

IF2240 - Basis Data

Praktikum 2

Operasi SQL Lanjut

14 Maret 2024

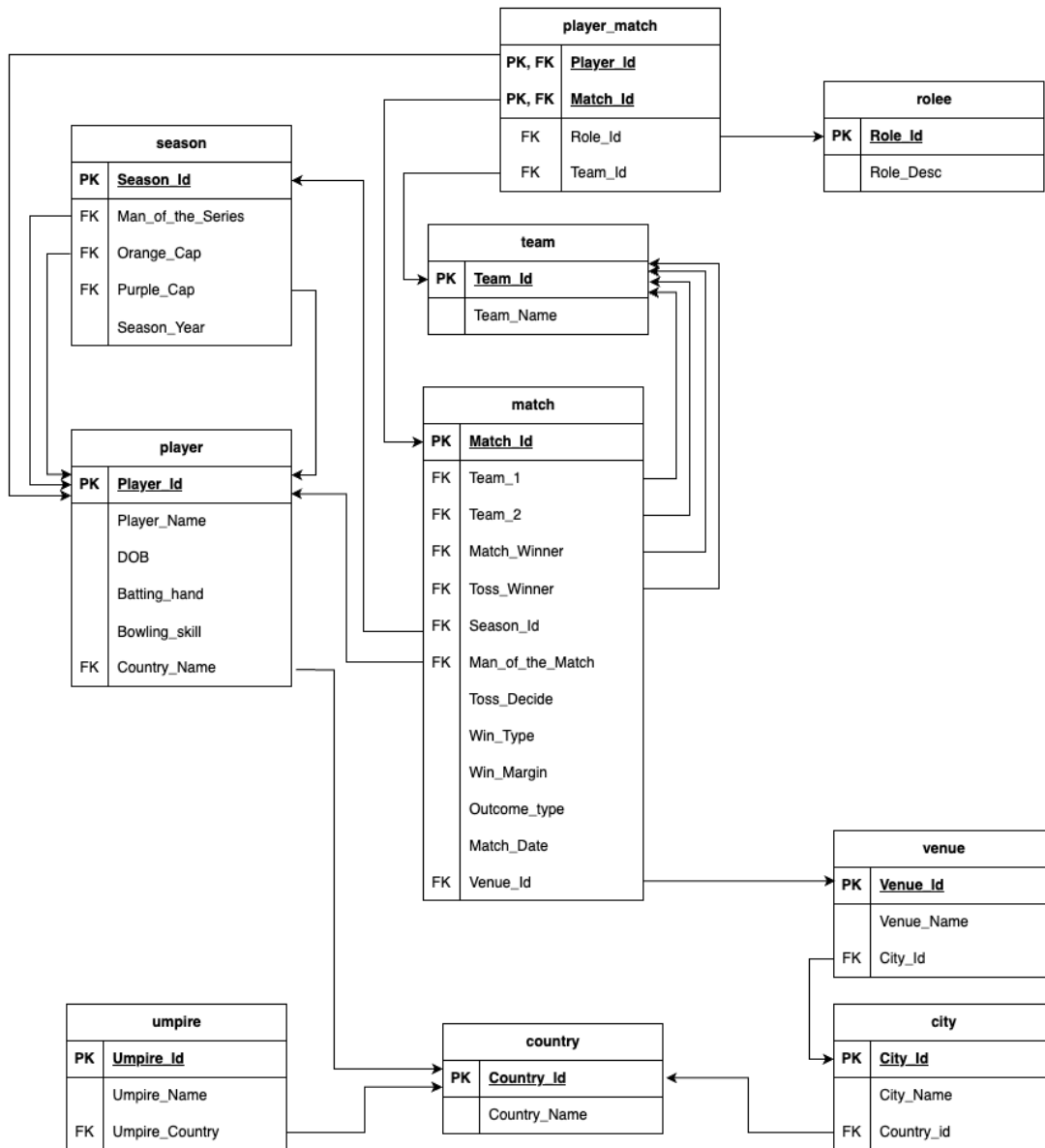


Erdianti Wiga Putri Andini

13522053

**Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung
2024**

Indian Premier League



Legenda

No	Nama Tabel	Keterangan
1	player_match	Keterlibatan seorang pemain pada sebuah pertandingan
2	rolee	Deskripsi peran pemain
3	season	Musim pertandingan
4	team	Tim
5	player	Pemain
6	match	Pertandingan
7	venue	Tempat diadakannya pertandingan
8	city	Kota
9	country	Negara
10	umpire	Wasit

