

```
USE Baseball;
select * From dbo.Batting
select * From dbo.People
Select * From dbo.Pitching
select * From dbo.Teams
select * From dbo.TeamsFranchises
```

```
SELECT Top (3)* FROM dbo.Batting
SELECT Top (3)* From dbo.People
SELECT Top (3)* FROM dbo.Teams
SELECT Top (3)* FROM dbo.Pitching
select Top (3)* From dbo.TeamsFranchises
```

--1. Which active player had the most runs batted in ("RBI" from the Batting table) from 2015-2018? ↗

--First, make sure you have the active players' information. Assuming you have a way to determine if a player was active between 2015 and 2018 (e.g., through the debut and finalGame fields from the People table), you can query the Batting table to find the player with the most RBI. ↗

```
SELECT TOP 1 playerID, SUM(CAST(RBI AS INT)) AS Total_RBI
FROM dbo.Batting
WHERE yearID BETWEEN 2015 AND 2018
GROUP BY playerID
ORDER BY Total_RBI DESC;
```

--2. How many double plays did Albert Pujols ground into ("GIDP" from the Batting table) in 2016? ↗

-- GIDP (Ground Into Double Play).

```
SELECT Top (2) * FROM dbo.Batting

SELECT SUM(CAST(GIDP AS INT)) AS Total_GIDP
FROM dbo.Batting
WHERE playerID = 'pujolal01'
AND yearID = 2016;
```

-- if we know the name and or Player ID can execute the syntax as below

```
SELECT * FROM dbo.People

SELECT p.playerID, p.nameFirst, p.nameLast, SUM(CAST(b.GIDP AS INT)) AS Total_GIDP
FROM dbo.Batting b
JOIN dbo.People p ON b.playerID = p.playerID
WHERE b.playerID = 'pujolal01'
AND p.nameFirst = 'Albert'
AND b.yearID = 2016
GROUP BY p.playerID, p.nameFirst, p.nameLast;
```

--1. Find Active Players Who Played at Least 50 Games
--To identify players who have played at least 50 games and are still active.

```
SELECT DISTINCT p.playerID, p.nameFirst, p.nameLast, p.weight, pthrows, p.bats,  
                p.birthYear, p.birthMonth, p.birthDay, p.nameGiven  
FROM dbo.Batting b  
JOIN dbo.People p ON b.playerID = p.playerID  
WHERE b.G >= 50  
      AND p.finalGame IS NULL -- Player is active  
      AND b.playerID = 'TEST123';
```

```
SELECT p.playerID, p.nameFirst, p.nameLast, COUNT(*) AS NumberOfRecords  
FROM dbo.Batting b  
JOIN dbo.People p ON b.playerID = p.playerID  
WHERE b.G >= 50  
      AND p.finalGame IS NULL -- Player is active  
GROUP BY p.playerID, p.nameFirst, p.nameLast  
ORDER BY NumberOfRecords DESC;
```

```
SELECT b.playerID, b.G, p.finalGame  
FROM dbo.Batting b  
JOIN dbo.People p ON b.playerID = p.playerID  
WHERE b.G >= 50;
```

```
SELECT b.playerID, b.G, p.finalGame  
FROM dbo.Batting b  
JOIN dbo.People p ON b.playerID = p.playerID  
WHERE b.G >= 50  
      AND p.finalGame IS NULL; -- Ensures that the player is still active  
SELECT DISTINCT p.finalGame  
FROM dbo.People p  
WHERE p.finalGame IS NULL;
```

```
SELECT b.playerID, b.G, p.finalGame  
FROM dbo.Batting b  
JOIN dbo.People p ON b.playerID = p.playerID  
WHERE p.finalGame IS NULL;
```

```
SELECT b.playerID, b.G, p.finalGame  
FROM dbo.Batting b  
JOIN dbo.People p ON b.playerID = p.playerID  
WHERE b.G = 50  
      AND p.finalGame IS NULL;
```

```
SELECT p.finalGame, COUNT(*) As 'Nu_of_Records'  
FROM dbo.People p  
GROUP BY p.finalGame
```

```
ORDER BY COUNT(*) DESC;
```

```
SELECT b.playerID, b.G, p.finalGame
FROM dbo.Batting b
JOIN dbo.People p ON b.playerID = p.playerID
WHERE b.G >= 50
      AND (p.finalGame IS NULL OR p.finalGame = '');
```

```
SELECT b.playerID, b.G, p.finalGame
FROM dbo.Batting b
JOIN dbo.People p ON b.playerID = p.playerID
WHERE b.G >= 50
      AND p.finalGame = '';
```

```
-- Players with games >= 50
SELECT DISTINCT b.playerID, b.G
FROM dbo.Batting b
WHERE b.G >= 50;
```

```
-- Players with finalGame = '' or NULL
SELECT DISTINCT p.playerID
FROM dbo.People p
WHERE p.finalGame IS NULL OR p.finalGame = '';
```

```
SELECT DISTINCT p.finalGame
FROM dbo.People p
WHERE p.finalGame IS NULL;
```

```
SELECT p.playerID, p.nameFirst, p.nameLast
FROM dbo.Batting b
JOIN dbo.People p ON b.playerID = p.playerID
WHERE b.G >= 50
      AND p.finalGame IS NULL;
```

```
SELECT DISTINCT b.playerID, b.G
FROM dbo.Batting b
WHERE b.G >= 50;
```

```
SELECT DISTINCT p.playerID, p.finalGame
FROM dbo.People p
WHERE p.finalGame IS NULL;
SELECT p.playerID, p.nameFirst, p.nameLast, COUNT(*) AS NumberOfRecords
FROM dbo.Batting b
JOIN dbo.People p ON b.playerID = p.playerID
WHERE b.G >= 50
      AND p.finalGame IS NULL
GROUP BY p.playerID, p.nameFirst, p.nameLast;
```