Create an Empty List

Completely Empty:

`list = []`

Null values in a certain length:

`list = [None] * length`

Create an empty 2D list (using List Comprehension)

`empty_arr = [[None] * columns for i in range(rows)]`

Useful List-Building Functions

Input_List

```
def input_list(length = 6): # Create a user-input list
    list = [None] * length
    print(f'Forming a list. Please enter {length} numbers: ')
    for item in range(len(list)):
        list[item] = int(input())
    return list
```

Random_List

```
def random_list(length = 6, max = 100): # Create a random list
    list = [None] * length
    for item in range(len(list)):
        list[item] = random.randint(0, max)
    return list
```

Random_into_2D_list

```
# This function overwrites the original list!
def rng_into_2d_arr(two_d_arr): # Insert random values into en empty 2D list
    for row in range(len(arr)):
        for column in range(len(arr[row])):
            arr[row][column] = random.randint(1, 99)
```

View_as_Matrix

```
def view_as_matrix(two_d_arr): # View a 2D List as a Matrix
    matrix = ''
    for row in range(len(two_d_arr)):
        for column in range(len(two_d_arr[row])):
            matrix += f'{str(two_d_arr[row][column]).rjust(2)} '
        matrix += '\n'
    return matrix
```

3. Tuple

Creating / Packing

`opt1 | tuple = num1, num2, num3` `opt2 | tuple = (num1, num2)` `empty | tuple = ()`

Unpacking

Tuple_Unpacking

```
t1 = (10, 20, 30)
num1, num2, num3 = t1
print(num1 + num2 + num3) # Output will be: 60
```

Convert from List to Tuple

`List_To_Tuple | tup1 = tuple(list1)`

Main

`if __name__ == '__main__':`