Petunjuk

1. Praktikum ini wajib dikerjakan di dalam Laboratorium Teknik Informatika ITB.
2. Tidak ada restriksi penggunaan keyword pada praktikum ini. Anda dapat menggunakan keyword apapun untuk menjawab soal praktikum.
3. Aturan No 2 tidak berlaku untuk *join* relasi. Anda harus menggunakan *keyword join* yang bersesuaian, tidak boleh melakukan *join* secara manual (seperti pada praktikum 1).

1. Pemain Sepuh

Tampilkan ID pemain yang pernah menjadi *man of the match* pada pertandingan yang diadakan pada bulan Maret sekaligus *man of the series* pada musim pertandingan yang sama. Berikan judul kolom 'Pemain Sepuh' pada hasil pencarian. Pastikan jawaban tidak duplikat.

[HINT] Gunakan backtick (`) untuk nama tabel yang sama dengan salah satu reserved keyword SQL. Gunakan fungsi dasar SQL untuk mengekstrak bagian dari sebuah tanggal seperti MONTH(), YEAR(), WEEK().

Query	<pre>SELECT Player_id as 'Pemain Sepuh' FROM player p NATURAL JOIN `match` m NATURAL JOIN season s WHERE p.Player_id = m.Man_of_the_Match AND p.Player_id = s.Man_of_the_series AND s.Season_id = m.Season_id;</pre>
SS Query dan Hasil	

```

MariaDB [prak2]> SELECT Player_id as 'Pemain Sepuh'
  -> FROM player p NATURAL JOIN `match` m NATURAL JOIN season s
  -> WHERE p.Player_id = m.Man_of_the_Match AND p.Player_id = s.Man_of_the_series AND s
son_id = m.Season_id;
+-----+
| Pemain Sepuh |
+-----+
|          32 |
|          32 |
|          32 |
|          32 |
|          53 |
|          53 |
|         133 |
|         133 |
|         133 |
|         133 |
|         162 |
|         162 |
|         162 |
|         162 |
|         162 |
|         315 |
|         315 |
|         315 |
|          32 |
|         305 |
|         305 |
|         305 |
|         305 |
|         334 |
|         334 |
|         334 |
|           8 |
|           8 |
|           8 |
|           8 |
|           8 |
+-----+
32 rows in set (0,002 sec)

```

2. Riwayat Jumlah Pemain

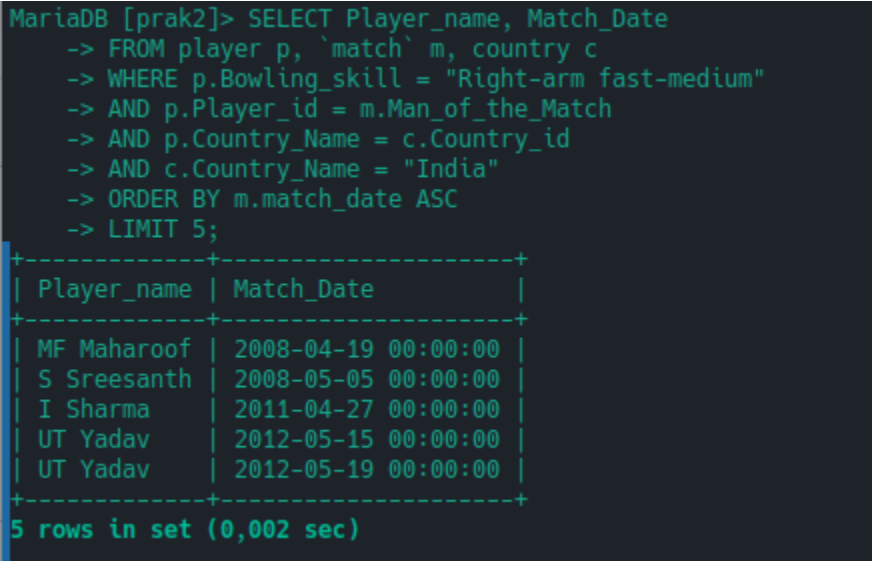
Pak Jay ingin melihat jumlah pemain yang pernah bertanding dalam tim. Tampilkan **nama tim** serta **jumlah pemainnya** (nama kolom dibebaskan) lalu urutkan berdasarkan jumlah pemain secara menurun. Pastikan pemain tidak duplikat.

