



# Lecture 12

---

Table Examples

# Weekly Goals

---

- **Monday**
    - No class
  - **Today**
    - Table review
    - Table examples
  - **Friday**
    - Conditionals and iteration
    - Simulation
-

# **Review: Pivot and Join**

# Pivot

---

- Cross-classifies according to two categorical variables
- Produces a grid of counts or aggregated values
- Two required arguments:
  - First: variable that forms column labels of grid
  - Second: variable that forms row labels of grid
- Two optional arguments (include **both** or **neither**)
  - **values**='column\_label\_to\_aggregate'
  - **collect**=function\_to\_aggregate\_with

(Demo)

---

# Group or Pivot?

---

For cross-classification:

## Pivot Table

- One combo of grouping variables **per entry**
- **Two** grouping variables: columns and rows
- Aggregate values of **values column**
- Missing combos = **0**  
(or empty string)

## Grouped Table

- One combo of grouping variables **per row**
  - **Any number** of grouping variables
  - Aggregate values of **all other columns** in table
  - Missing combos **absent**
-

# Joining Two Tables

```
drinks.join('Cafe', discounts, 'Location')
```

Match rows in  
this table ...

... using values  
in this column ...

... with rows in  
that table ...

... using values  
in that column.

Columns from  
both tables

**drinks**

Drink	Cafe	Price
Milk Tea	Asha	5.5
Espresso	Strada	1.75
Latte	Strada	3.25
Espresso	FSM	2

**discounts**

Coupon	Location
10%	Asha
25%	Strada
5%	Asha

The joined column is  
sorted automatically

Cafe	Drink	Price	Coupon
Asha	Milk Tea	5.5	10%
Asha	Milk Tea	5.5	5%
Strada	Espresso	1.75	25%
Strada	Latte	3.25	25%

# Important Table Methods

---

`t.select(column, ...) or t.drop(column, ...)`

`t.take([row_num, ...]) or t.exclude([row_num, ...])`

`t.sort(column, descending=False, distinct=False)`

`t.where(column, are.condition(...))`

`t.apply(function_name, column, ...)`

`t.group(column) or t.group(column, function_name)`

`t.group([column, ...]) or t.group([column, ...], function_name)`

`t.pivot(cols, rows) or t.pivot(cols, rows, vals, function_name)`

`t.join(column, other_table, other_table_column)`

<http://data8.org/sp20/python-reference.html>

---

# Examples