

Exam 2 — Part 2 — 11/6/2024

Instructions

- This part is worth 30 points total. The exam (all three parts) is worth 100 points total.
- You have 50 minutes to complete Parts 1 and 2 of the exam.
- For Parts 1 and 2 of the exam, you may *not* use any outside assistance. These parts of the exam are closed book, closed notes, and closed internet.
- **No collaboration allowed.** All work must be your own.
- **You should have submitted Part 1 before starting Part 2.**
- **Do not discuss the contents of this exam with any midshipmen until it is returned to you.**

Background

This part is based on 2 DataFrames, `track_df` and `meet3_df` :

- Here is `track_df` :

	athlete	team	meet1_400	meet2_400	meet1_800	meet2_800
0	Fazio	Blue	60.7	59.1	135.6	132.0
1	Martell	Gray	62.2	62.6	139.5	133.2
2	Ohlsson	Blue	56.7	54.7	132.9	133.2
3	Tovcach	Blue	63.0	63.2	121.9	122.8
4	OConnell	Gray	61.6	58.5	135.0	133.4
5	Acosta	Gray	59.5	60.4	125.1	123.7

- Here is `meet3_df` :

	athlete	meet3_400	meet3_800
0	Tovcach	62.4	120.9
1	OConnell	59.1	132.3
2	Acosta	60.6	NaN
3	Fazio	58.7	130.1
4	Martell	63.1	134.5
5	Ohlsson	55.2	130.4

Problem 1

What is the result of the code below?

```
track_df.set_index(keys='athlete', drop=True)
```

A.

	index	team	meet1_400	meet2_400	meet1_800	meet2_800
athlete						
Fazio	0	Gray	60.7	59.1	135.6	132.0
Martell	1	Gray	62.2	62.6	139.5	133.2
Ohlsson	2	Blue	56.7	54.7	132.9	133.2
Tovcach	3	Blue	63.0	63.2	121.9	122.8
OConnell	4	Gray	61.6	58.5	135.0	133.4
Acosta	5	Gray	59.5	60.4	125.1	123.7

B.

	athlete	team	meet1_400	meet2_400	meet1_800	meet2_800
athlete						
Fazio	Fazio	Blue	60.7	59.1	135.6	132.0
Martell	Martell	Gray	62.2	62.6	139.5	133.2
Ohlsson	Ohlsson	Blue	56.7	54.7	132.9	133.2
Tovcach	Tovcach	Blue	63.0	63.2	121.9	122.8
OConnell	OConnell	Gray	61.6	58.5	135.0	133.4
Acosta	Acosta	Blue	59.5	60.4	125.1	123.7

C.

	team	meet1_400	meet2_400	meet1_800	meet2_800
athlete					
Fazio	Blue	60.7	59.1	135.6	132.0
Martell	Gray	62.2	62.6	139.5	133.2
Ohlsson	Blue	56.7	54.7	132.9	133.2
Tovcach	Blue	63.0	63.2	121.9	122.8
OConnell	Gray	61.6	58.5	135.0	133.4
Acosta	Gray	59.5	60.4	125.1	123.7

D.

	index	athlete	team	meet1_400	meet2_400	meet1_800	meet2_800
0	0	Fazio	Blue	60.7	59.1	135.6	132.0
1	1	Martell	Gray	62.2	62.6	139.5	133.2
2	2	Ohlsson	Blue	56.7	54.7	132.9	133.2
3	3	Tovcach	Blue	63.0	63.2	121.9	122.8

Problem 2

What is the result of the code below?

```
(  
    track_df  
    .query('(meet1_400 < 60) and (meet2_400 < 60)')  
)
```

A.

	athlete	team	meet1_400	meet2_400	meet1_800	meet2_800
2	Ohlsson	Blue	56.7	54.7	132.9	133.2
5	Acosta	Gray	59.5	60.4	125.1	123.7

B.