Query	<pre> SELECT Team_name, COUNT(DISTINCT Player_id) AS countPlayerTeam FROM team t NATURAL JOIN player_match pm GROUP BY Team_name ORDER BY countPlayerTeam DESC; </pre>																												
SS Query dan Hasil																													
<pre> MariaDB [prak2]> SELECT Team_name, COUNT(DISTINCT Player_id) AS countPlayerTeam -> FROM team t NATURAL JOIN player_match pm -> GROUP BY Team_name -> ORDER BY countPlayerTeam DESC; </pre> <table> <thead> <tr> <th>Team_name</th><th>countPlayerTeam</th></tr> </thead> <tbody> <tr><td>Royal Challengers Bangalore</td><td>100</td></tr> <tr><td>Mumbai Indians</td><td>89</td></tr> <tr><td>Delhi Daredevils</td><td>88</td></tr> <tr><td>Kings XI Punjab</td><td>87</td></tr> <tr><td>Rajasthan Royals</td><td>86</td></tr> <tr><td>Kolkata Knight Riders</td><td>83</td></tr> <tr><td>Deccan Chargers</td><td>62</td></tr> <tr><td>Chennai Super Kings</td><td>58</td></tr> <tr><td>Pune Warriors</td><td>46</td></tr> <tr><td>Sunrisers Hyderabad</td><td>43</td></tr> <tr><td>Rising Pune Supergiants</td><td>23</td></tr> <tr><td>Kochi Tuskers Kerala</td><td>20</td></tr> <tr><td>Gujarat Lions</td><td>19</td></tr> </tbody> </table> <pre> 13 rows in set (0,007 sec) </pre>		Team_name	countPlayerTeam	Royal Challengers Bangalore	100	Mumbai Indians	89	Delhi Daredevils	88	Kings XI Punjab	87	Rajasthan Royals	86	Kolkata Knight Riders	83	Deccan Chargers	62	Chennai Super Kings	58	Pune Warriors	46	Sunrisers Hyderabad	43	Rising Pune Supergiants	23	Kochi Tuskers Kerala	20	Gujarat Lions	19
Team_name	countPlayerTeam																												
Royal Challengers Bangalore	100																												
Mumbai Indians	89																												
Delhi Daredevils	88																												
Kings XI Punjab	87																												
Rajasthan Royals	86																												
Kolkata Knight Riders	83																												
Deccan Chargers	62																												
Chennai Super Kings	58																												
Pune Warriors	46																												
Sunrisers Hyderabad	43																												
Rising Pune Supergiants	23																												
Kochi Tuskers Kerala	20																												
Gujarat Lions	19																												

3. Pemain India OP

Tampilkan 5 baris nama pemain dan tanggal pertandingan terurut membesar untuk pemain yang menjadi *man_of_the_match* dan memiliki *bowling_skill* "Right-arm fast-medium" yang berasal dari India. (Note: Pemain bisa menjadi *man_of_the_match* lebih dari sekali pada *match* yang berbeda)

[HINT] Gunakan backtick (`) untuk nama tabel yang sama dengan salah satu reserved keyword SQL. Gunakan LIMIT untuk membatasi jumlah rows yang ditampilkan

Query	<pre> SELECT Player_name, Match_Date FROM player p, `match` m, country c WHERE p.Bowling_skill = "Right-arm fast-medium" AND p.Player_id = m.Man_of_the_Match AND p.Country_Name = c.Country_id AND c.Country_Name = "India" ORDER BY m.match_date ASC LIMIT 5; </pre>
SS Query dan Hasil	
 <pre> MariaDB [prak2]> SELECT Player_name, Match_Date -> FROM player p, `match` m, country c -> WHERE p.Bowling_skill = "Right-arm fast-medium" -> AND p.Player_id = m.Man_of_the_Match -> AND p.Country_Name = c.Country_id -> AND c.Country_Name = "India" -> ORDER BY m.match_date ASC -> LIMIT 5; +-----+-----+ Player_name Match_Date +-----+-----+ MF Maharoor 2008-04-19 00:00:00 S Sreesanth 2008-05-05 00:00:00 I Sharma 2011-04-27 00:00:00 UT Yadav 2012-05-15 00:00:00 UT Yadav 2012-05-19 00:00:00 +-----+-----+ 5 rows in set (0,002 sec) </pre>	

