SQL1

SELECT year, AVG(salary)/12 AS "average salary"

FROM salaries

GROUP BY year;

SQL2

SELECT year, salary

FROM salaries

WHERE player\_id = (SELECT id FROM players WHERE last\_name LIKE '%Ripken%')

ORDER BY year DESC;

SQL3

find Ken Griffey Jr.’s home run history.

* Sort by year in descending order.
* Note that there may be two players with the name “Ken Griffey.” *This* Ken Griffey was born in 1969.
* Your query should return a table with two columns, one for year and one for home runs.

SELECT year, HR

FROM performances

WHERE player\_id = (SELECT id FROM players WHERE last\_name LIKE '%Griffey%' AND birth\_year = 1969)

ORDER BY year DESC;

SQL4

the 50 players paid the least in 2001.

* Sort players by salary, lowest to highest.
* If two players have the same salary, sort alphabetically by first name and then by last name.
* If two players have the same first and last name, sort by player ID.
* Your query should return three columns, one for players’ first names, one for their last names, and one for their salaries.

SELECT players.first\_name, players.last\_name, salaries.salary

FROM salaries

JOIN players ON salaries.player\_id = players.id

WHERE salaries.year = 2001

ORDER BY salaries.salary ASC, players.first\_name ASC, players.last\_name ASC, players.id ASC

LIMIT 50;

SQL5

find all teams that [Satchel Paige](https://en.wikipedia.org/wiki/Satchel_Paige) played for.

* Your query should return a table with a single column, one for the name of the teams.

SELECT DISTINCT teams.name

FROM teams

JOIN performances ON teams.id = performances.team\_id

JOIN players ON performances.player\_id = players.id

WHERE players.last\_name LIKE '%Paige%'

ORDER BY teams.name ASC;

SQL6

return the top 5 teams, sorted by the total number of hits by players in 2001.

* Call the column representing total hits by players in 2001 “total hits”.
* Sort by total hits, highest to lowest.
* Your query should return two columns, one for the teams’ names and one for their total hits in 2001.

***Attempt***

**SELECT DISTINCT teams.name, performances.H**

**FROM teams, performances**

**JOIN** performances ON teams.id = performances.team\_id

JOIN players ON performances.player\_id = players.id

WHERE performances.year = 2001

ORDER BY p**erformances.H** ASC;

***Answer***

**SELECT teams.name, SUM(performances.H) AS "total hits"**

**FROM teams**

**JOIN performances ON teams.id = performances.team\_id**

**WHERE performances.year = 2001**

**GROUP BY teams.name**

**ORDER BY "total hits" DESC**

**LIMIT 5;**

[**7.sql**](https://cs50.harvard.edu/sql/2023/psets/1/moneyball/#7sql)

find the name of the player who’s been paid the highest salary, of all time, in Major League Baseball.

* Your query should return a table with two columns, one for the player’s first name and one for their last name.

SELECT players.first\_name, players.last\_name

FROM players

JOIN salaries ON players.id = salaries.player\_id

ORDER BY salaries.salary DESC

LIMIT 1;

[**8.sql**](https://cs50.harvard.edu/sql/2023/psets/1/moneyball/#8sql)

***Attempt***

find the 2001 salary of the player who hit the most home runs in 2001.

* Your query should return a table with one column, the salary of the player.

SELECT salaries.salary

FROM salaries

JOIN salaries ON performances.id = salaries.player\_id

ORDER BY performances.H DESC

WHERE performances.year = 2001

LIMIT 1;

***Answer***

SELECT salaries.salary

FROM performances

JOIN salaries ON performances.player\_id = salaries.player\_id

WHERE performances.year = 2001 AND salaries.year = 2001

ORDER BY performances.HR DESC

LIMIT 1;

[**9.sql**](https://cs50.harvard.edu/sql/2023/psets/1/moneyball/#9sql)

find the 5 lowest paying teams (by average salary) in 2001.

* Round the average salary column to two decimal places and call it “average salary”.
* Sort the teams by average salary, least to greatest.
* Your query should return a table with two columns, one for the teams’ names and one for their average salary.

***Attempt***

SELECT teams.id, salaries.salary AS average salary

FROM salaries

JOIN teams.id on salaries.team\_id

WHERE salary is average?

***Answer***

SELECT teams.name, ROUND(AVG(salaries.salary), 2) AS "average salary"

FROM salaries

JOIN teams ON salaries.team\_id = teams.id

WHERE salaries.year = 2001

GROUP BY teams.name

ORDER BY "average salary" ASC

LIMIT 5;

[**10.sql**](https://cs50.harvard.edu/sql/2023/psets/1/moneyball/#10sql)

the table should include:

* All player’s first names
* All player’s last names
* All player’s salaries
* All player’s home runs
* The year in which the player was paid that salary *and* hit those home runs

In 10.sql, write a query to return just such a table.

* Your query should return a table with five columns, per the above.
* Order the results, first and foremost, by player’s IDs (least to greatest).
* Order rows about the same player by year, in descending order.
* Consider a corner case: suppose a player has multiple salaries or performances for a given year. Order them first by number of home runs, in descending order, followed by salary, in descending order.
* Be careful to ensure that, for a single row, the salary’s year and the performance’s year match.

***Attempt***

SELECT first\_name, last\_name, ASC(salary, 1), HR, DESC(year)

JOIN ON players.id, performances.player\_id, salaries.player\_id

WHERE salary > 1 IN year

ORDER BY DESC(HR, salary)

***Answer***

SELECT players.first\_name, players.last\_name, salaries.salary, performances.HR, salaries.year

FROM players

JOIN salaries ON players.id = salaries.player\_id

JOIN performances ON players.id = performances.player\_id AND salaries.year = performances.year

ORDER BY players.id ASC, salaries.year DESC, performances.HR DESC, salaries.salary DESC;

[**11.sql**](https://cs50.harvard.edu/sql/2023/psets/1/moneyball/#11sql)

find the 10 least expensive players *per hit* in 2001.

* Your query should return a table with three columns, one for the players’ first names, one of their last names, and one called “dollars per hit”.
* You can calculate the “dollars per hit” column by dividing a player’s 2001 salary by the number of hits they made in 2001. Recall you can use AS to rename a column.
* Dividing a salary by 0 hits will result in a NULL value. Avoid the issue by filtering out players with 0 hits.
* Sort the table by the “dollars per hit” column, least to most expensive. If two players have the same “dollars per hit”, order by first name, followed by last name, in alphabetical order.
* As in 10.sql, ensure that the salary’s year and the performance’s year match.
* You may assume, for simplicity, that a player will only have one salary and one performance in 2001.

[**12.sql**](https://cs50.harvard.edu/sql/2023/psets/1/moneyball/#12sql)

find the players among the 10 least expensive players per hit **and** among the 10 least expensive players per RBI in 2001.

* Your query should return a table with two columns, one for the players’ first names and one of their last names.
* You can calculate a player’s salary per RBI by dividing their 2001 salary by their number of RBIs in 2001.
* You may assume, for simplicity, that a player will only have one salary and one performance in 2001.
* Order your results by player ID, least to greatest (or alphabetically by last name, as both are the same in this case!).
* Keep in mind the lessons you’ve learned in 10.sql and 11.sql!