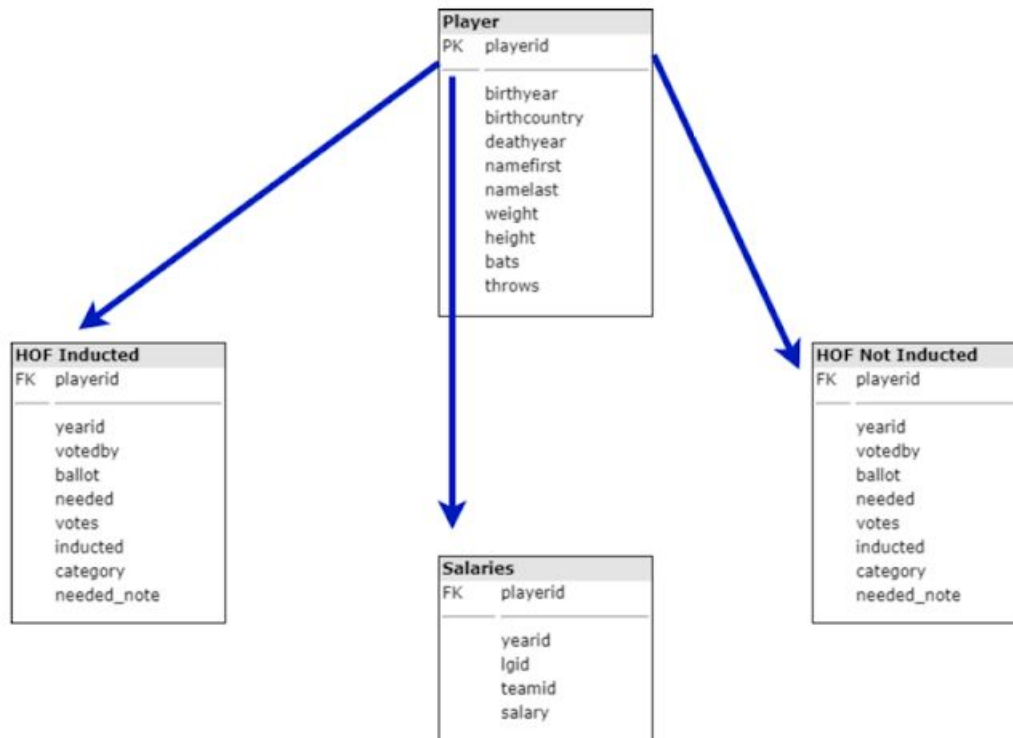


Questions

1. Draw a schema of the relationship between the people, salaries, and hof_inducted tables. There are several online database schema drawers that you can use for free, including draw.io, which we recommend. We discuss the basics of how to use draw.io at the end of this checkpoint.

1. Label the primary and foreign keys. (Note that a field can be a primary key to one table and a foreign key to another.)



2. What are the parent and child tables? Are these one-to-one, one-to-many, or many-to-many relationships?

Parent : Player Child : HOF Inducted One-To-One Relationship
Parent : Player Child : HOF Not Inducted One-To-One Relationship

Parent : Player
Relationship

Child : Salaries

One-To-Many

2. Write a query that returns the namefirst and namelast fields of the people table, along with the inducted field from the hof_inducted table. All rows from the people table should be returned, and NULL values for the fields from hof_inducted should be returned when there is no match found.

```
SELECT namefirst, namelast, inducted  
FROM people LEFT OUTER JOIN hof_inducted  
ON people.playerid = hof_inducted.playerid;
```

3. In 2006, a special Baseball Hall of Fame induction was conducted for players from the negro baseball leagues of the 20th century. In that induction, 17 players were posthumously inducted into the Baseball Hall of Fame. Write a query that returns the first and last names, birth and death dates, and birth countries for these players. Note that the year of induction was 2006, and the value for votedby will be "Negro League."

```
SELECT namefirst, namelast, birthyear, deathyear, birthcountry  
FROM people INNER JOIN hof_inducted  
ON people.playerid = hof_inducted.playerid  
WHERE yearid = 2006 AND votedby = 'Negro League';
```

	namefirst character varying	namelast character varying	birthyear smallint	deathyear smallint	birthcountry character varying	inducted character (1)
1	Ray	Brown	1908	1965	USA	Y
2	Willard	Brown	1915	1996	USA	Y
3	Andy	Cooper	1898	1941	USA	Y
4	Frank	Grant	1865	1937	USA	Y
5	Pete	Hill	1882	1951	USA	Y
6	Biz	Mackey	1897	1965	USA	Y
7	Effa	Manley	1897	1981	USA	Y
8	Jose	Mendez	1885	1928	Cuba	Y
9	Alex	Pompez	1890	1974	USA	Y
10	Cum	Posey	1890	1946	USA	Y
11	Louis	Santop	1889	1942	USA	Y
12	Mule	Suttles	1901	1966	USA	Y
13	Ben	Taylor	1888	1953	USA	Y
14	Cristobal	Toriente	1893	1938	Cuba	Y
15	Sol	White	1868	1955	USA	Y
16	J. L.	Wilkinson	1878	1964	USA	Y
17	Jud	Wilson	1896	1963	USA	Y

