lecture_12

September 27, 2022

1 Lecture 12

[2]:
```
class Foo:
    def __init__(self, name):
        self.name = name
```

[3]:
```
a = Foo("Adam")
```

[4]:
```
a.name
```

[4]:
```
'Adam'
```

[5]:
```
a.name.lower()
```

[5]:
```
'adam'
```

[6]:
```
class Employee:
    def __init__(self, name, salary, position):
        self.name = name
        self.salary = salary
        self.position = position
```

[7]:
```
employee_1 = Employee(name="Adam", salary=100_000, position="Junior Engineer")
```

[24]:
```
# "Mastrots 1, Yerevan 002, Armenia"
class Address:
    def __init__(
        self,
        street_name,
        building_number,
        city,
        postal_index,
        country,
        apartment_number=None,
    ):
        self.street_name = street_name
        self.building_number = building_number
```

```
        self.apartment_number = apartment_number
        self.city = city
        self.postal_index = postal_index
        self.country = country

    def __str__(self):
        return self.full_address

    @property
    def full_address(self):
        return f"{self.street_name} {self.building_number}{f', apt. {self.
 ↪apartment_number}' if self.apartment_number else ''}, {self.city} {self.
 ↪postal_index}, {self.country}"

    def __repr__(self):
        return self.full_address
```

[25]:
```
address_1 = Address(
    street_name="Mashtots",
    building_number=1,
    city="Yerevan",
    postal_index="0002",
    country="Armenia",
)
```

[26]:
```
print(address_1)
```

```
Mashtots 1, Yerevan 0002, Armenia
```

[27]:
```
address_1.full_address
```

[27]:
```
'Mashtots 1, Yerevan 0002, Armenia'
```

[28]:
```
address_1.city
```

[28]:
```
'Yerevan'
```

[29]:
```
class Employee:
    def __init__(self, name, salary, position, address):
        self.name = name
        self.salary = salary
        self.position = position
        self.address = address
```

[30]:
```
employee_1 = Employee(name="Adam", salary=100_000, position="Junior Engineer",␣
 ↪address=address_1)
```

[31]:
```
employee_1.address.full_address
```

[31]:
```
'Mashtots 1, Yerevan 0002, Armenia'
```

[32]:
```
employee_1.address.city
```

[32]:
```
'Yerevan'
```

[33]:
```
employee_1.address
```

[33]:
```
Mashtots 1, Yerevan 0002, Armenia
```

[38]:
```
employee_1 = Employee(
    name="Adam",
    salary=100_000,
    position="Junior Engineer",
    address=Address(
        street_name="Mashtots",
        building_number=1,
        city="Yerevan",
        postal_index="0002",
        country="Armenia",
    ),
)
```

[39]:
```
employee_1.address.city = "Ashtarak"
```

[40]:
```
employee_1.address.full_address
```

[40]:
```
'Mashtots 1, Ashtarak 0002, Armenia'
```

[41]:
```
employee_1.address.city
```

[41]:
```
'Ashtarak'
```

[43]:
```
employee_1 = Employee(
    name="Adam",
    salary=100_000,
    position="Junior Engineer",
    address=Address(
        street_name="Mashtots",
        building_number=1,
        city="Yerevan",
        postal_index="0002",
        country="Armenia",
    ),
)
```

```
employee_2 = Employee(
    name="Yeva",
    salary=120_000,
    position="Junior Engineer",
    address=Address(
        street_name="Mashtots",
        building_number=1,
        city="Yerevan",
        postal_index="0002",
        country="Armenia",
    ),
)
```

[44]:
```
employee_1.address.full_address
```

[44]: 'Mashtots 1, Yerevan 0002, Armenia'

[45]:
```
employee_2.address.full_address
```

[45]: 'Mashtots 1, Yerevan 0002, Armenia'

[46]:
```
employee_1.address.city = "Ashtarak"
```

[47]:
```
employee_1.address.full_address
```

[47]: 'Mashtots 1, Ashtarak 0002, Armenia'

[48]:
```
employee_2.address.full_address
```

[48]: 'Mashtots 1, Yerevan 0002, Armenia'

