

**INSTRUCTIONS**

You have 1 hour and 50 minutes to complete the exam. The exam is worth a total of 80 points.

- The exam is closed book, closed notes, closed computer/calculator, except for the provided cheat sheet.
- Mark your answers on the exam itself in the spaces provided.
- If you need to use the restroom, bring your phone and exam to the front of the room.

For questions with **circular bubbles**, you should fill in exactly *one* choice.

- You must choose either this option
- Or this one, but not both!

**Preliminaries**

You can complete these questions before the exam starts.

- (a) What is your full name?

- (b) What is your student ID number?

- (c) Who is sitting to your left? (Write *no one* if no one is next to you.)

- (d) Who is sitting to your right? (Write *no one* if no one is next to you.)

- (e) **UC Berkeley Honor Code**

As a member of the UC Berkeley community, I act with honesty, integrity, and respect for others. By signing my name below, I affirm that all of my answers are my own work, and that I have used no external resources (other than the Data 6 cheat sheet) during this exam.

- i. (2.0 pt) **Sign your name in the space provided.**

**1. (9.0 points) True/False**

For each of the following statements, indicate whether it is true or false.

- (a) (1.0 pt) In the Data 6 context, a *variable* is the same thing as a *name* in Python.
- True
- False
- (b) (1.0 pt) The best way to visualize the distributions of two categorical variables is to make a overlaid histogram.
- True
- False
- (c) (1.0 pt) `True and 1 == True` evaluates to `True`.
- True
- False
- (d) (1.0 pt) Python dictionaries can contain other dictionaries.
- True
- False
- (e) (1.0 pt) When using `tbl.join`, the order of the tables affects which rows are included in the joined table.
- True
- False
- (f) (1.0 pt) It is easy for a computer to generate truly random numbers.
- True
- False
- (g) (1.0 pt) “Raw data” is the purest form of data, untouched by humans.
- True
- False
- (h) (1.0 pt) Table methods like `tbl.join`, `tbl.group` and `tbl.pivot` create copies of the table instead of modifying the original tables.
- True
- False
- (i) (1.0 pt) A single row of a table can only contain values from the same data type, but a single column of a table can contain values from multiple different data types.
- True
- False

## 2. (8.0 points) Sunya's Scorebook

Sunya is a middle school teacher who wants to check in on students. She organizes her gradebook as a Python dictionary called `student_scores`, where each student's name is a key and the values are dictionaries with each academic subject (Math, English and History) as a key and the student's score in the subject as the corresponding value.

```
student_scores = {
    "Alice": {"Math": 85, "English": 92, "History": 78},
    "Bob": {"Math": 88, "English": 81, "History": 94},
    "Charlie": {"Math": 91, "English": 78, "History": 88},
    "Diana": {"Math": 92, "English": 87, "History": 85}
}
```

- (a) (3.0 pt) Write a line of code to add a new student, "Eve", who scored 90 in Math, 88 in English, and 80 in History to the `student_scores` dictionary.

### (b) (5.0 points)

Fill in the blanks to write a function named `average_score` that takes a dictionary of student scores and a subject. The function should return the average score of all students in that subject.

```
def average_score(scores, subject):
    total_score = ___
                    (a)
    for student in _____:
                    (b)
        total_score = _____
                    (c)
    average = _____
                    (d)
    return _____
                    (e)
```

- i. Fill in blank (a).

- ii. Fill in blank (b).

- iii. Fill in blank (c).

iv. Fill in blank (d).

v. Fill in blank (e).

### 3. (13.0 points) Wildfires

In the midst of ongoing wildfires in Canada and air quality concerns in the United States, Andy has built a smartphone app to allow people across the country to self-report the air quality in their location using air quality kits that cost \$50. The app collects each user's location (including their exact coordinates) and syncs with their air quality sensor to measure the Air Quality Index (AQI) at multiple times of the day, even if the user is not on the app. Additionally, users can describe their observations about the sky in the "Sky Condition" field.

Andy compiles the air quality reports from the app on July 6th into the `reports` table below.

Location	Longitude	Latitude	AQI	Sky Condition
New York, NY	-74.006	40.722	251	Hazy
Falls Church, VA	-77.1711	38.8823	196	Cloudy
New York, NY	-74.0109	40.7128	212	Hazy
Gary, IN	-87.3372	41.6020	105	Clear
Lexington, KY	-84.5037	38.0406	246	Smoky

... (21405 rows omitted)

#### (a) (7.0 points) Tracking Air Quality

Andy wants to investigate the air quality across the US on July 6th. He decides to visualize the distribution of AQIs based on the reported "Sky Condition".

i. (1.0 pt) What type of variable is "Sky Condition"?

- Numerical discrete
- Numerical continuous
- Categorical ordinal
- Categorical nominal

ii. (4.0 points)

Complete the following code to generate an appropriate graph for visualizing the distribution of AQIs organized by Sky Condition.

```
reports.__(a)__("AQI", group=__(b), __(c))
```

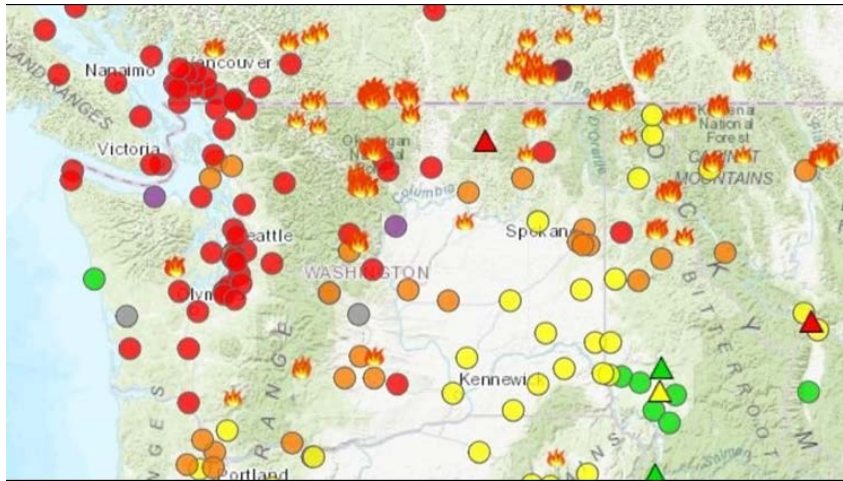
A. Fill in blank (a).

B. Fill in blank (b).

C. Fill in blank (c).

**iii. (3.0 points)**

Looking for inspiration, Andy finds this map showing air quality and wildfire data from Washington state.



**A. (1.0 pt)** Aside from the color of each point, how many variables are encoded in this map?

**B. (2.0 pt)** Describe how each variable (excluding color) is encoded in the map above.

**(b) (6.0 points) Concerns**

Eunice, who has heard about the AQI app from friends, raises concerns about the AQI tracking project.

- i. (6.0 pt)** Describe 2 concerns Eunice may have about Andy's app relating to privacy, accessibility, trustworthiness or other HCE-related considerations, and their importance.



#### 4. (34.0 points) Jonathan has Swiftie Fever

Jonathan is a data analyst for a music streaming service. He's been tasked with analyzing song data from different albums of an artist's career to understand the artist's trajectory and growth. The artist in question is Taylor Swift, a well known singer-songwriter (Jonathan happens to be a big fan). Each of Taylor's albums is represented as an array, where each element of the array is the length of the song in seconds.

You are provided with two arrays that represent her albums, "Red" (2012) and "Folklore" (2020), respectively:

```
red_album = make_array(211, 231, ...)
folklore_album = make_array(201, 351, ...)
```

##### (a) (20.0 points) TayTay's Top Tracks

Jonathan wants to determine if there is a significant difference in the length of songs between these two albums. You've been asked to help Jonathan.

##### i. (6.0 points)

First, write a function called `longer_than_average` that takes an album array as input, and returns the number of songs that are longer than the average length of all songs in that album. You should use a Numpy function to calculate the average length of songs.

```
def longer_than_average(album):
    average_length = _____
                        (a)
    num_long_songs = ___
                        (b)
    for index in np.arange(_____):
                        (c)
        if _____:
            _____
                        (d)
                        (e)
    return num_long_songs
```

A. Fill in blank (a).

B. Fill in blank (b).

C. Fill in blank (c).

D. Fill in blank (d).



**E.** Fill in blank (e).

**ii. (6.0 points)**

You tell Jonathan that it would be simpler to find the number of longer-than-average songs using a table method. Fill in the blanks of the code below to return the number of songs that are longer than the average length of all songs in the album.

```
def longer_than_average_table(album):
    album_tbl = Table().with_column("Song Length", _____)
                                                    (a)
    average_length = np.average(album_tbl._____)
                                                    (b)
    return album_tbl._____("Song Length", _____)._____
                    (c)                                (d)                (e)
```

**A.** Fill in blank (a).

**B.** Fill in blank (b).

**C.** Fill in blank (c).

**D.** Fill in blank (d).

**E.** Fill in blank (e).

- iii. (2.0 pt) Next, write a function called `num_long_songs` that takes an album as input, and returns the number of songs that are longer than 300 seconds.

```
def num_long_songs(album):
    return _____
```

Fill in the blank in the function above.