4. Write a query that returns the yearid, playerid, teamid, and salary fields from the salaries table, along with the category field from the hof_inducted table. Keep only the records that are in both salaries and hof_inducted. Hint: While a field named yearid is found in both tables, don't JOIN by it. You must, however, explicitly name which field to include.

```
SELECT salaries.yearid, salaries.playerid, salaries.teamid,
salaries.salary, hof_inducted.category
FROM salaries INNER JOIN hof_inducted
ON salaries.playerid = hof_inducted.playerid;
```

	yearid smallint	playerid character varying	teamid character (3)	salary integer	category character varying
1	1985	suttebr01	ATL	1354167	Player
2	1985	murraed02	BAL	1472819	Player
3	1985	ripkeca01	BAL	800000	Player
4	1985	boggswa01	BOS	1000000	Player
5	1985	ricejl01	BOS	779227	Player
6	1985	carewro01	CAL	875000	Player
7	1985	jacksre01	CAL	1058894	Player
8	1985	suttodo01	CAL	750000	Player
9	1985	fiskca01	CHA	685000	Player
10	1985	seaveto01	CHA	1136262	Player
11	1985	eckerde01	CHN	750000	Player
12	1985	sandbry01	CHN	505000	Player
13	1985	perezto01	CIN	225000	Player
14	1985	blylebe01	CLE	650000	Player

5. Write a query that returns the playerid, yearid, teamid, lgid, and salary fields from the salaries table and the inducted field from the hof_inducted table. Keep all records from both tables.

```
SELECT salaries.playerid, salaries.yearid, salaries.teamid,
salaries.lgid, salaries.salary, hof_inducted.inducted
FROM salaries FULL OUTER JOIN hof_inducted
ON salaries.playerid = hof_inducted.playerid;
```

	playerid character varying	yearid smallint	teamid character (3)	lgid character (2)	salary integer	inducted character (1)
1	barkele01	1985	ATL	NL	870000	[null]
2	bedrost01	1985	ATL	NL	550000	[null]
3	benedbr01	1985	ATL	NL	545000	[null]
4	campri01	1985	ATL	NL	633333	[null]
5	ceronri01	1985	ATL	NL	625000	[null]
6	chambch01	1985	ATL	NL	800000	[null]
7	dedmoje01	1985	ATL	NL	150000	[null]
8	forstte01	1985	ATL	NL	483333	[null]
9	garbege01	1985	ATL	NL	772000	[null]
10	harpete01	1985	ATL	NL	250000	[null]
11	hornebo01	1985	ATL	NL	1500000	[null]
12	hubbagl01	1985	ATL	NL	455000	[null]
13	mahleri01	1985	ATL	NL	407500	[null]
14	mcmurcr01	1985	ATL	NL	275000	[null]

6. There are 2 tables, hof_inducted and hof_not_inducted, indicating successful and unsuccessful inductions into the Baseball Hall of Fame, respectively.
1. Combine these 2 tables by all fields. Keep all records.
 2. Get a distinct list of all player IDs for players who have been put up for HOF induction.

1. Use UNION to combine two tables with same field

```
SELECT * FROM hof_inducted
UNION ALL
SELECT * FROM hof_not_inducted;
```

2. Use Common Table Expression (CTE) to query all players put up for HOF from the two table and then get the distinct playerid from combined table

```
WITH all_player AS
(
    SELECT playerid FROM hof_inducted
    UNION ALL
    SELECT playerid FROM hof_not_inducted
)
SELECT DISTINCT playerid FROM all_player;
```

	playerid character varying	
1	shortch02	
2	mcgwima01	
3	tidrodi01	
4	dunnja01	
5	pendite01	
6	hickmji02	
7	bibbyji01	
8	parrila01	
9	mitchke01	
10	morame01	
11	tennefr02	
12	younger01	
13	milledo02	
14	willijo99	

