In [6]:
```
a=45678
for  i in a:
    print(i)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[6], line 2
      1 a=45678
----> 2 for  i in a:
      3     print(i)

TypeError: 'int' object is not iterable
```

In [8]:
```
digits=[int(d) for d in str(a)]
print(digits)
for digit in digits:
    print(digit)
```

```
[4, 5, 6, 7, 8]
4
5
6
7
8
```

In [13]:
```
print(dir(digits))
```

```
['__add__', '__class__', '__class_getitem__', '__contains__', '__delattr__', '__delitem__', '__dir__', '__doc__', '__eq__', '__format__', '__ge__', '__getattribute__', '__getitem__', '__getstate__', '__gt__', '__hash__', '__iadd__', '__imul__', '__init__', '__init_subclass__', '__iter__', '__le__', '__len__', '__lt__', '__mul__', '__ne__', '__new__', '__reduce__', '__reduce_ex__', '__repr__', '__reversed__', '__rmul__', '__setattr__', '__setitem__', '__sizeof__', '__str__', '__subclasshook__', 'append', 'clear', 'copy', 'count', 'extend', 'index', 'insert', 'pop', 'remove', 'reverse', 'sort']
```

In [14]:
```
names=['Era','Tom','Hinton']
```

In [15]:
```
Looper=names.__iter__()
print(type(names))
```

```
<class 'list'>
```

In [16]:
```
print(next(Looper))
```

```
Era
```

In [17]:
```
print(next(Looper))
```

```
Tom
```

In [18]:
```
print(next(Looper))
```

```
Hinton
```

In [19]:
```
print(next(Looper))
```

```
---------------------------------------------------------------------------
-
StopIteration                             Traceback (most recent call las
t)
Cell In[19], line 1
----> 1 print(next(Looper))

StopIteration:
```

In [20]:
```
print(Looper)
```

```
<list_iterator object at 0x796e1058f820>
```

In [25]:
```
Looper=iter(names)
while True:
    try
```

```
  Cell In[25], line 2
    while True
               ^
SyntaxError: expected ':'
```

In [30]:
```
numbers=[1,2,3,4,5]
print(list(map(lambda x: x*x, numbers)))
```

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

In [31]:
```
print(list(filter(lambda x: x>3, numbers)))
```

```
[4, 5]
```

In [32]:
```
names = ["ALICE", "BoB", "chARlie", "DAVID"]
lowercase_names = [name.lower() for name in names]

print(lowercase_names)
```

```
['alice', 'bob', 'charlie', 'david']
```

In [35]:
```
numbers=[1,2,3,4,5]
from functools import reduce
print(reduce(lambda x,y:x+y,numbers))
```

```
15
```

In [36]:
```
from functools import reduce

numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

# Square even numbers and then sum them up
result = reduce(lambda x, y: x + y, map(lambda x: x**2, filter(lambda x:

print(result)  # Output: 220
```

```
220
```

In [ ]:
```

```