	athlete	team	meet1_400	meet2_400	meet1_800	meet2_800
0	Fazio	Blue	60.7	59.1	135.6	132.0
2	Ohlsson	Blue	56.7	54.7	132.9	133.2
3	Tovcach	Blue	63.0	63.2	121.9	122.8

C.

	athlete	team	meet1_400	meet2_400	meet1_800	meet2_800
2	Ohlsson	Blue	56.7	54.7	132.9	133.2

D.

	meet1_400	meet2_400
0	60.7	59.1
1	62.2	62.6
2	56.7	54.7
3	63.0	63.2
4	61.6	58.5
5	59.5	60.4

Problem 3

Which code produced the DataFrame below?

	athlete	meet3_400	meet3_800
0	Tovcach	62.4	120.9
1	OConnell	59.1	132.3
3	Fazio	58.7	130.1
4	Martell	63.1	134.5
5	Ohlsson	55.2	130.4

A.

```
meet3_df.query('meet3_800.notna()')
```

B.

```
meet3_df.query('meet3_800.isna()')
```

C.

```
meet3_df.filter('meet3_800.notna()')
```

D.

```
meet3_df['meet3_800'].isna()
```

Problem 4

Which code produced the DataFrame below?

	athlete	team	meet1_400	meet2_400	meet1_800	meet2_800	avg_400
0	Fazio	Blue	60.7	59.1	135.6	132.0	59.90
1	Martell	Gray	62.2	62.6	139.5	133.2	62.40
2	Ohlsson	Blue	56.7	54.7	132.9	133.2	55.70
3	Tovcach	Blue	63.0	63.2	121.9	122.8	63.10
4	OConnell	Gray	61.6	58.5	135.0	133.4	60.05
5	Acosta	Gray	59.5	60.4	125.1	123.7	59.95

A.

```
(
    track_df
    .assign(
        avg_400 = lambda x: x[['meet1_400', 'meet2_400']].mean(axis='rows')
    )
)
```

B.

```
(
    track_df
    .groupby(['meet1_400', 'meet2_400'])
    .agg(
        avg_400 = ('400m', 'mean')
    )
)
```

C.

```
(
    track_df(
        avg_400 = mean(['meet1_400', 'meet2_400'])
    )
)
```

D.

```
(
    track_df
    .assign(
        avg_400 = lambda x: x[['meet1_400', 'meet2_400']].mean(axis='columns')
    )
)
```

Problem 5

Which code produced the DataFrame below?

	athlete	team	meet1_400	meet1_800
0	Fazio	Blue	60.7	135.6
1	Martell	Gray	62.2	139.5
2	Ohlsson	Blue	56.7	132.9
3	Tovcach	Blue	63.0	121.9
4	OConnell	Gray	61.6	135.0
5	Acosta	Gray	59.5	125.1

A.

```
(  
    track_df  
    .drop(columns=['meet2_400', 'meet2_800'])  
)
```

B.

```
(  
    track_df  
    ['athlete', 'team', 'meet1_400', 'meet1_800'])  
)
```

C.

```
(  
    track_df  
    .delete(columns=['meet2_400', 'meet2_800'])  
)
```

D.

```
(  
    track_df - track_df['meet2_400'] - track_df['meet2_800']  
)
```

Problem 6

What is the result of the code below?

```
track_df[['meet1_800', 'meet2_800']].mean(axis='rows')
```

A.

	meet1_800	meet2_800	mean
0	135.6	132.0	133.80
1	139.5	133.2	136.35
2	132.9	133.2	133.05
3	121.9	122.8	122.35
4	135.0	133.4	134.20
5	125.1	123.7	124.40

B.

```
0    96.850
1    99.375
2    94.375
3    92.725
4    97.125
5    92.175
dtype: float64
```

C.

```
meet1_800    131.666667
meet2_800    129.716667
dtype: float64
```

D.

```
0    133.80
1    136.35
2    133.05
3    122.35
4    134.20
5    124.40
dtype: float64
```

Problem 7

Which code produced the DataFrame below?