7. Write a query that returns the last name, first name (see people table), and total recorded salaries for all players found in the salaries table.

```
-- Compares the playerid and distinct playerid in people table
SELECT COUNT(playerid), COUNT(DISTINCT playerid)
FROM people;

-- Compares the playerid and distinct playerid in salaries table:
SELECT COUNT(playerid), COUNT(DISTINCT playerid)
FROM salaries;

-- Only 5149 out of 26428 records are unique, hence will add salary
for all years present
SELECT people.namefirst, people.namelast, SUM(salary) AS
total_salary
FROM salaries INNER JOIN people
ON people.playerid = salaries.playerid
GROUP BY people.playerid;
```

	namefirst character varying	namelast character varying	total_salary bigint
1	Julio	Ramirez	500000
2	Carlos	Rodriguez	276000
3	Tony	Saunders	1067500
4	David	Murphy	24535490
5	Shawn	Estes	22455500
6	Bob	Melvin	3165000
7	Randy	Flores	3120000
8	Wade	Davis	24939000
9	Brian	Raabe	160000
10	Marvin	Freeman	6298500
11	Jeff	D'Amico	6402892
12	Joe	Beimel	6436500
13	Brian	Hunter	2416500
14	Milt	Wilcox	700000

8. Write a query that returns all records from the hof_inducted and hof_not_inducted tables that include playerid, yearid, namefirst, and namelast. Hint: Each FROM statement will include a LEFT OUTER JOIN!

```
SELECT hof_inducted.playerid, hof_inducted.yearid,  
people.namefirst, people.namelast  
FROM hof_inducted LEFT OUTER JOIN people  
ON hof_inducted.playerid = people.playerid  
UNION ALL  
SELECT hof_not_inducted.playerid, hof_not_inducted.yearid,  
people.namefirst, people.namelast  
FROM hof_not_inducted LEFT OUTER JOIN people  
ON hof_not_inducted.playerid = people.playerid;
```

	playerid character varying	yearid smallint	namefirst character varying	namelast character varying
1	aaronha01	1982	Hank	Aaron
2	alexape01	1938	Pete	Alexander
3	alomaro01	2011	Roberto	Alomar
4	alstowa01	1983	Walter	Alston
5	andersp01	2000	Sparky	Anderson
6	ansonca01	1939	Cap	Anson
7	aparilu01	1984	Luis	Aparicio
8	applilu01	1964	Luke	Appling
9	ashburi01	1995	Richie	Ashburn
10	averiea01	1975	Earl	Averill
11	bagweje01	2017	Jeff	Bagwell
12	bakerfr01	1955	Home Run	Baker
13	bancrda01	1971	Dave	Bancroft
14	bankser01	1977	Ernie	Banks

9. Return a table including all records from both hof_inducted and hof_not_inducted, and include a new field, namefull, which is formatted as namelast , namefirst (in other words, the last name, followed by a comma, then a space, then the first name). The query should also return the yearid and inducted fields. Include only records since 1980 from both tables. Sort the resulting table by yearid, then inducted so that Y comes before N. Finally, sort by the namefull field, A to Z.

```
SELECT people.namefirst, people.namelast, CONCAT(people.namelast,',',
',people.namefirst) AS namefull, hof_inducted.yearid, hof_inducted.inducted
FROM hof_inducted LEFT OUTER JOIN people
ON hof_inducted.playerid = people.playerid
WHERE hof_inducted.yearid > 1980
UNION ALL
SELECT people.namefirst, people.namelast, CONCAT(people.namelast,',',
',people.namefirst) AS namefull, hof_not_inducted.yearid,
hof_not_inducted.inducted
FROM hof_not_inducted LEFT OUTER JOIN people
ON hof_not_inducted.playerid = people.playerid
WHERE hof_not_inducted.yearid > 1980
ORDER BY yearid, inducted DESC, namefull;
```

	namefirst character varying	namelast character varying	namefull text	yearid smallint	inducted character (1)
1	Rube	Foster	Foster, Rube	1981	Y
2	Bob	Gibson	Gibson, Bob	1981	Y
3	Johnny	Mize	Mize, Johnny	1981	Y
4	Luis	Aparicio	Aparicio, Luis	1981	N
5	Richie	Ashburn	Ashburn, Richie	1981	N
6	Glenn	Beckert	Beckert, Glenn	1981	N
7	Ken	Berry	Berry, Ken	1981	N
8	John	Briggs	Briggs, John	1981	N
9	Gates	Brown	Brown, Gates	1981	N
10	Jim	Bunning	Bunning, Jim	1981	N
11	Lew	Burdette	Burdette, Lew	1981	N
12	Leo	Cardenas	Cardenas, Leo	1981	N
13	Orlando	Cepeda	Cepeda, Orlando	1981	N

