

ASSIGNMENT – 10

Consider the following tables:

MATCH (match_id, team1, team2, ground, mdate, winner)

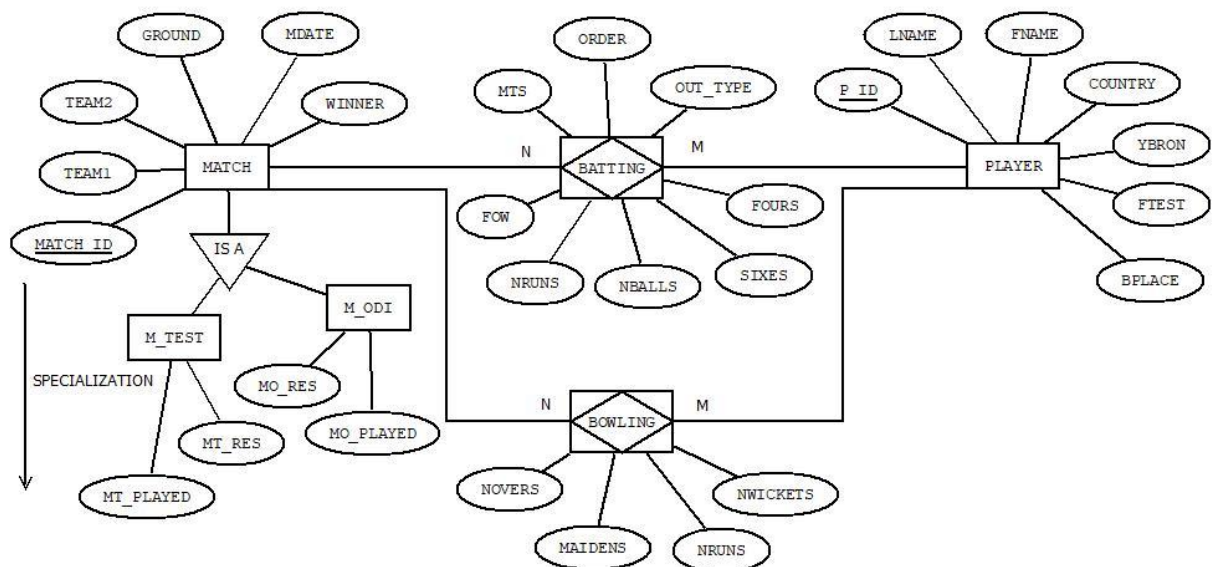
PLAYER (p_id, lname, fname, country, yborn, bplace, ftest)

BATTING (match_id, p_id, mts, order, out_type, fow, nruns, nballs, fours, sixes)

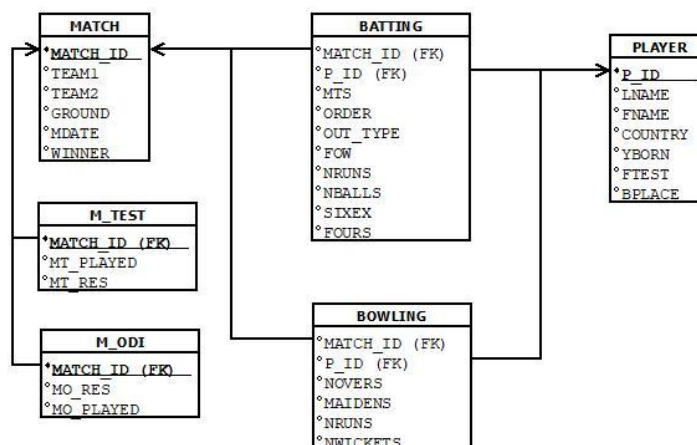
BOWLING (match_id, p_id, novers, maidens, nruns, nwickets)