4. Terhebat pada Musimnya

Untuk setiap season, terdapat pemain-pemain hebat yang meraih penghargaan seperti *Man of the Series*, *Orange Cap*, maupun *Purple Cap*. Tampilkan *player_id* dan *player_name* untuk seluruh pemain yang pernah mendapatkan penghargaan *Man of the Series* tetapi tidak pernah mendapatkan penghargaan *Orange Cap*! Terlebih, tampilkan juga *season_id* dimana pemain mendapatkan penghargaan *Man of the Series* tersebut! Anda harus menggunakan OUTER JOIN pada soal ini.

Query	<pre> SELECT DISTINCT Player_id, Player_Name, Season_id, Orange_Cap, Man_of_the_series FROM player p NATURAL JOIN season s WHERE p.Player_id = s.Man_of_the_series AND p.Player_id IN (SELECT Player_id FROM player p1 LEFT OUTER JOIN season s1 ON p1.Player_id = s1.Orange_Cap); </pre>
-------	---

SS Query dan Hasil

```

MariaDB [prak2]> SELECT DISTINCT Player_id, Player_Name, Season_id, Orange_Cap, Man_of_the_se
ries
-> FROM player p NATURAL JOIN season s
-> WHERE p.Player_id = s.Man_of_the_series
-> AND p.Player_id IN
-> (SELECT Player_id
-> FROM player p1 LEFT OUTER JOIN season s1
-> ON p1.Player_id = s1.Orange_Cap);
+-----+-----+-----+-----+-----+
| Player_id | Player_Name | Season_id | Orange_Cap | Man_of_the_series |
+-----+-----+-----+-----+-----+
| 32 | SR Watson | 1 | 100 | 32 |
| 53 | AC Gilchrist | 2 | 18 | 53 |
| 133 | SR Tendulkar | 3 | 133 | 133 |
| 162 | CH Gayle | 4 | 162 | 162 |
| 315 | SP Narine | 5 | 162 | 315 |
| 32 | SR Watson | 6 | 19 | 32 |
| 305 | GJ Maxwell | 7 | 46 | 305 |
| 334 | AD Russell | 8 | 187 | 334 |
| 8 | V Kohli | 9 | 8 | 8 |
+-----+-----+-----+-----+-----+
9 rows in set (0,002 sec)

```

5. Pemain Beban

Tampilkan 5 nama pemain yang tidak memiliki *bowling skill* (ditandai dengan *bowling skill* bernilai NULL) dengan jumlah pertandingan kalah terbanyak. Berikan pula jumlah pertandingan kalah tersebut

[HINT] Gunakan backtick (`) untuk nama tabel yang sama dengan salah satu reserved keyword SQL. Gunakan keyword LIMIT untuk membatasi jumlah baris keluaran.

Query	SELECT DISTINCT Player_Name
-------	-----------------------------

```

FROM player p NATURAL JOIN player_match pm
WHERE p.Bowling_skill IS NULL
AND p.Player_id = pm.Player_id
AND pm.Player_id IN
(SELECT Player_id
FROM player_match pm1 NATURAL JOIN `match` m1
WHERE pm1.match_id = m1.match_id
GROUP BY Player_id
HAVING COUNT(m1.match_winner) <
ALL(SELECT COUNT(match_winner)
FROM `match`))
LIMIT 5;

```

SS Query dan Hasil

```

MariaDB [prak2]> SELECT DISTINCT Player_Name
-> FROM player p NATURAL JOIN player_match pm
-> WHERE p.Bowling_skill IS NULL
-> AND p.Player_id = pm.Player_id
-> AND pm.Player_id IN
-> (SELECT Player_id
-> FROM player_match pm1 NATURAL JOIN `match` m1
-> WHERE pm1.match_id = m1.match_id
-> GROUP BY Player_id
-> HAVING COUNT(m1.match_winner) <
-> ALL(SELECT COUNT(match_winner)
-> FROM `match`))
-> LIMIT 5;
+-----+
| Player_Name |
+-----+
| PA Patel    |
| M Rawat     |
| L Ronchi    |
| PR Shah     |
| WP Saha     |
+-----+
5 rows in set (0,009 sec)

```

6. Venue Terpopuler

Anak pak Jay ingin mengetahui **5 venue** terpopuler di Indian Premier League, yaitu yang paling sering menyelenggarakan pertandingan. Tampilkan **nama venue**, banyaknya pertandingan yang diselenggara (kolom diberi nama

total_match), dan **nama negara** tempat **venue**. Urutkan berdasarkan **total_match**.

[HINT] Gunakan backtick (`) untuk nama tabel yang sama dengan salah satu reserved keyword SQL. Gunakan keyword LIMIT untuk membatasi jumlah baris keluaran.

Query	<pre>SELECT Venue_Name, COUNT(match_id) AS total_match, Country_Name FROM venue v NATURAL JOIN `match` m NATURAL JOIN city c NATURAL JOIN country ct WHERE v.venue_id = m.venue_id AND v.city_id = c.city_id AND c.country_id = ct.country_id GROUP BY venue_id ORDER BY total_match DESC LIMIT 5;</pre>
SS Query dan Hasil	
<pre>MariaDB [prak2]> SELECT Venue_Name, COUNT(match_id) AS total_match, Country_Name -> FROM venue v NATURAL JOIN `match` m NATURAL JOIN city c NATURAL JOIN country ct -> WHERE v.venue_id = m.venue_id -> AND v.city_id = c.city_id -> AND c.country_id = ct.country_id -> GROUP BY venue_id -> ORDER BY total_match DESC -> LIMIT 5; +-----+-----+-----+ Venue_Name total_match Country_Name +-----+-----+-----+ M Chinnaswamy Stadium 58 India Eden Gardens 54 India Feroz Shah Kotla 53 India Wankhede Stadium 49 India MA Chidambaram Stadium, Chepauk 48 India +-----+-----+-----+ 5 rows in set (0,004 sec)</pre>	

7. MVP Royal Challengers Bangalore

Tampilkan nama pemain dan *batting_hand* pemain untuk pemain yang pernah meraih *man_of_the_match* lebih dari 5 kali untuk tim “Royal Challengers Bangalore”

[HINT] Gunakan CONCAT() untuk melakukan konkatenasi string.

Query	<pre>SELECT Player_Name, Batting_hand FROM player p NATURAL JOIN `match` m NATURAL JOIN player_match pm NATURAL JOIN team t WHERE p.Player_id = pm.Player_id AND pm.team_id = t.team_id AND t.team_name = "Royal Challengers Bangalore" AND p.Player_id = m.Man_of_the_Match GROUP BY Player_id HAVING COUNT(Man_of_the_Match) > 5;</pre>
SS Query dan Hasil	
<pre>MariaDB [prak2]> SELECT Player_Name, Batting_hand -> FROM player p NATURAL JOIN `match` m NATURAL JOIN player_match pm NATURAL JOIN team t -> WHERE p.Player_id = pm.Player_id -> AND pm.team_id = t.team_id -> AND t.team_name = "Royal Challengers Bangalore" -> AND p.Player_id = m.Man_of_the_Match -> GROUP BY Player_id -> HAVING COUNT(Man_of_the_Match) > 5; +-----+-----+ Player_Name Batting_hand +-----+-----+ V Kohli Right-hand bat AB de Villiers Right-hand bat CH Gayle Left-hand bat +-----+-----+ 3 rows in set (0,006 sec)</pre>	

8. Pemain Terbaik dari yang Terburuk

Seorang *database administrator*, Julala, menganggap bahwa tim dengan *skill* yang buruk adalah tim yang memiliki akumulasi *win_margin* kurang dari 100. Maka dari itu, ia beranggapan bahwa jika tim yang termasuk ke dalam golongan

tersebut memenangkan pertandingan, maka hal tersebut akan disebut *anomali*. Tampilkan atribut pemain yakni ***player_id***, ***player_name***, dan jumlah pemain tersebut menjadi ***man_of_the_match*** (tampilkan sebagai ***count_motm***) untuk pemain yang menjadi *man_of_the_match* dari *match* yang dianggap *anomali*. Urutkan hasil berdasarkan *player_id* secara *ascending*. Anda diwajibkan menerapkan *subquery* untuk persoalan ini. *Keyword subquery* yang diperbolehkan adalah IN (tidak boleh menggunakan *keyword subquery* lain).

[HINT] Gunakan AS untuk melakukan penamaan ulang atribut yang ditampilkan.

Query	<pre> SELECT player_id, Player_Name, COUNT(Man_of_the_Match) AS count_motm FROM player p NATURAL JOIN player_match pm NATURAL JOIN team t NATURAL JOIN `match` m WHERE p.player_id = pm.player_id AND pm.team_id = t.team_id AND t.team_id = m.match_winner AND player_id IN (SELECT player_id FROM player p1 NATURAL JOIN `match` m1 WHERE m1.win_margin < 100) ORDER BY player_id ASC; </pre>
SS Query dan Hasil	
<pre> MariaDB [prak2]> SELECT player_id, Player_Name, COUNT(Man_of_the_Match) AS count_motm -> FROM player p NATURAL JOIN player_match pm NATURAL JOIN team t NATURAL JOIN `match` m -> WHERE p.player_id = pm.player_id -> AND pm.team_id = t.team_id -> AND t.team_id = m.match_winner -> AND player_id IN -> (SELECT player_id -> FROM player p1 NATURAL JOIN `match` m1 -> WHERE m1.win_margin < 100) -> ORDER BY player_id ASC; +-----+-----+-----+ player_id Player_Name count_motm +-----+-----+-----+ 1 SC Ganguly 6276 +-----+-----+-----+ 1 row in set (0,026 sec) </pre>	

9. MyHand Menyerang Tim (blm)

Setelah puas melakukan phising pada pemilik waralaba, MyHand ingin menemukan nama dari tim yang pernah menang dalam pertandingan pada tahun 2009 dan menang *toss* pada tahun 2016 untuk dikirimkan sebuah paket. Ia juga ingin memastikan bahwa tim yang terpilih, pada tahun 2009, *man of the match*-nya harus berasal dari negara yang diawali huruf A. Bantulah MyHand menemukan nama-nama tim yang sesuai dengan keperluannya tanpa duplikat! Anda diwajibkan menerapkan *subquery* untuk persoalan ini. *Keyword subquery* yang diperbolehkan adalah EXISTS (tidak boleh menggunakan *keyword subquery* lain)

HINT: Ada beberapa nama tabel yang merupakan reserved keywords SQL. Gunakan backtick (`) untuk membungkus nama tabel tersebut jika digunakan. Gunakan fungsi bawaan SQL untuk mendapatkan bagian dari sebuah data dengan tipe data timestamp.

Query	<pre>SELECT DISTINCT team_name FROM team t NATURAL JOIN `match` m NATURAL JOIN player p NATURAL JOIN country c WHERE c.country_name LIKE 'A%' AND m.Man_of_the_Match = p.Player_id\ AND p.country_name = c.country_id AND YEAR(m.match_date) = 2009 AND EXISTS (SELECT team_id FROM team t1 NATURAL JOIN `match` m1 WHERE YEAR(m1.toss_winner) = 2016);</pre>
SS Query dan Hasil	

```

MariaDB [prak2]> SELECT DISTINCT team_name
-> FROM team t NATURAL JOIN `match` m NATURAL JOIN player p NATURAL JOIN country c
-> WHERE c.country_name LIKE 'A%'
-> AND m.Man_of_the_Match = p.Player_id\
-> AND p.country_name = c.country_id
-> AND YEAR(m.match_date) = 2009
-> AND EXISTS
-> (SELECT team_id
-> FROM team t1 NATURAL JOIN `match` m1
-> WHERE YEAR(m1.toss_winner) = 2016);
Empty set, 577 warnings (0,003 sec)

```

10. Pemain Baru

Coach Didot ingin mencari pemain baru untuk timnya. Pemain-pemain yang ingin ia cari pernah bermain untuk suatu tim yang paling sering bertanding sebagai team_1 di Sharjah Cricket Stadium. Bantulah Coach Didot untuk menemukan nama-nama pemain tersebut dengan catatan pemain-pemain tersebut harus memiliki bowling_skill = Legbreak dan batting_hand = Left-hand bat.

Anda diwajibkan menerapkan *subquery* untuk persoalan ini. *Keyword subquery* yang diperbolehkan adalah WITH (tidak boleh menggunakan *keyword* lain)

[HINT]: *Gunakan backtick (`) untuk nama tabel yang sama dengan salah satu reserved keyword SQL.*

Query	
SS Query dan Hasil	

