Two-Dimensional Lists

- **Two-dimensional list**: a list that contains other lists as its elements
  - Also known as nested list
  - Common to think of two-dimensional lists as having rows and columns
  - Useful for working with multiple sets of data
- To process data in a two-dimensional list need to use two indexes
- Typically use nested loops to process

Creating Two-Dimensional Lists

grid = [[1, 3, 5, 7], [2, 4, 6, 8], [5, 10, 15, 20]]

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>3</th>
<th>5</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>grid[0]</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td>8</td>
</tr>
<tr>
<td>grid[1]</td>
<td>5</td>
<td>10</td>
<td>15</td>
<td>20</td>
</tr>
</tbody>
</table>
### Computing Number of Rows/Columns

grid = [[1, 3, 5, 7], [2, 4, 6, 8], [5, 10, 15, 20]]

- **How do we calculate the number of rows in a 2-D list?**
  - `len(grid)` = # of rows

- **How do we calculate the number of columns in a 2-D list?**
  - `len(grid[rowid])` # use rowid = 0 if you're unsure which row

### Assigning Values to a 2-D List

```python
# this program assigns random numbers to a two-dimensional list.
import random

# constants for rows and columns
ROWS = 3
COLS = 4

def main():
    # create a two-dimensional list.
    values = [[0, 0, 0, 0],
              [0, 0, 0, 0],
              [5, 0, 0, 0]]

    # fill the list with random numbers.
    for r in range(ROWS):
        for c in range(COLS):
            values[r][c] = random.randint(1, 100)

    # display the random numbers.
    print(values)

# call the main function.
main()
```

Program Output
[[99, 46, 8, 19], [1, 52, 38, 48], [55, 4, 88, 78]]
Printing Values in a 2-D List

```python
def main():
    # Create a two-dimensional list.
    values = [[4, 17, 24, 1],
              [46, 21, 54, 10],
              [54, 92, 10, 100]]

    # Print the 2-D list, 1 value per line
    for r in range(len(values)):
        for c in range(len(values[r])):
            print(values[r][c])

    # Call the main function.
    main()
```

Practice

Using Nov12.py from my Public directory, fill in the code for the 3 functions listed:
- `sumAll` – returns the sum of all elements in the 2-D list
- `sumColumns` – prints out the sums of each column
- `maxRow` – compute the sum of each row and return the index and sum of the maximum row

Printing Values in a 2-D List

```python
def main():
    values = [[4, 17, 24, 1],
              [46, 21, 54, 10],
              [54, 92, 10, 100]]

    for r in range(len(values)):
        for c in range(len(values[r])):
            print(f'{values[r][c]:6d}', end=' ')  # Output

    print()  # Call the main function
    main()
```

Tic-Tac-Toe

- Two player game, X and O
- Take turns marking the spaces in a 3×3 grid.
- The player who succeeds in placing three respective marks in a horizontal, vertical, or diagonal row wins the game
Tic-Tac-Toe

- Use a 3x3 grid of numbers to store the game of tic-tac-toe
- Initial grid is filled with all zeros
- A move for X uses a 1 on the board.
- A move for O uses a -1 on the board.
  - Makes life easier later on.

What other functions do we need?

- Think about the following:
  - How do moves get saved on our board?
  - How to determine if the game is over?
  - Can we make helper functions for the winning scenario functions?

Code for Printing Board

```python
def main():
    grid = [[0,0,0],[0,0,0],[0,0,0]]
    print_board(grid)

def print_board(board):
    for row in range(0, len(board)):
        for col in range(0, len(board[row])):
            if board[row][col] == 1:
                print("X", end = " ")
            elif board[row][col] == 0:
                print("-", end = " ")
            else:
                print("O", end = " ")
        print()

main()
```