	team	avg_400_meet1	avg_400_meet2
0	Blue	60.133333	59.0
1	Gray	61.100000	60.5

A.

```
(
    track_df
    .groupby(['team'])
    .agg(
        meet1_400 = mean('avg_400_meet1'),
        meet2_400 = mean('avg_400_meet2')
    )
)
```

B.

```
(
    track_df
    .groupby(['team'])
    .agg(
        avg_400_meet1 = ('meet1_400', 'mean'),
        avg_400_meet2 = ('meet2_400', 'mean')
    )
    .reset_index()
)
```

C.

```
(
    track_df['team']
    .agg(
        meet1_400 = ('avg_400_meet1', 'mean'),
        meet2_400 = ('avg_400_meet2', 'mean')
    )
    .reset_index()
)
```

D.

```
(
    track_df['team']
    .assign(
        avg_400_meet1 = ('meet1_400', 'mean'),
        avg_400_meet2 = ('meet2_400', 'mean')
    )
)
```


Problem 8

What is the result of the code below?

```
(
    track_df
    .query('meet2_400 > 60')
    .groupby(['team'])
    .agg(
        n_slow=('athlete', 'count')
    )
    .reset_index()
)
```

A.

	team	n_slow
0	Blue	63.0
1	Gray	121.7

B.

	team	n_slow
0	Blue	2
1	Gray	1

C.

	team	n_slow
0	Blue	1
1	Gray	2

D.

	team	n_slow
0	Blue	3
1	Gray	3

Problem 9

What is the result of the code below?

```
(  
    track_df  
    .groupby(['team'])  
    .filter(  
        lambda x: x['meet1_800'].min() > 125  
    )  
)
```

A.

	athlete	team	meet1_400	meet2_400	meet1_800	meet2_800
0	Fazio	Blue	60.7	59.1	135.6	132.0
2	Ohlsson	Blue	56.7	54.7	132.9	133.2
3	Tovcach	Blue	63.0	63.2	121.9	122.8

B.

	athlete	team	meet1_400	meet2_400	meet1_800	meet2_800
0	Fazio	Blue	60.7	59.1	135.6	132.0
1	Martell	Gray	62.2	62.6	139.5	133.2
2	Ohlsson	Blue	56.7	54.7	132.9	133.2
4	OConnell	Gray	61.6	58.5	135.0	133.4
5	Acosta	Gray	59.5	60.4	125.1	123.7

C.

	athlete	team	meet1_400	meet2_400	meet1_800	meet2_800
3	Tovcach	Blue	63.0	63.2	121.9	122.8

D.

	athlete	team	meet1_400	meet2_400	meet1_800	meet2_800
1	Martell	Gray	62.2	62.6	139.5	133.2
4	OConnell	Gray	61.6	58.5	135.0	133.4
5	Acosta	Gray	59.5	60.4	125.1	123.7

Problem 10

	athlete	team	meet2_400	in_team_rank
0	Fazio	Blue	59.1	2.0
1	Martell	Gray	62.6	3.0
2	Ohlsson	Blue	54.7	1.0
3	Tovcach	Blue	63.2	3.0
4	OConnell	Gray	58.5	1.0
5	Acosta	Gray	60.4	2.0

The following code was used to produce the DataFrame above. Some of the code has been replaced by the words `METHOD_A` and `METHOD_B`. Identify the missing methods.

```
track_df.assign(  
    in_team_rank = (  
        lambda x:  
            x.METHOD_A(['team'])  
            ['meet2_400']  
            .METHOD_B(lambda s: s.rank(ascending=True))  
    )  
)[['athlete', 'team', 'meet2_400', 'in_team_rank']]
```

A.

```
METHOD_A = groupby  
METHOD_B = transform
```

B.

```
METHOD_A = collect  
METHOD_B = query
```

C.

```
METHOD_A = groupby  
METHOD_B = agg
```

D.

```
METHOD_A = group  
METHOD_B = filter
```