HW1

July 25, 2020

1 Homework 1

1.1 Write a Jupyter Magic

[10]:
```
from IPython.core.magic import (register_line_magic,␣
↪register_cell_magic,register_line_cell_magic)
```

[11]:
```
@register_line_cell_magic
def wordcountlc(line, cell=None):
    if cell == None:
        return len(line.split())
    else:
        return len((line+cell).split())
```

[12]:
```
%wordcountlc Simple test as line count
```

[12]: 5

[17]:
```
%%wordcountlc

now let's see cell count
cell
```

[17]: 6

[20]:
```
%%wordcountlc

now let's see cell count
cell.. .
```

[20]: 7

1.2 Profile the speed of list comprehension vs. for loops

[28]:
```
%time

[x for x in range(10) if x%2==1]
```

```
CPU times: user 3 µs, sys: 0 ns, total: 3 µs
Wall time: 5.01 µs
```

[28]:
```
[1, 3, 5, 7, 9]
```

[29]:
```
%%time

[x for x in range(10) if x%2==1]
```

```
CPU times: user 6 µs, sys: 0 ns, total: 6 µs
Wall time: 9.06 µs
```

[29]:
```
[1, 3, 5, 7, 9]
```

[38]:
```
%time
x=[]
for i in range(10):
    if i%2==1:
        x.append(i)
x
```

```
CPU times: user 3 µs, sys: 0 ns, total: 3 µs
Wall time: 4.77 µs
```

[38]:
```
[1, 3, 5, 7, 9]
```

[37]:
```
%%time
x=[]
for i in range(10):
    if i%2==1:
        x.append(i)
x
```

```
CPU times: user 24 µs, sys: 0 ns, total: 24 µs
Wall time: 26.2 µs
```

[37]:
```
[1, 3, 5, 7, 9]
```

1.3 Prime numbers

[47]:
```
def prime_number(listin):
    return [x for x in listin if x>1 and 0 not in [x%y for y in␣
↪range(2,int(x**0.5)+1)]]
```

[48]:
```
prime_number(range(50))
```

[48]:
```
[2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47]
```

1.4 Extend Vector Class

[106]:
```
from math import hypot

class Vector:
    def __init__(self,*nums):
        self.v=nums

    def __getitem__(self,po):
        return self.v[po]

    def __len__(self):
        return len(self.v)


    def __pow__(self,power):
        return Vector(*(self[i]**power for i in range(len(self.v))))
```

[67]:
```
print(dir(1))
```

```
['__abs__', '__add__', '__and__', '__bool__', '__ceil__', '__class__',
'__delattr__', '__dir__', '__divmod__', '__doc__', '__eq__', '__float__',
'__floor__', '__floordiv__', '__format__', '__ge__', '__getattribute__',
'__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__', 'bit_length',
'conjugate', 'denominator', 'from_bytes', 'imag', 'numerator', 'real',
'to_bytes']
```

[108]:
```
v = Vector(1, 2, 3, 4, 5)
# get item
v[2]

#slicing
v[2:3]
#Vector(2)

# length
len(v)
#5

#power
print(list(v ** 2))
#Vector(1, 4, 9, 16, 25)
```

```
[1, 4, 9, 16, 25]
```

1.5 Case insensitive dictionary

[159]:
```
class CaseInsensitiveDict(dict):

    def __getitem__(self,key):
        for i in self.keys():
            if i.lower() == key.lower():
                return self.get(i)

    def __setitem__(self,key,char):
        for i in self.keys():
            if i.lower()== key.lower():
                self.update({i:char})
                return
        self.update({key:char})
```

```
␣
↪---------------------------------------------------------------------------

TypeError                                 Traceback (most recent call␣
↪last)

<ipython-input-159-2f41e015667f> in <module>
----> 1 class CaseInsensitiveDict(dict):
      2
      3     def __getitem__(self,key):
      4         for i in self.keys():
```

```
          5             if i.lower() == key.lower():


        TypeError: dict expected at most 1 arguments, got 3
```

[158]:
```
dic= CaseInsensitiveDict()
dic['Abc']=1

print(dic['aBc'])
#print(dic['aBC'])
#print(dic['DEf'])
#dic['deF']=2
```

```
␣
↪---------------------------------------------------------------------------

        AttributeError                            Traceback (most recent call␣
↪last)

        <ipython-input-158-f6e73cbec440> in <module>
          1 dic= CaseInsensitiveDict()
    ----> 2 dic['Abc']=1
          3
          4 print(dic['aBc'])
          5 #print(dic['aBC'])


        <ipython-input-157-f3b4ce404ec9> in __setitem__(self, key, char)
          7
          8     def __setitem__(self,key,char):
    ----> 9         for i in self.keys():
         10             if i.lower()== key.lower():
         11                 self.update({i:char})


        AttributeError: 'CaseInsensitiveDict' object has no attribute 'keys'
```

[ ]:
```

```