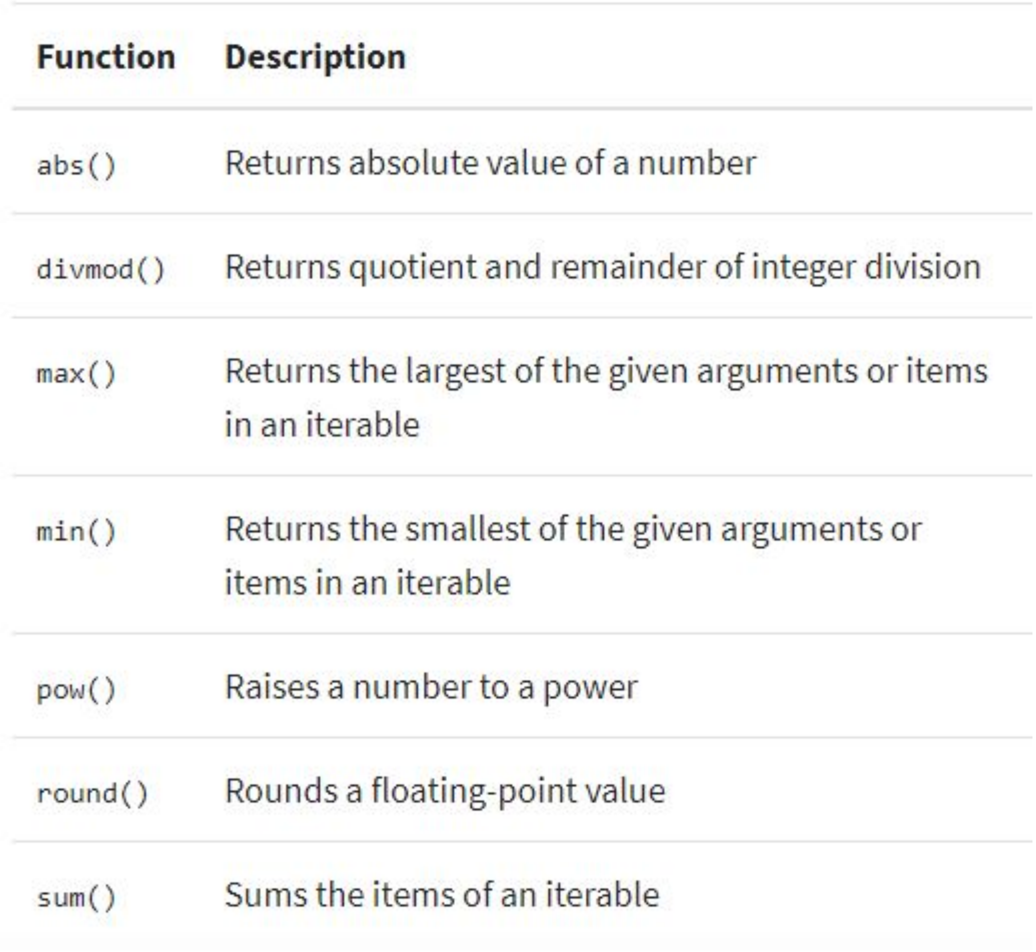


Escape Sequence	Usual Interpretation of Character(s) After Backslash	“Escaped” Interpretation
`\'`	Terminates string with single quote opening delimiter	Literal single quote (') character
`\"`	Terminates string with double quote opening delimiter	Literal double quote (") character
`\newline`	Terminates input line	Newline is ignored
`\\`	Introduces escape sequence	Literal backslash (\) character

