List comprehension

```
li = []

for i in range(10):
    li.append(i)

print(li)
```

```
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

```
# Adding i in the list
# From where: The for loop

# new = ["What do you want to add" -> "From where do you want to add"]

new = [i for i in range(10)]

new
```

```
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

```
# Adding only even elements

li = []

for i in range(10):
    if i % 2 == 0:
        li.append(i)

print(li)
```

```
[0, 2, 4, 6, 8]
```

```
new = [i for i in range(10) if i % 2 == 0]

new
```

```
[0, 2, 4, 6, 8]
```

2D lists problem solving

```
runs = [[100, 99, 20], [200, 150, 90, 180], [20, 50, 100]]
```

```
# Challenge: Print maximum in odi, test and t20 separately

# Maximum in whole list

# Find total runs using indexes


# Find max of each format

for i in runs:
    print(max(i))
```

```
100
200
100
```

```
for i in runs:
    print(min(i))
```

```
20
90
20
```

```
# HW: Find maximum in each format without max function


runs
```

```
[[100, 99, 20], [200, 150, 90, 180], [20, 50, 100]]
```

```
maxi_list = []

for i in runs:
    maxi_list.append(max(i))

max(maxi_list)
```

```
200
```

```


odi = [100, 99, 20, 200]

maxi = odi[0]

for i in odi:
    # Find if any element is larger than maxi
    if i > maxi:
        maxi = i

maxi
```

```
200
```

```
# without range

for i in runs:
    #maxi =
    for j in i:
        print(j)
```

```
100
99
20
200
150
90
180
20
50
100
```

```
# Indexes

# Iterating in 2D list

maxi = runs[0][0]
for i in range(len(runs)):
    for j in range(len(runs[i])):
        #check for maxi
        if runs[i][j] > maxi:
            maxi = runs[i][j]
print(maxi)
```

```
200
```

```
# HW find total runs uisng indexes and without using sum function


# Quiz

l = [[1,2,3] , [5,6] ]
print(len(l))
```

```
2
```

```
l = [[1,2,3] , [5,6] ]
print(l[1])
```

```
[5, 6]
```

Question:

You are a data scientist at ICC and you need to find the runs scored by Sachin and Ganguly in partnerships, also store them in a new list

```
# len of both lists is same: Row * Col

sachin = [
        [1, 2, 3],
        [4, 5, 6],
        [7, 8, 9],
        [10, 11, 12]
]

ganguly = [
        [-1, 2, 3],
        [4, -5, 6],
        [7, 8, -9],
        [10, 11, 12]
]

# 0 4 6
# 8 0 12
# 14 16 0
# 20 22 24


# Traversing on sachin

sachin
```

```
[[1, 2, 3], [4, 5, 6], [7, 8, 9], [10, 11, 12]]
```

```
for i in range(len(sachin)):
    print(sachin[i])
```

```
[1, 2, 3]
[4, 5, 6]
[7, 8, 9]
[10, 11, 12]
```

```
# Getting indexes of runs
```

```
for i in range(len(sachin)):
    for j in range(len(sachin[i])):
        print(i, j)
```

```
0 0
0 1
0 2
1 0
1 1
1 2
2 0
2 1
2 2
3 0
3 1
3 2
```

```
# Traversing on ganguly
```

```
ganguly
```

```
[[-1, 2, 3], [4, -5, 6], [7, 8, -9], [10, 11, 12]]
```

```
for i in range(len(ganguly)):
    print(ganguly[i])
```

```
[-1, 2, 3]
[4, -5, 6]
[7, 8, -9]
[10, 11, 12]
```

```
for i in range(len(ganguly)):
    for j in range(len(ganguly[i])):
        print(i, j, ganguly[i][j])
    print()
```

```
0 0 -1
0 1 2
0 2 3

1 0 4
1 1 -5
1 2 6
```

```
2 0 7
2 1 8
2 2 -9

3 0 10
3 1 11
3 2 12
```

```
for i in range(len(ganguly)):
    for j in range(len(ganguly[i])):
        print(ganguly[i][j], end=" ")
    print()
```

```
-1 2 3
4 -5 6
7 8 -9
10 11 12
```

```
for i in range(len(ganguly)):
    for j in range(len(ganguly[i])):
        print(sachin[i][j], end=" ")
    print()
```

```
1 2 3
4 5 6
7 8 9
10 11 12
```

```
for i in range(len(ganguly)):
    for j in range(len(ganguly[i])):
        print(i, j, end=" ")
    print()
```

```
0 0 0 1 0 2
1 0 1 1 1 2
2 0 2 1 2 2
3 0 3 1 3 2
```

```
for i in range(len(sachin)):
    for j in range(len(sachin[i])):
        print(i, j, end=" ")
    print()
```

```
0 0 0 1 0 2
1 0 1 1 1 2
```

```
2 0 2 1 2 2
3 0 3 1 3 2
```

```
for i in range(len(ganguly)):
    for j in range(len(ganguly[i])):
        print(sachin[i][j], ganguly[i][j], end=" ")
    print()
```

```
1 -1 2 2 3 3
4 4 5 -5 6 6
7 7 8 8 9 -9
10 10 11 11 12 12
```

```
## Adding 2 matrix
```

```
for i in range(len(ganguly)):
    for j in range(len(ganguly[i])):
        print(sachin[i][j] + ganguly[i][j], end=" ")
    print()
```

```
0 4 6
8 0 12
14 16 0
20 22 24
```

```
# Make a list after adding them
```

```
partnership = []

for i in range(len(ganguly)):
    row = []
    for j in range(len(ganguly[i])):
        total = sachin[i][j] + ganguly[i][j]
        row.append(total)
    partnership.append(row)

partnership
```

```
[[0, 4, 6], [8, 0, 12], [14, 16, 0], [20, 22, 24]]
```