1. Draw the appropriate ER, EER and Relational model for the given data.

EER DIAGRAM



RELATIONAL MODEL



Assumptions: -

- Each match is uniquely identified by a match_id.
- Every match involves exactly two teams: team1 and team2.
- The match takes place at a specific ground on a specific mdate.
- The winner must be either team1, team2, or 'Draw'.
- Matches are specialized into two types: M_TEST (Test matches) and M_ODI (One Day Internationals).
- M_TEST includes mt_played (number of days played) and mt_res (detailed Test result).
- M_ODI includes mo_played (overs played) and mo_res (detailed ODI result).
- Every player is uniquely identified by p_id.
- Player details include fname, lname, country, yborn (year of birth), bplace (birthplace), and ftest (year of first Test).
- A player can play in multiple matches and may bat, bowl, or both in a match.
- The batting table records a player's batting details per match including order, mts (minutes), out_type, fow, nruns, nballs, fours, and sixes.
- The bowling table records a player's bowling stats per match including novers, maidens, nruns, and nwickets.
- match_id and p_id in batting and bowling act as foreign keys referencing the match and player tables respectively.
- Every performance entry in batting or bowling assumes the player participated in that match.
- No attribute in a primary key or required foreign key can be null.
- All statistical values such as runs, balls, overs, and wickets must be valid and within logical constraints (e.g., overs < 10 in ODI).
- A player cannot appear in a match unless they have either batting or bowling performance recorded.
- The winner column in the match table must not be null and must match one of the participating teams or be 'Draw'.
- The model supports specialization for storing match-type-specific data in separate tables (M_TEST and M_ODI).
- Many-to-many relationships exist between player and match through batting and bowling.

2. Write SQL expressions for the following:

Create Table and Insert Value: -

a) MATCH

```
create table match (  
    match_id varchar2(10) check (match_id like 'M%') primary key,  
    team1 varchar2(20) not null,  
    team2 varchar2(20) not null,  
    ground varchar2(20) not null,  
    mdate date not null,  
    winner varchar2(20),  
    constraint winner_check check (winner = team1 or winner = team2 or  
winner = 'Draw')  
);
```

```
SQL> create table match (  
2     match_id varchar2(10) check (match_id like 'M%') primary key,  
3     team1 varchar2(20) not null,  
4     team2 varchar2(20) not null,  
5     ground varchar2(20) not null,  
6     mdate date not null,  
7     winner varchar2(20),  
8     constraint winner_check check (winner = team1 or winner = team2 or winner = 'Draw')  
9 );
```

Table created.

```
SQL> desc match;
```

Name	Null?	Type
MATCH_ID	NOT NULL	VARCHAR2(10)
TEAM1	NOT NULL	VARCHAR2(20)
TEAM2	NOT NULL	VARCHAR2(20)
GROUND	NOT NULL	VARCHAR2(20)
MDATE	NOT NULL	DATE
WINNER		VARCHAR2(20)

insert all

```
    INTO match VALUES ('M001', 'India', 'Australia', 'Wankhede', TO_DATE  
( '2024-03-15', 'YYYY-MM-DD'), 'India')
```

```
    INTO match VALUES ('M002', 'England', 'Pakistan', 'Lords', TO_DATE  
( '2024-04-10', 'YYYY-MM-DD'), 'Draw')
```

```
    INTO match VALUES ('M003', 'England', 'New Zealand', 'Newlands',  
TO_DATE ( '2024-02-25', 'YYYY-MM-DD'), 'New Zealand')
```

```
    INTO match VALUES ('M004', 'Sri Lanka', 'Bangladesh', 'Sri Lanka',  
TO_DATE ( '2024-05-05', 'YYYY-MM-DD'), 'Bangladesh')
```

```
    INTO match VALUES ('M005', 'India', 'England', 'Eden Gardens',  
TO_DATE ( '2024-06-12', 'YYYY-MM-DD'), 'England')
```

```
    INTO match VALUES ('M006', 'India', 'Australia', 'Sydney', TO_DATE  
( '2024-03-18', 'YYYY-MM-DD'), 'India')
```

INTO match VALUES ('M007', 'India', 'Sri Lanka', 'Sri Lanka', TO_DATE ('2018-04-28', 'YYYY-MM-DD'), 'India')

select * from dual;

```
SQL>
SQL> insert all
  2      INTO match VALUES ('M001', 'India', 'Australia', 'Wankhede', TO_DATE ('2024-03-15', 'YYYY-MM-DD'), 'India')
  3      INTO match VALUES ('M002', 'England', 'Pakistan', 'Lords', TO_DATE ('2024-04-10', 'YYYY-MM-DD'), 'Draw')
  4      INTO match VALUES ('M003', 'England', 'New Zealand', 'Newlands', TO_DATE ('2024-02-25', 'YYYY-MM-DD'), 'New Zealand')
  5      INTO match VALUES ('M004', 'Sri Lanka', 'Bangladesh', 'Sri Lanka', TO_DATE ('2024-05-05', 'YYYY-MM-DD'), 'Bangladesh')
  6      INTO match VALUES ('M005', 'India', 'England', 'Eden Gardens', TO_DATE ('2024-06-12', 'YYYY-MM-DD'), 'England')
  7      INTO match VALUES ('M006', 'India', 'Australia', 'Sydney', TO_DATE ('2024-03-18', 'YYYY-MM-DD'), 'India')
  8      INTO match VALUES ('M007', 'India', 'Sri Lanka', 'Sri Lanka', TO_DATE ('2018-04-28', 'YYYY-MM-DD'), 'India')
  9  select * from dual;

7 rows created.

SQL> select * from match;
```

