CS 425-02, Group 3 - Luis Manzo, Joshua Mathew
Hw2 - due 7/5/2023

Query Statement #1

Output the number of rows in each of your 4 relations (using 4 SELECT statements) in this order: Coach, Person, Player, Team. Call the result column LOADED each time.

SQL Command #1

```
SELECT
(SELECT COUNT(*) FROM Coach) AS LOADED,
(SELECT COUNT(*) FROM Person) AS LOADED,
(SELECT COUNT(*) FROM Player) AS LOADED,
COUNT(*) AS LOADED FROM Team;
```

Query Result/Output #1

	LOADED	LOADED	LOADED	LOADED
▶	33	4981	454	30

Query Statment #2

Output everything in the Team relation, in ascending (increasing) order of TmID.

SQL Command #2

```
SELECT * FROM TEAM ORDER BY TmID;
```

Query Result/Output #2

	TmID	ConfID	Ranking	Playoff	Name	Won	Lost	Games
▶	ATL	EC	3	CS	Atlanta Hawks	44	38	82
	BOS	EC	1	CS	Boston Celtics	56	26	82
	CHA	EC	4		Charlotte Bobcats	34	48	82
	CHI	EC	1	CF	Chicago Bulls	62	20	82
	CLE	EC	5		Cleveland Cavaliers	19	63	82
	DAL	WC	2	NC	Dallas Mavericks	57	25	82
	DEN	WC	2	C1	Denver Nuggets	50	32	82
	DET	EC	4		Detroit Pistons	30	52	82
	GSW	WC	3		Golden State Warriors	36	46	82
	HOU	WC	5		Houston Rockets	43	39	82
	IND	EC	2	C1	Indiana Pacers	37	45	82
	LAC	WC	4		Los Angeles Clippers	32	50	82
	LAL	WC	1	CS	Los Angeles Lakers	57	25	82
	MEM	WC	4	CS	Memphis Grizzlies	46	36	82
	MIA	EC	1	F	Miami Heat	58	24	82
	MIL	EC	3		Milwaukee Bucks	35	47	82
	MIN	WC	5		Minnesota Timberwol...	17	65	82
	NJN	EC	4		New Jersey Nets	24	58	82

Query Statement #3

Show the TmID of teams with Milwaukee in their name, but your query must cater for the fact that people spell this incorrectly – everyone starts with “MIL”, then somewhere later they have a “W” and even later a “K”. Example misspellings are Millwaukee, Milwakee, Milwuakee, Milwaukey, etc.

SQL Command #3

SELECT TmID FROM Team WHERE Name LIKE "MIL%W%K%";

Query Result/Output #3

	TmID
▶	MIL
*	NULL

Query Statement #4

What are the lowest and the highest number of Games (played) in the coach data? Call the first result column LOWEST and the second result column HIGHEST.

SQL Command #4

SELECT Min(Games) AS Lowest, Max(Games) AS Highest FROM Coach;

Query Result/Output #4

	Lowest	Highest
▶	28	82

Query Statement #5

Show the BioID, TmID, Points and Attempts for each player that scored more than 2000 points, in decreasing order of Points. Players with the same number of Points should be shown in alphabetical order of BioID.

SQL Command #5

SELECT BioID, TmID, Points, Attempts FROM Player WHERE Points > 2000 ORDER BY Points DESC, BioID;

Query Result/Output #5

	BioID	TmID	Points	Attempts
▶	duranke01	OKC	2161	675
	jamesle01	MIA	2111	663
	bryanko01	LAL	2078	583
	rosede01	CHI	2026	555
*	NULL	NULL	NULL	NULL

Query Statement #6

Show BioID of persons born in Gabon (GAB) and BioID of persons born in Egypt (EGY), if any.

SQL Command #6

SELECT BioID FROM Person WHERE BirthCountry in ("GAB", "EGY");

Query Result/Output #6

BioID
abdelal01
lasmest01
NULL

Query Statement #7

Show the following for Jamesle01: his Points, the full Name of his team, & the BioID of his team's coach.