Check for Identity Matrix

- You are given a N X N square integer matrix A. You have to tell whether A is an identity matrix or not.
- Identity matrix is a special square matrix whose main diagonal elements are equal to 1 and all other elements are 0.

Input:

- First and only argument is an integer matrix A.

Output:

- Return 1 if A is an identity matrix, else return 0.

```
A = [[1, 1],
  [0, 1]]

# Iterate on the list

for i in range(len(A)):
    for j in range(len(A[i])):
        print(i, j, end=" ")
    print()
```

```
0 0 0 1
1 0 1 1
```

```
## i == j  # main daigonal

## i != j # For rest of the elements
```

```
A = [[1, 1],
  [0, 1]]

n = len(A)
for i in range(n):
    for j in range(n):
        print(A[i][j], end=" ")
    print()
```

```
1 1
0 1
```

```
# print diagonal elements
```

```
A
```

```
[[1, 1], [0, 1]]
```

```
n = len(A)
for i in range(n):
    for j in range(n):
        if i == j:
            print(A[i][j])
```

```
1
1
```

```
# non diagonal
```

```
A
```

```
[[1, 1], [0, 1]]
```

```
n = len(A)
for i in range(n):
    for j in range(n):
        if i != j:
            print(A[i][j])
```

```
1
0
```

```
# Final code
```

```
def identity(A):
    n = len(A)

    for row in range(n):
        for col in range(n):

            # Check for main diagonal
            if row == col and A[row][col] != 1:
                return 0

            # Check for non diagonal
            if row != col and A[row][col] != 0:
                return 0
    return 1

identity(A)
```

```
0
```

```
A
```

```
[[1, 1], [0, 1]]
```

```
A = [[1, 0, 0],
     [0, 1, 0],
     [0, 0, 1]]
```

```
identity(A)
```

```
1
```

For given 2 n * m matrix check if both are equal

```
mat1 = [
        [1, 2, 3],
        [4, 5, 6],
        [7, 8, 9],
        [10, 11, 12]
]

mat2 = [
        [1, 2, 3],
        [4, 5, 6],
        [7, 8, 9],
        [10, 11, 12]
]
```

```
mat1
```

```
[[1, 2, 3], [4, 5, 6], [7, 8, 9], [10, 11, 12]]
```

```
mat2
```

```
[[1, 2, 3], [4, 5, 6], [7, 8, 9], [10, 11, 12]]
```

```
# ==
```

```
mat1 == mat2
```

```
True
```

```
# Since sizes are same so corresponding index will be same

for row in range(len(mat1)):
    for col in range(len(mat1[row])):
```

```
        print(row, col, end=" ")
    print()
```

```
0 0 0 1 0 2
1 0 1 1 1 2
2 0 2 1 2 2
3 0 3 1 3 2
```

```
for row in range(len(mat2)):
    for col in range(len(mat1[row])):
        print(row, col, end=" ")
    print()
```

```
0 0 0 1 0 2
1 0 1 1 1 2
2 0 2 1 2 2
3 0 3 1 3 2
```

```
# Get the values
```

```
for row in range(len(mat2)):
    for col in range(len(mat1[row])):
        print(mat1[row][col], end=" ")
    print()
```

```
1 2 3
4 5 6
7 8 9
10 11 12
```

```
for row in range(len(mat2)):
    for col in range(len(mat1[row])):
        print(mat2[row][col], end=" ")
    print()
```

```
1 2 3
4 5 6
7 8 9
10 11 12
```

```
for row in range(len(mat2)):
    for col in range(len(mat1[row])):
        print(mat2[row][col] == mat1[row][col], end=" ")
    print()
```

```
True True True
True True True
```

```
True True True
True True True
```

```
ans = True
for row in range(len(mat2)):
    for col in range(len(mat1[row])):
        # check for equals
        if mat1[row][col] != mat2[row][col]:
            ans = False
if ans == True:
    print("Same")
else:
    print("Not Same")
```

```
Same
```