In [1]:
```
a = [1,2,3,4]
b = [3,4,5,6]
c = list()
d = list()
```

In [2]:
```
#Method 01
for i in range(0,len(a)):
    for j in range(0,len(b)):
        if a[i] == b[j]:
            c.append(a[i])
print(c)
```

```
[3, 4]
```

In [3]:
```
#Method 02 (Shortest Code)
for i in a:
    if i in b:
            d.append(i)
print(d)
```

```
[3, 4]
```

In [4]:
```
# Intersection of two lists
list1 = list()
list2 = list()
intersection = list()
```

In [5]:
```
# Taking Input for first list
n = int(input("Enter the no.of elements:- "))
for i in range(0,n):
    number = int(input("Enter the element:- "))
    list1.append(number)
```

In [6]:
```
# Taking Input for Second list
n = int(input("Enter the no.of elements:- "))
for i in range(0,n):
    number = int(input("Enter the element:- "))
    list2.append(number)
```

In [7]:
```
# Taking Intersection of first list and second list
# Method 01 for Intersection of lists
for i in list1:
    if i in list2:
            intersection.append(i)
print(intersection)
```

```
[5, 8, 6]
```

In [8]:
```
#Method 02 for Intersection of lists
e = list(set(list1) & set(list2))
print(e)
```

```
[8, 5, 6]
```

In [9]:
```
#Method 03 for Intersection of lists
f = [1,2,3,4]
g = [3,4,5,6]
h = [1,2,3,6]
i = set(f).intersection(g,h)
print(i)
```

```
{3}
```

In [10]:
```
# lambda function named 'x' that takes one argument 'a' and performs as
x = lambda a: a + 10
print(x(5))
```

```
15
```

In [11]:
```
print(x(90))
```

```
100
```

In [12]:
```
#Method 04 for Intersection of lists
intersection2 = list(filter(lambda x: x in f,g))
print(intersection2)
```

```
[3, 4]
```