SQL Command #7

```
SELECT Player.POINTS, Team.TmID, Coach.BioID FROM Player, Team, Coach
WHERE Player.BioID = "Jamesle01" AND Player.TMID = Team.TmID AND Player.TmID =
Coach.TmID;
```

Query Result/Output #7

POINTS	TmID	BioID
2111	MIA	spoeler01

Query Statement #8

Show the BioID of players whose BirthCountry is in the data, but their BirthCity is not in the data

SQL Command #8

```
SELECT BioID FROM Person
WHERE BirthCountry <> "" AND BirthCountry IS NOT NULL
AND BirthCity = "" AND BirthCity IS NOT NULL;
```

Query Result/Output #8

	BioID
▶	alvaral01
	ardonja01
	bloedch01
	bowlior01
	carlich01
	carlsdo01
	charlke01
	croftbo01
	freyfr01
	grimswo01
	isaacjo01
	minorda01
	robiner01
	vocega01
	zellega01
*	NULL

Query Statement #9
Which coach(es) Won the most games? Give BioID and the number Won.
SQL Command #9
SELECT BioID, Won FROM Coach WHERE Won = (SELECT Max(Won) FROM Coach);
Query Result/Output #9

	BioID	Won
▶	thiboto01	62
*	NULL	NULL

Query Statement #10
What percentage of players have Points scored as zero? Call the result column NONSCORERS.
SQL Command #10
SELECT SUM(Points = 0) / COUNT(*) * 100 AS NONSCORERS FROM Player;
Query Result/Output #10

	NONSCORERS
▶	1.1013

Query Statement #11
How many teams have Lost more games than they have Won? Call the result column LOSERS.
SQL Command #11
SELECT count(*) AS LOSERS FROM team WHERE Won < Lost;
Query Result/Output #11

	LOSERS
▶	14

Query Statement #12
How many teams belong to each ConfID? Call the 2nd column CONFSIZE
SQL Command #12
SELECT ConfID, count(ConfID) as CONFSIZE from team group by ConfID;
Query Result/Output #12

ConfID	CONFSIZE
EC	15
WC	15

Query Statement #13
How many countries do the persons in this data come from? Call the result column NUMLANDS
SQL Command #13
SELECT count(DISTINCT BirthCountry) as NUMLANDS from Person;
Query Result/Output #13

	NUMLANDS
▶	79

Query Statement #14

Which pairs of teams have the exact same record (meaning the same values for Won and the same values for Lost)? Show their Won value and then their Lost value and then the 2 Names, making sure that the 3rd column Name is alphabetically before the 4th column Name so information is not repeated. Call the 3rd column TEAM1 and the 4th column TEAM2.

SQL Command #14

```
SELECT t1.Won, t1.Lost, t1.Name AS TEAM1, t2.Name AS TEAM2 FROM Team AS t1, Team AS t2 WHERE t1.Won = t2.Won AND t1.Lost = t2.Lost AND t1.Name < t2.Name;
```

Query Result/Output #14

Won	Lost	TEAM1	TEAM2
57	25	Dallas Mavericks	Los Angeles Lakers
46	36	Memphis Grizzlies	New Orleans Hornets
24	58	New Jersey Nets	Sacramento Kings

Query Statement #15

For each of the 5 Rankings in ConfID “EC”, show the average number of games Lost by those teams. Call the 2nd column AVLOSSES.

SQL Command #15

```
SELECT Ranking, AVG(Lost) AS AVLOSSES FROM Team
WHERE ConfID = "EC" GROUP BY Ranking ORDER BY Ranking ASC;
```

Query Result/Output #15

Ranking	AVLOSSES
1	23.3333
2	38.3333
3	42.0000
4	52.6667
5	60.6667

Query Statement #16

Give a SQL statement to output “INVALID” if any information in any row (tuple) of any relation has invalid data (e.g., the Games value is not equal to Won value plus Lost

value). If all the data is valid then it should output an empty table. Call the result column ANYPROBLEMS.

SQL Command #16

```
SELECT "INVALID" AS ANYPROBLEMS FROM Person, Coach, Team, Player
WHERE Team.Games <> Team.Won + Team.Lost OR Coach.Games <> Coach.Won +
Coach.Lost OR Person.BirthCountry = "" OR Person.BirthCity = "";
```

Query Result/Output #16