[49]:
```
address = Address(
    street_name="Mashtots",
    building_number=1,
    city="Yerevan",
    postal_index="0002",
    country="Armenia",
)

employee_1 = Employee(
    name="Adam",
    salary=100_000,
    position="Junior Engineer",
    address=address,
)
```

```
employee_2 = Employee(
    name="Yeva",
    salary=120_000,
    position="Junior Engineer",
    address=address,
)
```

[50]:
```
employee_1.address.full_address
```

[50]: 'Mashtots 1, Yerevan 0002, Armenia'

[51]:
```
employee_2.address.full_address
```

[51]: 'Mashtots 1, Yerevan 0002, Armenia'

[54]:
```
employee_1.address.city = "Ashtarak"
```

[55]:
```
employee_1.address.full_address
```

[55]: 'Mashtots 1, Ashtarak 0002, Armenia'

[56]:
```
employee_2.address.full_address
```

[56]: 'Mashtots 1, Ashtarak 0002, Armenia'

[58]:
```
class School:
    def __init__(self, name, number, address):
        self.name = name
        self.address = address
        self.number = number
```

[59]:
```
school_1 = School(
    number=19,
    name="Kurupski",
    address=Address(
        street_name="Mashtots",
        building_number=1,
        city="Yerevan",
        postal_index="0002",
        country="Armenia",
    )
)
```

[60]:
```
class Child:
    def __init__(self, name, date_of_birth, school):
        self.name = name
        self.date_of_birth = date_of_birth
```

```
        self.school = school
```

[61]:
```
child_1 = Child("Jack", date_of_birth="2001-01-01", school=school_1)
```

[62]:
```
child_2 = Child("John", date_of_birth="2003-03-03", school=school_1)
```

[63]:
```
child_1.school.name
```

[63]:
```
'Kurupski'
```

[64]:
```
child_2.school.name
```

[64]:
```
'Kurupski'
```

[65]:
```
school_1.name = "Nikol Aghbalyab"
```

[66]:
```
child_1.school.name
```

[66]:
```
'Nikol Aghbalyab'
```

[67]:
```
child_2.school.name
```

[67]:
```
'Nikol Aghbalyab'
```

[68]:
```
child_2.school = School(
    name="Pushkin",
    number=8,
    address=address_1,
)
```

[69]:
```
child_1.school.name
```

[69]:
```
'Nikol Aghbalyab'
```

[70]:
```
child_2.school.name
```

[70]:
```
'Pushkin'
```

[72]:
```
10 * 365 * 24 * 60 * 60 + 2 * 24 * 360
```

[72]:
```
315377280
```

[73]:
```
import datetime
```

[75]:
```
the_date = datetime.date(2002, 2, 21)
```

[76]:
```
the_date
```

[76]:
```
datetime.date(2002, 2, 21)
```

[77]:
```
type(the_date)
```

[77]:
```
datetime.date
```

[78]:
```
the_date.isoformat()
```

[78]:
```
'2002-02-21'
```

[79]:
```
the_date.strftime("%m-%d-%Y")
```

[79]:
```
'02-21-2002'
```

[80]:
```
the_date.strftime("%m-%d-%y")
```

[80]:
```
'02-21-02'
```

[82]:
```
the_date.strftime("%m/%d/%Y")
```

[82]:
```
'02/21/2002'
```

[88]:
```
the_date.strftime("  %a %B %d, %Y  ")
```

[88]:
```
'  Thu February 21, 2002  '
```

[91]:
```
datetime.datetime.strptime('  Thu February 21, 2002  ', "  %a %B %d, %Y  ")
```

[91]:
```
datetime.date(2002, 2, 21)
```

[98]:
```
new_date = datetime.datetime.strptime('  Thu February 21, 2002  ', "  %a %B %d,␣
↪%Y  ")
```

