



INTRO TO PYTHON FOR DATA SCIENCE

# Python Lists



# Python Data Types

- `float` – real numbers
- `int` – integer numbers
- `str` – string, text
- `bool` – `True`, `False`

```
In [1]: height = 1.73
```

```
In [2]: tall = True
```

- Each variable represents single value



# Problem

- Data Science: many data points
- Height of entire family

```
In [3]: height1 = 1.73
```

```
In [4]: height2 = 1.68
```

```
In [5]: height3 = 1.71
```

```
In [6]: height4 = 1.89
```

- Inconvenient



# Python List

# [a, b, c]

```
In [7]: [1.73, 1.68, 1.71, 1.89]
```

```
Out[7]: [1.73, 1.68, 1.71, 1.89]
```

```
In [8]: fam = [1.73, 1.68, 1.71, 1.89]
```

```
In [9]: fam
```

```
Out[9]: [1.73, 1.68, 1.71, 1.89]
```

- Name a collection of values
- Contain any type
- Contain different types



# Python List

# [a, b, c]

```
In [10]: fam = ["liz", 1.73, "emma", 1.68, "mom", 1.71, "dad", 1.89]
```

```
In [11]: fam
```

```
Out[11]: ['liz', 1.73, 'emma', 1.68, 'mom', 1.71, 'dad', 1.89]
```

```
["liz", 1.73]  
["emma", 1.68]  
["mom", 1.71]  
["dad", 1.89]
```



# Python List

# [a, b, c]

```
In [10]: fam = ["liz", 1.73, "emma", 1.68, "mom", 1.71, "dad", 1.89]
```

```
In [11]: fam
```

```
Out[11]: ['liz', 1.73, 'emma', 1.68, 'mom', 1.71, 'dad', 1.89]
```

```
In [11]: fam2 = [
    ["liz", 1.73],
    ["emma", 1.68],
    ["mom", 1.71],
    ["dad", 1.89]]
```

```
In [12]: fam2
```

```
Out[12]: [['liz', 1.73], ['emma', 1.68],
          ['mom', 1.71], ['dad', 1.89]]
```



# List type

```
In [13]: type(fam)
Out[13]: list
```

```
In [14]: type(fam2)
Out[14]: list
```

- Specific functionality
- Specific behavior



INTRO TO PYTHON FOR DATA SCIENCE

**Let's practice!**