Name- Jalaj Gupta

Subject- letsupgrade python batch 7 Day 9 Assignment

Question 1

Write a Python Function for finding a given number is prime or not and do unit testing on it using PyLint and Unittest Library

```
get_ipython().system(' pip install pylint')
```

```
Requirement already satisfied: pylint in /usr/local/lib/python3.6/dist-packages (2.6
Requirement already satisfied: isort<6,>=4.2.5 in /usr/local/lib/python3.6/dist-pack
Requirement already satisfied: astroid<=2.5,>=2.4.0 in /usr/local/lib/python3.6/dist
Requirement already satisfied: toml>=0.7.1 in /usr/local/lib/python3.6/dist-packages
Requirement already satisfied: mccabe<0.7,>=0.6 in /usr/local/lib/python3.6/dist-pac
Requirement already satisfied: typed-ast<1.5,>=1.4.0; implementation_name == "cpytho
Requirement already satisfied: six~=1.12 in /usr/local/lib/python3.6/dist-packages (
Requirement already satisfied: lazy-object-proxy==1.4.* in /usr/local/lib/python3.6/
Requirement already satisfied: wrapt~=1.11 in /usr/local/lib/python3.6/dist-packages
```

```
get_ipython().system(' pip install unittest2')
```

```
Requirement already satisfied: unittest2 in /usr/local/lib/python3.6/dist-packages (
Requirement already satisfied: six>=1.4 in /usr/local/lib/python3.6/dist-packages (f
Requirement already satisfied: traceback2 in /usr/local/lib/python3.6/dist-packages
Requirement already satisfied: argparse in /usr/local/lib/python3.6/dist-packages (f
Requirement already satisfied: linecache2 in /usr/local/lib/python3.6/dist-packages
```

```
get_ipython().run_cell_magic('writefile', 'primefile.py', "'''\nThis file returns true if
```

```
Overwriting primefile.py
```

```
get_ipython().system(' pylint "primefile.py"')
```

```
************* Module primefile
primefile.py:18:0: C0304: Final newline missing (missing-final-newline)
primefile.py:13:8: W0120: Else clause on loop without a break statement (useless-els
primefile.py:5:0: R1710: Either all return statements in a function should return an

------------------------------------------------------------------
Your code has been rated at 6.67/10 (previous run: 6.67/10, +0.00)
```

```
import primefile
```

```
Enter the number - 1234
1234 is not prime
```

```
get_ipython().run_cell_magic('writefile', 'testprime.py', '\nimport primefile\nimport uni
```

```
Writing testprime.py
```

```
import testprime
```

```
get_ipython().system(' python testprime.py')
```

```
Enter the number - 17
17 is prime
.
----------------------------------------------------------------------
Ran 1 test in 0.000s

OK
```

Question 2

Make a generator program for returning armstrong numbers in between 1-1000 in a generator object

```
def armstrongnumber(num):
   for item in num:
       order=len(str(item))
       temp=item
       summ=0
       while temp>0:
           digit=temp%10
           summ+=digit**order
           temp=temp//10
       if item==summ:
           yield item
```

```
lst=list(range(1,1001))
```

```
print(list(armstrongnumber(lst)))
```

```
[1, 2, 3, 4, 5, 6, 7, 8, 9, 153, 370, 371, 407]
```