In Python 3, there is no xrange, but the range function behaves like xrange in Python 2.
range and xrange are built-in functions in Python that are used to generate a list of integers within a given range.

Python 2: It returns a list of integers.
>>> range(10)
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
	Python 3:
	>>> range(10)
	range(0, 10)
	
	>>> list(range(10))
	[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

Python 2: It returns a generator object. >>> xrange(10) xrange(10)	Python 3: >>> xrange(10) Traceback (most recent call last): File "<stdin>", line 1, in <module> NameError: name 'xrange' is not defined

Python 2: type of range
>>> r = range(10)
>>> type(r)
<type 'list'>
	Python 3:
	>>> r = range(10)
	>>> type(r)
	<class 'range'>

Python 2: type of xrange
>>> x = xrange(10)
>>> print(type(x))
<type 'xrange'>
	Python 3:
	>>> x = xrange(10)
	Traceback (most recent call last):
	File "<stdin>", line 1, in <module>
	NameError: name 'xrange' is not defined

Python 2:

range: execution speed is slower

```
>>> import timeit
>>> r = 'range(10)'
>>> print(timeit.timeit(r))
0.359163158616
```

xrange: execution speed is faster

```
>>> r = 'xrange(10)'
>>> print(timeit.timeit(r))
0.232242625909
```

Python 3:

```
>>> r = 'range(10)'
>>> print(timeit.timeit(r))
0.21158485599971755
```

```
>>> x = xrange(1,20,2)
>>> for i in x:
...     print(i)
...
1
3
5
7
9
11
13
15
17
19
```

```
>>> r = range(10)
>>> r = range(1,20,2)
>>> for i in r:
...     print(i)
...
1
3
5
7
9
11
13
15
17
19
```

https://stackoverflow.com/questions/94935/what-is-the-difference-between-range-and-xrange-functions-in-python-2-x