



**DSFA**  
Spring 2021

# Lecture 12

---

Table Examples and Advanced Where

# Announcements

---

- Project 1 out, part 1 due 3/19, part 2 due 3/26.
    - Get started early! Don't put it off!
    - You can work in pairs.
    - Vocareum is not Google Docs. You cannot both work on the assignment at the same time and have both people's work be saved.
  - This week:
    - No class Wednesday (Wellness day)
    - No labs Wednesday or Thursday
    - No office hours Tuesday or Wednesday
-

# Prelim 1

---

Prelim 1 on Tuesday, March 16, 8:30PM-10PM

- Here in this room (Call Auditorium) for all Ithaca resident students (whether in-person or online); assigned seating
  - Same time online for all Ithaca non-resident students
  - Coverage: From Lecture 1 to Lecture 11 (last Friday)
  - Format:
    - Short answer (e.g. write a line of Python that does this...), multiple choice
    - Closed `book`, but you may bring one page (8.5" x 11") double-sided set of notes that you write yourself
    - You will be provided with a sheet of standard Python function definitions
-

# Prelim 1 resources

---

- Study guide (with the list of table functions to be given to you in the exam) will be posted today on Canvas
  - Practice exam to be posted today
  - Review session/`ask me anything (about 1380)' session late this week TBA (Saturday?)
-

# Combining Table Methods

# Important Table Methods

---

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

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

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

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

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

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

`t.groups([column, ...])` or `t.groups([column, ...], function)`

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

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

---

# Apply

---

The `apply` method creates an array by calling a function on every element in input column(s)

- First argument: Function to apply
- Other arguments: The input column(s)

```
table_name.apply(function_name, 'column_label')
```

(Demo)

---

# Group

---

The **group** method aggregates all rows with the same value for a column into a single row in the result

- First argument: Which column to group by
- Second argument: (Optional) How to combine values
  - **len** — number of grouped values (default)
  - **sum** — total of all grouped values
  - **list** — list of all grouped values

(Demo)

---



# Grouping By Two Columns

---

The `groups` method can also aggregate all rows that share the combination of values in multiple columns

- First argument: A list of which columns to group by
- Second argument: (Optional) How to combine values

(Demo)

---

# 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_with_which_to_aggregate`

(Demo)

---

# Joining Two Tables

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

Keep all rows in the table that have a match ...

... for the value in this column ...

... somewhere in this other table's ...

... column that contains matching values.

**drinks**

Drink	Cafe	Price
Milk tea	Panda Tea	4
Espresso	Gimme	2
Latte	Gimme	3
Espresso	Cafe Gola	2

**discounts**

Coupon	Location
25%	Panda Tea
50%	Gimme
5%	Gimme

The joined column is sorted automatically

(Demo)

Cafe	Drink	Price	Coupon
Gimme	Espresso	2	50%
Gimme	Espresso	2	5%
Gimme	Latte	3	50%
Gimme	Latte	3	5%
Panda Tea	Milk Tea	4	25%

# Discussion Question

---

Generate a table with one row per cafe that has the name and discounted price of its cheapest discounted drink

**drinks**

Drink	Cafe	Price
Milk tea	Panda Tea	4
Espresso	Gimme	2
Coffee	Gimme	3
Espresso	Cafe Gola	2

**discounts**

Coupon	Location
25%	Panda Tea
50%	Gimme
5%	Gimme

**cheapest**

Cafe	Drink	Discounted Price
Panda Tea	Milk Tea	3
Gimme	Espresso	1

---

# **Booleans and Advanced Where**

# Comparison Operators

---

The result of a comparison expression is a **bool** value

`x = 2`

`y = 3`

Assignment statements

`x > 1`

`x > y`

`y >= 3`

Comparison  
expressions

`x == y`

`x != 2`

`2 < x < 5`

`t.where(array_of_bool_values)` returns a table with only the rows of `t` for which the corresponding **bool** is **True**.

---

(Demo)

When poll is active, respond at [Pollev.com/dsfa](https://Pollev.com/dsfa)

Text **DSFA** to **22333** once to join

**'Dog' > 'Cat' > 'Catastrophe'**

True

False



# Some Table Questions



# Table Questions

---

start	end	duration
Metropolitan Ave & Bedford Ave	Bedford Ave & Nassau Ave	6.06667
Lafayette St & E 8 St	2 Ave & E 104 St	35.7
Schermerhorn St & Court St	Court St & Nelson St	5.46667

- What is the average duration of all trips?
  - What is the average duration of all trips that started and ended at the same station?
  - What is the average duration of all trips that started and ended at different stations?
  - What is the name of the station where the most rentals ended? (Assume no ties.)
  - For how many stations was the average duration of a trip ending at that station at least 10 minutes?
-

# Take-Home Question

---

Generate a table of the names of the starters for each team

TEAM	C	PF	PG	SF	SG
Atlanta Hawks	Al Horford	Paul Millsap	Jeff Teague	Thabo Sefolosha	Kyle Korver
Boston Celtics	Tyler Zeller	Jonas Jerebko	Avery Bradley	Jae Crowder	Evan Turner
Brooklyn Nets	Andrea Bargnani	Thaddeus Young	Jarrett Jack	Joe Johnson	Bojan Bogdanovic
Charlotte Hornets	Al Jefferson	Marvin Williams	Kemba Walker	Michael Kidd-Gilchrist	Nicolas Batum
Chicago Bulls	Joakim Noah	Nikola Mirotic	Derrick Rose	Doug McDermott	Jimmy Butler
Cleveland Cavaliers	Tristan Thompson	Kevin Love	Kyrie Irving	LeBron James	Iman Shumpert
Dallas Mavericks	Zaza Pachulia	David Lee	Deron Williams	Chandler Parsons	Justin Anderson
Denver Nuggets	JJ Hickson	Kenneth Faried	Jameer Nelson	Danilo Gallinari	Gary Harris
Detroit Pistons	Aron Baynes		Reggie Jackson	Stanley Johnson	Jodie Meeks
Golden State Warriors	Andrew Bogut	Draymond Green	Stephen Curry	Andre Iguodala	Klay Thompson

---