str

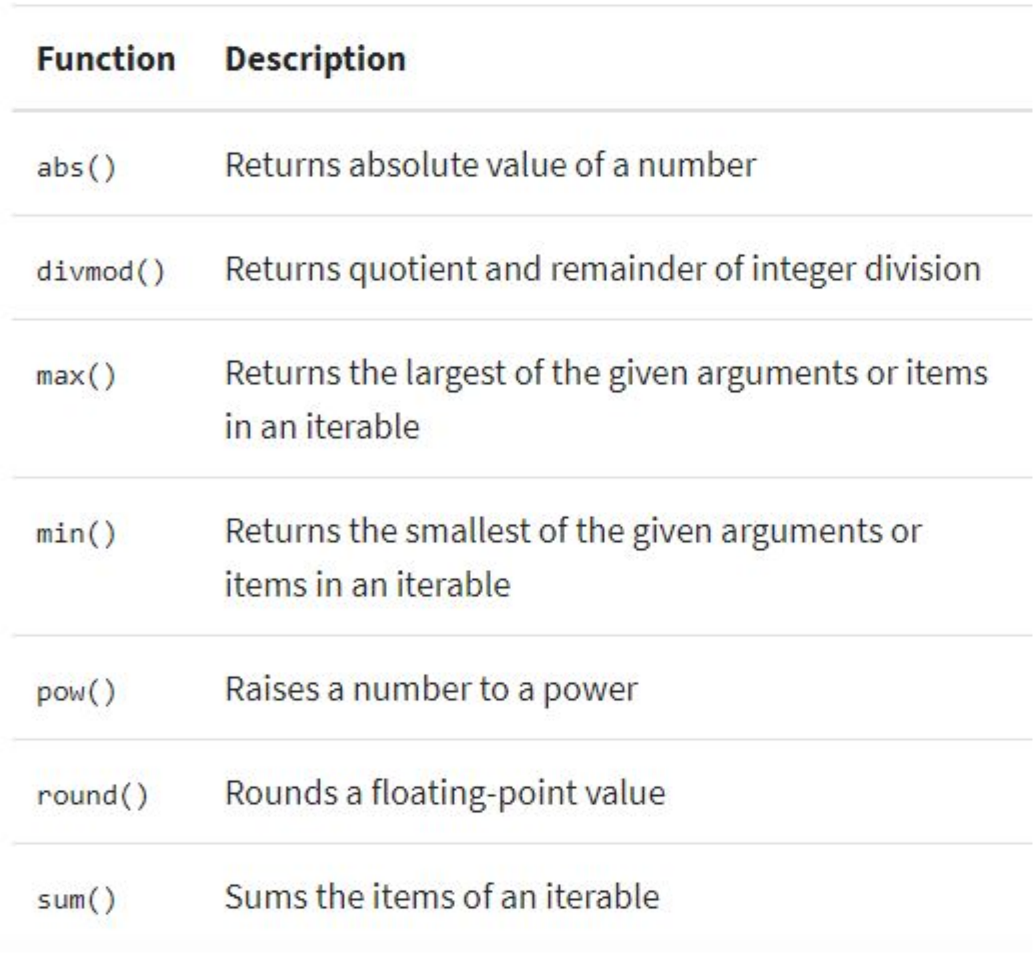
- Raw Strings
  - Preceded by r or R
    - print(r'foo\nbar') or print(R'foo\\bar')
  - Backslashes aren’t translated and are left in the string
- Triple-Quoted Strings
  - Single quotes, double quotes, and newlines can be included without escaping them

Math!



Function	Description
`abs()`	Returns absolute value of a number
`divmod()`	Returns quotient and remainder of integer division
`max()`	Returns the largest of the given arguments or items in an iterable
`min()`	Returns the smallest of the given arguments or items in an iterable
`pow()`	Raises a number to a power
`round()`	Rounds a floating-point value
`sum()`	Sums the items of an iterable

Math!

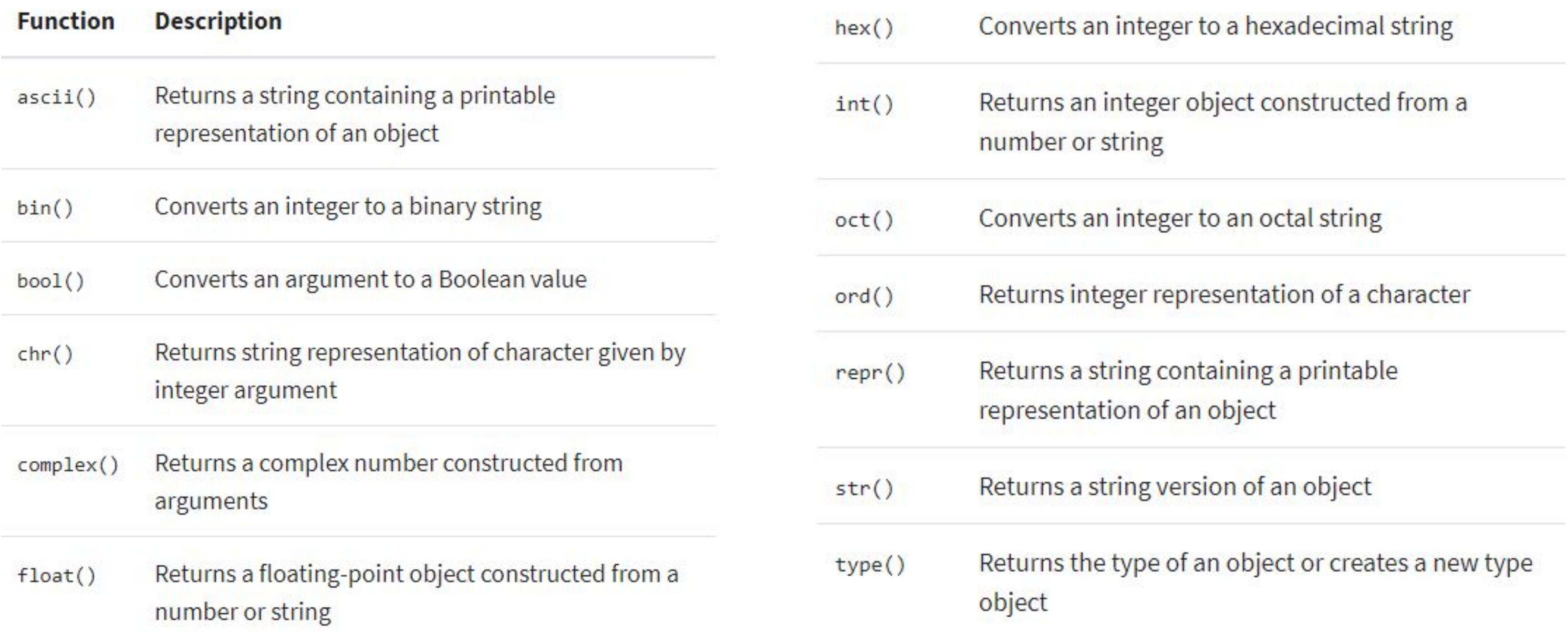


Function	Description
`abs()`	Returns absolute value of a number
`divmod()`	Returns quotient and remainder of integer division
`max()`	Returns the largest of the given arguments or items in an iterable
`min()`	Returns the smallest of the given arguments or items in an iterable
`pow()`	Raises a number to a power
`round()`	Rounds a floating-point value
`sum()`	Sums the items of an iterable

More math!

- import math
  - math.ceil(x)
  - math.factorial(x)
  - math.gcd(integers)
  - math.remainder(x, y)
  - math.trunc(x)
  - And more! https://docs.python.org/3/library/math.html

Type Conversion



Function	Description
`ascii()`	Returns a string containing a printable representation of an object
`bin()`	Converts an integer to a binary string
`bool()`	Converts an argument to a Boolean value
`chr()`	Returns string representation of character given by integer argument
`complex()`	Returns a complex number constructed from arguments
`float()`	Returns a floating-point object constructed from a number or string
`hex()`	Converts an integer to a hexadecimal string
`int()`	Returns an integer object constructed from a number or string
`oct()`	Converts an integer to an octal string
`ord()`	Returns integer representation of a character
`repr()`	Returns a string containing a printable representation of an object
`str()`	Returns a string version of an object
`type()`	Returns the type of an object or creates a new type object

Practice!

- Open Python! You can use repl.it if you don’t have it downloaded on your computer
- Print something with quotes using as many methods as you can think of
- Print something with a tab
- Print this using one print statement

  ```
  ****
   ***
    **
     *
  ```

- Try one of the math functions

Acknowledgments

Thanks to:

- Ms. Lola Piper for sponsoring our club and guidance