Announcements

Reminder:
- Program 7 due tomorrow by 11:55pm

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]]
```

```
<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>5</td>
<td>7</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>6</td>
<td>8</td>
</tr>
<tr>
<td>5</td>
<td>10</td>
<td>15</td>
<td>20</td>
</tr>
</tbody>
</table>
```
Accessing Individual Elements

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

grid[0] →  
<table>
<thead>
<tr>
<th>1</th>
<th>3</th>
<th>5</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>4</td>
<td>6</td>
<td>8</td>
</tr>
</tbody>
</table>

grid[1] →  
<table>
<thead>
<tr>
<th>3</th>
<th>5</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>6</td>
<td>8</td>
</tr>
</tbody>
</table>

grid[2] →  
<table>
<thead>
<tr>
<th>5</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>8</td>
</tr>
<tr>
<td>10</td>
<td>15</td>
</tr>
</tbody>
</table>

Computing Number of Rows/Columns

• 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 2-dimensional list.
import random

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

# Create a two-dimensional list.
values = [[0, 0, 0, 0],
          [0, 0, 0, 0],
          [0, 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)
```

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, 41],
              [46, 21, 54, 10],
              [84, 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()
```

### Output

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>17</td>
<td>24</td>
<td>41</td>
</tr>
<tr>
<td>46</td>
<td>21</td>
<td>54</td>
<td>10</td>
</tr>
<tr>
<td>84</td>
<td>92</td>
<td>10</td>
<td>100</td>
</tr>
</tbody>
</table>

Printing Values in a 2-D List

```python
def main():
    values = [[4, 17, 24, 41],
              [46, 21, 54, 10],
              [84, 92, 10, 100]]

    for r in range(len(values)):
        for c in range(len(values[r])):
            print(format(values[r][c], "d"), end="")

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

### Program Output

```
4 17 24
46 21 54 10
84 92 10 100
```

Sum of Rows

- Write a function to print the sum of each row in your table.

Practice

Using Nov15.py from my Box.com 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
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.