

# Representation

---

# Announcements

# String Representations

# String Representations

---

In Python, all objects produce two string representations:

- The `str` is (often) legible to **humans** & shows up when you **print**
- The `repr` is (often) legible to **Python** & shows up when you **evaluate** interactively

The `str` and `repr` strings are often the same, but not always

```
>>> from fractions import Fraction
>>> half = Fraction(1, 2)
>>> str(half)
'1/2'
>>> repr(half)
'Fraction(1, 2)'
>>> print(half)
1/2
>>> half
Fraction(1, 2)
```



<https://tinyurl.com/61asp26>

If a type only defines a `repr` string, then the `repr` string is also the `str` string.

(Demo)

# Special Method Names in Python

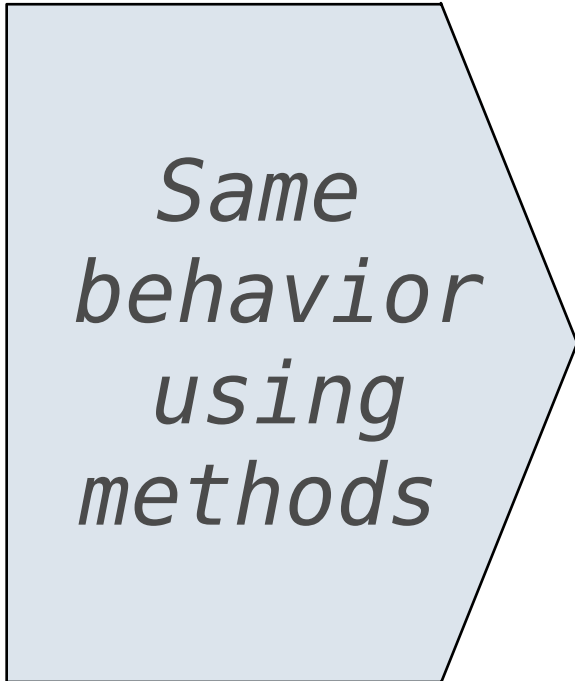
---

Certain names are special because they have built-in behavior

These names always start and end with two underscores

<code>__init__</code>	Method invoked automatically when an object is constructed
<code>__str__</code>	Method invoked by <code>str()</code> and <code>print()</code>
<code>__repr__</code>	Method invoked to display an object as a Python expression
<code>__eq__</code>	Method invoked by <code>==</code> , to compare two objects
<code>__bool__</code>	Method invoked to convert an object to True or False

```
>>> t0 = Transaction(0, 20, 5)
>>> t1 = Transaction(1, 5, 5)
>>> str(t1)
'1: no change'
>>> t0 == t1
False
>>> bool(t0)
True
```



Same  
behavior  
using  
methods

```
>>> t1.__str__()
'1: no change'
>>> t0.__eq__(t1)
False
>>> t0.__bool__()
True
```

## (Modified) Spring 2023 Midterm 2 Question 2(a)

---

```
class Letter:
    def __init__(self, contents: str):
        self.contents = contents
        self.sent = False

    def send(self):
        if self.sent:
            print(self, 'was already sent.')
        else:
            print(self, 'has been sent.')
            self.sent = True
            return Letter(self.contents.upper())

    def __repr__(self):
        return f'Letter({repr(self.contents)})'
```

Implement the **Letter** class. A **Letter** has two instance attributes: **contents** (a **str**) and **sent** (a **bool**). Each **Letter** can only be sent once. The **send** method prints whether the letter was sent, and if it was, returns the reply, which is a new **Letter** instance with the same contents, but in all caps.

*Hint:* 'hi'.upper() evaluates to 'HI'.

```
"""A letter receives an all-caps reply.

>>> hi = Letter('Hello, World!')
>>> hi.send()
Letter('Hello, World!') has been sent.
Letter('HELLO, WORLD!')
>>> hi.send()
Letter('Hello, World!') was already sent.
>>> Letter('Hey').send().send()
Letter('Hey') has been sent.
Letter('HEY') has been sent.
Letter('HEY')
"""
```