[99]:
```
new_date + datetime.timedelta(days=7, hours=7)
```

[99]:
```
datetime.datetime(2002, 2, 28, 7, 0)
```

[100]:
```
import calendar
```

[102]:
```
calendar.calendar(2022)
```

[102]:
```
'                                  2022\n\n      January
February                   March\nMo Tu We Th Fr Sa Su      Mo Tu We Th Fr Sa Su
Mo Tu We Th Fr Sa Su\n                1  2          1  2  3  4  5  6          1
2  3  4  5  6\n 3  4  5  6  7  8  9       7  8  9 10 11 12 13       7  8  9 10
11 12 13\n10 11 12 13 14 15 16      14 15 16 17 18 19 20      14 15 16 17 18 19
20\n17 18 19 20 21 22 23      21 22 23 24 25 26 27      21 22 23 24 25 26 27\n24
```

```
25 26 27 28 29 30      28                        28 29 30 31\n31\n\n       April
May                        June\nMo Tu We Th Fr Sa Su      Mo Tu We Th Fr Sa Su
Mo Tu We Th Fr Sa Su\n             1  2  3                          1
1  2  3  4  5\n 4  5  6  7  8  9 10       2  3  4  5  6  7  8       6  7  8  9
10 11 12\n11 12 13 14 15 16 17       9 10 11 12 13 14 15      13 14 15 16 17 18
19\n18 19 20 21 22 23 24      16 17 18 19 20 21 22      20 21 22 23 24 25 26\n25
26 27 28 29 30         23 24 25 26 27 28 29      27 28 29 30\n
30 31\n\n        July                      August                   September\nMo
Tu We Th Fr Sa Su      Mo Tu We Th Fr Sa Su      Mo Tu We Th Fr Sa Su\n
1  2  3       1  2  3  4  5  6  7               1  2  3  4\n 4  5  6  7  8  9
10       8  9 10 11 12 13 14       5  6  7  8  9 10 11\n11 12 13 14 15 16 17
15 16 17 18 19 20 21      12 13 14 15 16 17 18\n18 19 20 21 22 23 24      22 23
24 25 26 27 28      19 20 21 22 23 24 25\n25 26 27 28 29 30 31      29 30 31
26 27 28 29 30\n\n      October                   November
December\nMo Tu We Th Fr Sa Su      Mo Tu We Th Fr Sa Su      Mo Tu We Th Fr Sa
Su\n                1  2          1  2  3  4  5  6                1  2  3  4\n 3
4  5  6  7  8  9       7  8  9 10 11 12 13       5  6  7  8  9 10 11\n10 11 12
13 14 15 16      14 15 16 17 18 19 20      12 13 14 15 16 17 18\n17 18 19 20 21
22 23      21 22 23 24 25 26 27      19 20 21 22 23 24 25\n24 25 26 27 28 29 30
28 29 30                  26 27 28 29 30 31\n31\n'
```

1.1 Encapsulation

[1]:
```
class Test:
    foo = 42
    _bar = 24    # protected
    __baz = 12   # private
```

[104]:
```
a = Test()
```

[105]:
```
a.foo
```

[105]:
```
42
```

[106]:
```
a._bar
```

[106]:
```
24
```

[107]:
```
a.__baz
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Input In [107], in <cell line: 1>()
----> 1 a.__baz

AttributeError: 'Test' object has no attribute '__baz'
```

[108]:
```
a._Test__baz
```

[108]:
```
12
```

[123]:
```
class Test:
    foo = 42
    _bar = 24
    __baz = 12

    def test(self):
        print(self.foo)
        print(self._bar)
        print(self.__baz)

        self.__private()

    def __private(self):
        print("Super secret")
```

[124]:
```
b = Test()
```

[125]:
```
b.test()
```

```
42
24
12
Super secret
```

[126]:
```
b.__private()
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Input In [126], in <cell line: 1>()
----> 1 b.__private()

AttributeError: 'Test' object has no attribute '__private'
```

[116]:
```
class ChildTest(Test):
    def test(self):
        print(self.foo)
        print(self._bar)
        print(self.__baz)
```

[117]:
```
c = ChildTest()
```

[118]:
```
c.test()
```

```
42
24
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Input In [118], in <cell line: 1>()
----> 1 c.test()

Input In [116], in ChildTest.test(self)
      3 print(self.foo)
      4 print(self._bar)
----> 5 print(self.__baz)

AttributeError: 'ChildTest' object has no attribute '_ChildTest__baz'
```

[ ]: