Midterm Exam - DSC 10, Summer Session I 2024

Full Name:

PID:

Instructions:

- This exam consists of 9 questions, worth a total of 64 points.
- Write your PID in the top right corner of each page in the space provided.
- Please write **clearly** in the provided answer boxes; we will not grade work that appears elsewhere. Completely fill in bubbles and square boxes; if we cannot tell which option(s) you selected, you may lose points.

  ◯ A bubble means that you should only **select one choice**.

  ☐ A square box means you should **select all that apply**.
- Aside from the provided Reference Sheet, you may not refer to any other resources or technology during the exam (no notes, no calculators).

By signing below, you are agreeing that you will behave honestly and fairly during and after this exam.

Signature:

Version A/B

Please do not open your exam until instructed to do so.

Important: Before proceeding, make sure to rip off the last sheet of paper from this exam packet and read the data description.

Question 1 (11 pts)

a) (2 pts) Which column of `kart` would work best as the index?

◯ `"Ranking"` ◯ `"Team"` ◯ `"Division"` ◯ `"Total Points"`

b) (2 pts) Write a line of Python code that sets the index of `kart` to the column that you identified in part **(a)**.

c) (4 pts) Fill in the blanks below to complete the implementation of the function `division_to_int`, which takes in a string, `division`, the name of a division, and returns the division's number as an `int`. Example behavior is given below.

```
>>> division_to_int("Division 1")
1
>>> division_to_int("Division 2")
2
```

```
def division_to_int(division):
    __(i)__ __(ii)__(division.__(iii)__[__(iv)__])
```

(i):

(ii):

(iii):

(iv):

d) (3 pts) Write a *single* line of code that applies `division_to_int` to the existing `"Division"` column, and updates the `"Division"` column in `kart`.

For the rest of this exam, assume that the changes above *have not* been made and that `kart` is the same DataFrame that was described on the attached information sheet.

Question 2 (4 pts)

Fill in the blanks so that the expression below evaluates to the region in the `"Region"` column that comes first alphabetically.

```
kart.groupby(___(a)___).count().index.___(b)___
```

(a) (2 pts): ◯ `"Ranking"` ◯ `"Team"` ◯ `"Division"` ◯ `"Region"`

(b) (2 pts) : ◯ `max()` ◯ `min()` ◯ `len()` ◯ `split()`

Question 3 (6 pts)

Fill in the blanks so that the expression below evaluates to the `"University"` of the lowest scoring Division 2 team over the whole season.

```
kart[__(a)__].__(b)__(__(c)__).get("University").iloc[0]
```

(a) (2 pts):

(b) (2 pts):

(c) (2 pts):

Question 4 (4 pts)

Consider the scatterplot generated by the following expression:

```
kart.plot(kind="scatter, x="Total Points", y="Races Won")
```

Which of the following questions would you be able to answer from this scatterplot? Select all that apply.

☐ What is the name of the team that has the highest number of races won?

☐ How many teams scored more than 6000 total points?

☐ How many teams scored less than 100 total points per race that they won?

☐ Which Division 2 school scored the most total points?

Question 5 (7 pts)

a) (4 pts) Fill in the blanks so that the sentence below correctly describes the meaning of `mystery`.

"The __(i)__ __(ii)__ of the __(iii)__ __(iv)__."

```
mystery = (kart.get(["Region", "Ranking"])
                .groupby("Region").mean().get("Ranking").min())
```

(i): ◯ average ◯ median ◯ lowest ◯ highest

(ii): ◯ university ◯ team ◯ region ◯ ranking

(iii): ◯ average ◯ median ◯ worst ◯ best

(iv): ◯ university ◯ team ◯ region ◯ ranking

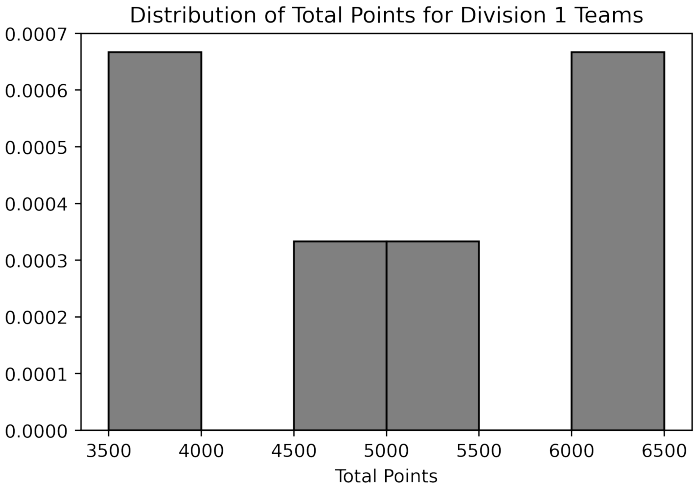
b) (3 pts) Fill in the blank in the code snippet below so that `my_bool` is `True` if there is at least one university with two teams in the same division, and `False` if there are no universities with two teams in the same division. Your answer must include the use of `.groupby()` in order to receive credit.

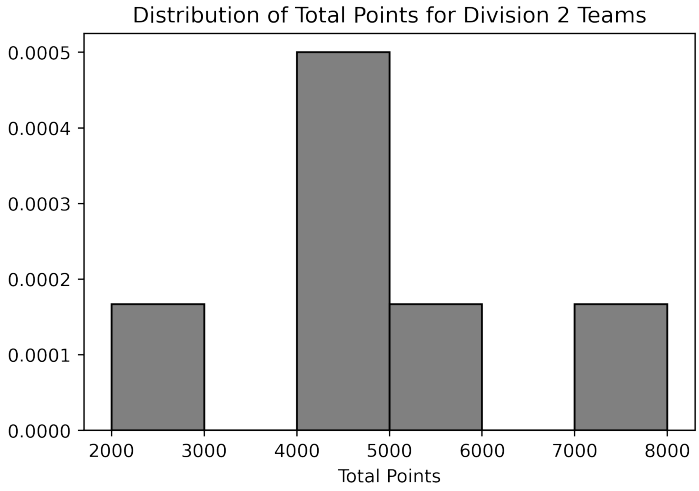
```
my_df = ______
my_bool = my_df.get("Team").max() > 1
```

PID: _____

Question 6 (10 pts)

Below are two density histograms representing the distribution of `Total Points` for Division 1 and the distribution of `Total Points` for Division 2 teams (remember, there are six teams in each division):





a) (4 pts) Assuming we know the bin values for each histogram, what can we conclude from these two histograms? Select all that apply.

☐ The number of teams in the rightmost bin on the histogram that displays the distribution of total points for Division 2 teams

☐ The number of Division 1 teams that scored 4000 points

☐ The number of Division 2 teams that scored between 2000 and 3000 points

☐ The number of Division 2 teams that scored fewer points than the lowest-scoring Division 1 team

b) (3 pts) Suppose that we changed the histogram of total points for Division 2 teams so that the bins were `[2000, 4000)`, `[4000, 6000)`, `[6000, 8000)`. If the bin defined by `[2000, 4000)` contained one team, as it does in the original graph, what would the height of the middle bar (with bin `[4000, 6000)`) be? Do not simplify your answer.

c) (3 pts) Suppose we drew different bins for the histogram of total points for Division 2 teams. If the bin defined by `[2000, 4000)` contained one team, as it does in the original graph, and the bin defined by `[4000, 4500)` contained two teams, what would the height of the bar with bin `[4500, 5000)` be? Do not simplify your answer.

PID: _____

Question 7 (7 pts)

The DataFrame `div_one`, shown below to the **left**, contains the six Division 1 teams from `kart`, with just the `"Team"` and `"Region"` columns selected.

The DataFrame `coach`, shown below to the **right**, contains some additional information about the coaches of the teams in `div_one`, with a piece of info missing.

	Team	Region
0	Triton Circus	Midwest
1	Wolverines	Midwest
2	Golden Bears	East
3	Sooners	Midwest
4	Patriots	West
5	Bruins	West

	Team	Coach	Region
0	Triton Circus	Coach Jason	___
1	Wolverines	Coach Jack	Midwest
2	Golden Bears	Coach Jason	Midwest
3	Sooners	Coach Ashley	East
4	Patriots	Coach Nick	East
5	Bruins	Coach Zoe	South

a) (4 pts) Fill in the blank in the `Region` column with either `"South"`, `"West"`, `"East"`, or `"Midwest"` so that the DataFrame `div_one.merge(coach, on="Region")` has:

8 rows: ☐

9 rows: ☐

10 rows: ☐

11 rows: ☐

b) (3 pts) What is the value of the following Python expression?

```
coach.merge(coach, on="Coach").merge(coach, on="Coach").shape[0]
```

☐

Question 8 (5 pts)

There's been an expansion of Collegiate Mario Kart across the country, so to determine a champion, we'd like to set up a bracket for a Mario Kart tournament and fill it with four college teams from `bigkart`. We won't display `bigkart` for this question, but assume it's a larger version of `kart`: there are 50 rows in `bigkart`, and thus 50 teams. Selection is performed uniformly at random, so that each team has the same chance of being selected. Please leave your answers in **unsimplified** form: answers of the form $(\frac{3}{4}) \cdot (\frac{2}{3})$ or $\left[1 - (\frac{1}{2})^4\right]$ are *preferred*.

a) (2 pts) Assume we populate our tournament by randomly selecting four teams from `bigkart` **with replacement**. What is the probability there are no duplicates among the four teams selected for the tournament? Do not simplify your answer.

b) (3 pts) Now, assume we populate our tournament by randomly selecting four teams from `bigkart` **without replacement**. Additionally, assume 30 are from Division 1 and 20 teams are from Division 2. What is the probability that there is at least one Division 2 team among the four teams selected for the tournament? Do not simplify your answer.

PID: _____

Question 9 (10 pts)

We want to estimate the probability that the University of Michigan is among the four teams selected when schools are selected **without replacement**.

```
schools = np.array(kart.get("University"))
mystery_one = 0
num_trials = 10000
for i in np.arange(num_trials):
    bracket = __(i)__
    if "Michigan" in bracket:
        mystery_one = __(ii)__
mystery_two = mystery_one / num_trials
```

a) Fill in the blanks to complete a simulation.

`(i)` (3 pts):

`(ii)` (2 pts):

b) (2 pts) What is the meaning of `mystery_two` after the code has finished running?

◯ The number of times Michigan was in the tournament

◯ The number of trials we ran

◯ The proportion of times Michigan was in the tournament

◯ None of these answers is what `mystery_two` represents

c) (1 pt) For the next two parts only, imagine we wanted to simulate a 16-team tournament, where teams are selected **with replacement**. Which blank should be filled in?

◯ blank `(i)` ◯ blank `(ii)`

d) (2 pts) What code should be used to fill in the blank you selected above?

Mario Kart

In this exam, you'll work with a data set representing the results of the Collegiate Mario Kart League. Each row represents a team and their performance in the league.
The columns of `kart` are as follows:

- `"Ranking" (int)`: The team's ranking **within their division**. Rankings are from 1 to 6, with the best teams receiving rank 1 and the worst teams receiving rank 6.
- `"Team" (str)`: The name of the team.
- `"University" (str)`: The university the team represents.
- `"Division" (str)`: The division in which the team competes (Division 1 or Division 2). Each division includes six teams.
- `"Total Points" (int)`: The total points scored by the team during the season.
- `"Races Won" (int)`: The total number of races won by the team.
- `"Region" (str)`: The geographic region of the university.

The first few rows of `kart` are shown below, though `kart` has 12 rows in total (since there are 12 teams total, 6 in each division).

	Ranking	Team	University	Division	Total Points	Races Won	Region
0	6	Bruins	UCLA	Division 1	3640	40	West
1	1	Crimson Tide	Alabama	Division 2	7840	108	South
2	4	Gaels	Iona	Division 2	4910	58	East
3	3	Golden Bears	Kutztown	Division 1	5200	65	East
4	3	Hokies	Virginia Tech	Division 2	4650	60	East
5	5	Panthers	UNI	Division 2	4900	56	Midwest

Throughout this exam, we will refer to `kart` repeatedly.

Assume that we have already run `import babypandas as bpd` and `import numpy as np`.