- iv. (6.0 points)

Finally, write a function called `compare_albums` that takes two albums as input. This function should use the `num_long_songs` function to compare the albums, and return a string with one of the following outcomes:

- “album1\_name has more long songs” if album1 has more long songs
- “album2\_name has more long songs” if album2 has more long songs
- “Both albums have the same number of long songs” otherwise

*Note:* Replace `album1_name` and `album2_name` with the actual album names.

```
def compare_albums(album1, album1_name, album2, album2_name):
    long_songs_album1 = _____
                                (a)

    long_songs_album2 = _____
                                (b)

    if _____:
        (c)
        return _____ " has more long songs"
        (d)

    elif _____:
        (e)
        return _____ " has more long songs."
        (f)

    return "Both albums have the same number of long songs"
```

- A. Fill in blank (a).

- B. Fill in blank (b).

- C. Fill in blank (c).

**D.** Fill in blank (d).

**E.** Fill in blank (e).

**F.** Fill in blank (f).

**(b) (14.0 points) The Eras Economic Effect**

After Jonathan finishes his album analysis, his supervisor Amanda asks him to draft a report on the economic benefits of Taylor Swift's latest tour, "Eras". He is given two tables, `concerts` and `economic_activity`.

The `concerts` table below contains one row for each city where Taylor Swift has performed an Eras tour concert. The table also lists the start and end dates for that stage of the tour, the combined attendance for the city, and the average ticket price.

City	Start Date	End Date	Combined Attendance	Avg. Ticket Price (\$)
Phoenix	3/17/23	3/18/23	156157	256.5
Las Vegas	3/24/23	3/25/23	144100	312.89
Dallas	3/31/23	4/2/23	210607	199.56
Tampa	4/13/23	4/15/23	140025	226.78
Houston	4/21/23	4/23/23	144581	245.68
Atlanta	4/28/23	4/30/23	150642	302.03

... (22 rows omitted)

The `economic_activity` table below contains information about the Gross Domestic Product (GDP) of each major American city for each month in 2023 (so far).

City	Population (Millions)	Month	GDP (Millions of \$)
Atlanta	5.14	January	39485.2
Atlanta	5.15	February	39589.8
Atlanta	5.15	March	39209.3
Atlanta	5.15	April	41730.7
Atlanta	5.16	May	398390.0

... (219 rows omitted)

- i. (1.0 pt) Jonathan believes that, in general, Taylor Swift performing in a city gives that city an immediate economic boost in the form of more hotel, restaurant, and shopping purchases. To test his theory, he wants to create a visualization showing the economic trends over months for particular cities. What type of visualization would be most appropriate here?

- Scatter Plot
- Histogram
- Line Plot
- Bar Chart

ii. (10.0 points)

Additionally, Jonathan has a suspicion that concert attendance is inversely related to the average ticket price for a particular city (that is, the cheaper the tickets are, the more people are likely to attend). To verify this suspicion, Jonathan first needs to normalize (divide) the attendance figures by the city's population and the number of concerts Taylor performed in the city.

Complete the code below to create a table called `attendance`, containing information from both the `concerts` and `economic_activity` tables.

*Note:* You may assume that a `duration` function has already been defined, which takes in two dates as strings and returns the number of days between the two dates (inclusive).

```
tblA = _____
      (a)
tblB = economic_activity._____("City", "Population")._____(_, np.mean)
      (b)                               (c)       (d)

concert_attendance = tblA.join("City", tblB)

days = attendance.______(duration, _____, "End Date")
      (e)                               (f)

normalized_attendance = attendance.column(______) /
      (g)
      (days * attendance.column(______) * 1_000_000)
      (h)

attendance = attendance.______("Normalized Attendance", _____)
      (i)                               (j)
```

- A. Fill in blank (a).

- B. Fill in blank (b).

**C.** Fill in blank (c).

**D.** Fill in blank (d).

**E.** Fill in blank (e).

**F.** Fill in blank (f).

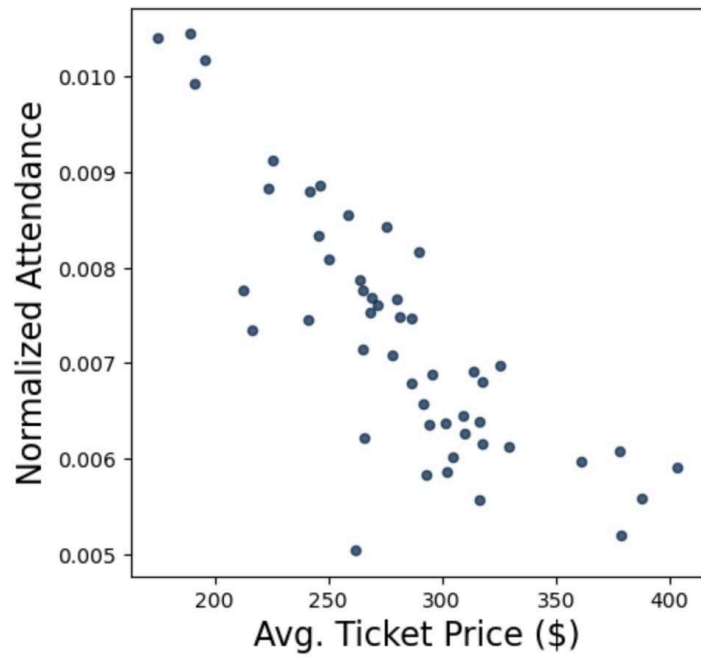
**G.** Fill in blank (g).

**H.** Fill in blank (h).

**I.** Fill in blank (i).

**J.** Fill in blank (j).

- iii. (3.0 pt) Finally, Jonathan uses the `concert_attendance` table to generate the following scatter plot showing the relationship between average ticket price and the normalized concert attendance. Does the scatter plot confirm Jonathan's suspicions? Why or why not?





### 5. (14.0 points) James' Dinner Decider

James needs help deciding where to go out for dinner next week. He's narrowed down his options to the restaurants in the `restaurants` table.

Restaurant	Cuisine	Avg. Price (\$)
Imm Thai	Thai	18.50
Noodle Dynasty	Chinese	20.35
Study Hall	American	25.68
Cholita Linda	Mexican	13.90
Mezzo	American	15.62

... (57 rows omitted)

#### (a) (8.0 points) Restaurant Merry-Go-Round

For each day of the week (Monday through Sunday), James lets Python decide where he will eat according to the following rules:

- i. If it's Tuesday, he must always go to Cholita Linda
- ii. If it's not a Tuesday, pick a restaurant at random from the restaurants in the `restaurant` table and go there.

Complete the skeleton code below so that, after running the code, `choices` is an array of seven restaurant selections corresponding to where James will go for each day of the week. The `restaurants` table has already been loaded in.

```

days = make_array("Monday", "Tuesday", "Wednesday", "Thursday",
                  "Friday", "Saturday", "Sunday")

restaurant_names = _____
                                (a)

choices = make_array()

for i in _____:
    (b)
    if _____:
        (c)
        choices = np._____ (_____, "Cholita Linda")
                                (d)      (e)
    else:
        random_selection = np.random._____ (_____)
                                                (f)      (g)
        choices = np._____ (choices, _____)
                                (h)      (i)

```

- i. Fill in blank (a).

- ii. Fill in blank (b).

**iii.** Fill in blank (c).

**iv.** Fill in blank (d).

**v.** Fill in blank (e).

**vi.** Fill in blank (f).

**vii.** Fill in blank (g).

**viii.** Fill in blank (h).

**ix.** Fill in blank (i).

**(b) (6.0 points) Cholita Linda**

James *really* wants to eat at Cholita Linda, but still wants Python to make a random selection from a set of five restaurant options: Imm Thai, Noodle Dynasty, Mezzo, Cholita Linda, and La Note.

i. (1.0 pt) James chooses a restaurant at random from the restaurant options five hundred times. How many times should we expect “Noodle Dynasty” to be selected, on average?

- 0  
 100  
 200  
 300

ii. (1.0 pt) James chooses a restaurant at random from the restaurant options fifty times. Is James guaranteed to select “Cholita Linda” at least once?

- Yes  
 No

**iii. (4.0 points)**

Complete the while loop below so that the loop only stops after Cholita Linda has been randomly selected from the five options: Imm Thai, Noodle Dynasty, Mezzo, Cholita Linda, and La Note.

```
options = make_array("Imm Thai", "Noodle Dynasty", "Mezzo",
                    "Cholita Linda", "La Note")

choice = ""

while choice _____:
    (a)
    _____ = np.random._____(options)
    (b)                (c)
    print(choice)
```

A. Fill in blank (a).

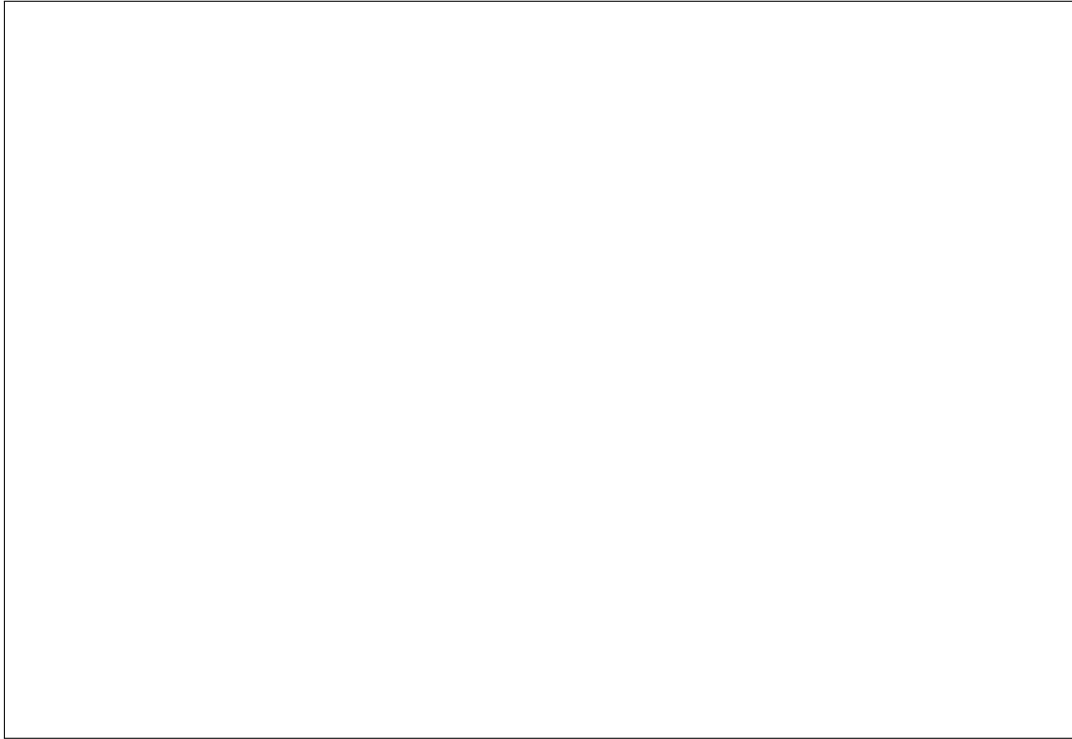
B. Fill in blank (b).

C. Fill in blank (c).

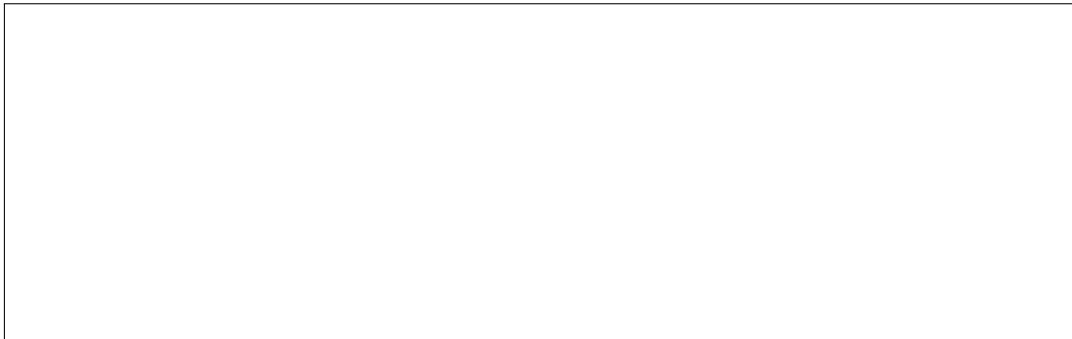
**6. Eunice's Exciting Bonus Questions**

These questions are extra bonus questions, and do not have anything to do with the course content.

(a) (0.0 pt) Draw us a picture :)



(b) (0.0 pt) How was your overall experience in Data 6?



(c) (0.0 pt) What is your favorite restaurant in Berkeley?



(d) (0.0 pt) What is Eunice's favorite TV show?