10. Write a query that returns the highest annual salary for each teamid, ranked from high to low, along with the corresponding playerid. Bonus! Return namelast and namefirst in the resulting table. (You can find these in the people table.)

```
SELECT DISTINCT s.teamid, p.playerid, s.salary, p.namefirst,  
p.namelast  
FROM salaries AS s  
INNER JOIN  
(SELECT teamid, MAX(salary) AS max_salary  
FROM salaries  
GROUP BY teamid) AS sgroup  
ON s.teamid = sgroup.teamid AND s.salary = sgroup.max_salary  
INNER JOIN player AS p  
ON s.playerid = p.playerid  
ORDER BY s.salaries DESC;
```

	teamid character (3)	playerid character varying	salary integer	namefirst character varying	namelast character varying
1	LAN	kershcl01	33000000	Clayton	Kershaw
2	NYA	rodria01	33000000	Alex	Rodriguez
3	ARI	greinza01	31799030	Zack	Greinke
4	BOS	priceda01	30000000	David	Price
5	DET	cabremi01	28000000	Miguel	Cabrera
6	DET	verlaju01	28000000	Justin	Verlander
7	NYN	cespeyo01	27328046	Yoenis	Cespedes
8	LAA	wellsve01	26187500	Vernon	Wells
9	SEA	hernafe02	25857143	Felix	Hernandez
10	CHN	lestejo01	25000000	Jon	Lester
11	PHI	howarry01	25000000	Ryan	Howard
12	PHI	leeci02	25000000	Cliff	Lee
13	TEX	fieldpr01	24000000	Prince	Fielder
14	TEX	hamiljo03	24000000	Josh	Hamilton
15	MIN	mauerjo01	23000000	Joe	Mauer
...

11. Select birthyear, deathyear, namefirst, and namelast of all the players born since the birth year of Babe Ruth (playerid = ruthba01). Sort the results by birth year from low to high.

```
SELECT birthyear, deathyear, namefirst, namelast
FROM people
WHERE birthyear > ALL
      (SELECT birthyear
       FROM people
       WHERE playerid='ruthba01')
ORDER BY birthyear
```

	birthyear smallint	deathyear smallint	namefirst character varying	namelast character varying
1	1896	1978	Mike	Wilson
2	1896	1983	Chick	Sorrells
3	1896	1963	Merito	Acosta
4	1896	1962	Mutt	Wilson
5	1896	1950	Dick	McCabe
6	1896	1969	Roy	Wilson
7	1896	1969	Ray	Schmandt
8	1896	1972	Rip	Conway
9	1896	1968	Bill	Sherdel
10	1896	1971	Sam	Post
11	1896	1981	Frank	McCrea
12	1896	1979	Bill	Schindler
13	1896	1968	Heinie	Meine

12. Using the people table, write a query that returns namefirst, namelast, and a field called usaborn where. The usaborn field should show the following: if the player's birthcountry is the USA, then the record is 'USA.' Otherwise, it's 'non-USA.' Order the results by 'non-USA' records first.

```

SELECT namefirst, namelast,
CASE
    WHEN birthcountry = 'USA' THEN 'USA'
    WHEN birthcountry <> 'USA' THEN 'non-USA'
END as usaborn
WHERE birthcountry IS NOT NULL
FROM people
ORDER BY usaborn ASC;

```

	namefirst character varying	namelast character varying	usa_stat text
1	Brayan	Pena	non-USA
2	Carlos	Pena	non-USA
3	Elvis	Pena	non-USA
4	Felix	Pena	non-USA
5	Francisco	Pena	non-USA
6	Geronimo	Pena	non-USA
7	Hipolito	Pena	non-USA
8	Jesus	Pena	non-USA
9	Michael	Campbell	non-USA
10	Jose	Pena	non-USA
11	Juan	Pena	non-USA
12	Orlando	Pena	non-USA
13	Ramon	Pena	non-USA

- Calculate the average height for players throwing with their right hand versus their left hand. Name these fields right_height and left_height, respectively.

```

SELECT
ROUND(AVG(CASE WHEN throws = 'R' THEN height END),2) AS right_height,
ROUND(AVG(CASE WHEN throws = 'L' THEN height END),2) AS left_height
FROM people;

```

	right_height numeric	left_height numeric
1	72.32	72.55

14. Get the average of each team's maximum player salary since 2010. Hint: WHERE will go inside your CTE.

```

WITH highest_sal AS
(
    SELECT yearid, teamid, MAX(salary) AS high_sal
    FROM salaries
    WHERE yearid > 2010
    GROUP BY yearid, teamid
)
SELECT teamid, ROUND(AVG(high_sal),0) AS avghigh_sal
FROM highest_sal
GROUP BY teamid;

```

	teamid character (3)	avghigh_sal numeric
1	HOU	13133333
2	PIT	12930556
3	SLN	17897041
4	ARI	13066505
5	OAK	9811111
6	TOR	15666667
7	TEX	20058325
8	ATL	13459386
9	LAN	23015802
10	CHA	15942879
11	CIN	14922609
12	NYA	27020833
13	KCA	10583333