MATCH_ID	TEAM1	TEAM2	GROUND	MDATE	WINNER
M001	India	Australia	Wankhede	15-MAR-24	India
M002	England	Pakistan	Lords	10-APR-24	Draw
M003	England	New Zealand	Newlands	25-FEB-24	New Zealand
M004	Sri Lanka	Bangladesh	Sri Lanka	05-MAY-24	Bangladesh
M005	India	England	Eden Gardens	12-JUN-24	England
M006	India	Australia	Sydney	18-MAR-24	India
M007	India	Sri Lanka	Sri Lanka	28-APR-18	India

```
7 rows selected.
```

b) PLAYER

create table player (

p_id varchar2(20) check (p_id like 'P%') primary key,

fname varchar2(20) not null,

lname varchar2(20) not null,

country varchar2(20) not null,

yborn number check (yborn < 2009) not null,

bplace varchar2(20) not null,

ftest number (2)

);

```
SQL> create table player(
  2      p_id varchar2(10) primary key,
  3      fname varchar2(20) not null,
  4      lname varchar2(20) not null,
  5      country varchar2(20) not null,
  6      yborn number check (yborn < 2009) not null,
  7      bplace varchar2(20) not null,
  8      ftest number(2)
  9 );
```

Table created.

```
SQL> desc player;
```

Name	Null?	Type
P_ID	NOT NULL	VARCHAR2(10)
FNAME	NOT NULL	VARCHAR2(20)
LNAME	NOT NULL	VARCHAR2(20)
COUNTRY	NOT NULL	VARCHAR2(20)
YBORN	NOT NULL	NUMBER
BPLACE	NOT NULL	VARCHAR2(20)
FTEST		NUMBER(2)

insert all

```

    INTO player VALUES ('P001', 'Virat', 'Kohli', 'India', 1988, 'Delhi', 11)
    INTO player VALUES ('27001', 'Steve', 'Smith', 'Australia', 1989, 'Sydney',
10)
    INTO player VALUES ('P003', 'Joe', 'Root', 'England', 1990, 'Sheffield', 12)
    INTO player VALUES ('P004', 'Kane', 'Williamson', 'New Zealand', 1990,
'Tauranga', 10)
    INTO player VALUES ('P005', 'Babar', 'Azam', 'Pakistan', 1994, 'Lahore',
16)
    INTO player VALUES ('P006', 'MS', 'Dhoni', 'India', 1987, 'Ranchi', 20)
    INTO player VALUES ('P007', 'Sakib Ul', 'Hasan', 'Bangladesh', 1995,
'Dhaka', 19)
select * from dual;

```

```

SQL>
SQL> insert all
2     INTO player VALUES ('P001', 'Virat', 'Kohli', 'India', 1988, 'Delhi', 11)
3     INTO player VALUES ('27001', 'Steve', 'Smith', 'Australia', 1989, 'Sydney', 10)
4     INTO player VALUES ('P003', 'Joe', 'Root', 'England', 1990, 'Sheffield', 12)
5     INTO player VALUES ('P004', 'Kane', 'Williamson', 'New Zealand', 1990, 'Tauranga', 10)
6     INTO player VALUES ('P005', 'Babar', 'Azam', 'Pakistan', 1994, 'Lahore', 16)
7     INTO player VALUES ('P006', 'MS', 'Dhoni', 'India', 1987, 'Ranchi', 20)
8     INTO player VALUES ('P007', 'Sakib Ul', 'Hasan', 'Bangladesh', 1995, 'Dhaka', 19)
9  select * from dual;

```

7 rows created.

SQL> select * from player;

P_ID	FNAME	LNAME	COUNTRY	YBORN	BPLACE	FTEST
P001	Virat	Kohli	India	1988	Delhi	11
27001	Steve	Smith	Australia	1989	Sydney	10
P003	Joe	Root	England	1990	Sheffield	12
P004	Kane	Williamson	New Zealand	1990	Tauranga	10
P005	Babar	Azam	Pakistan	1994	Lahore	16
P006	MS	Dhoni	India	1987	Ranchi	20
P007	Sakib Ul	Hasan	Bangladesh	1995	Dhaka	19

7 rows selected.

c) **BATTING**

```

create table batting (
    match_id varchar2(10) references match(match_id) on delete cascade,
    p_id varchar2(10) references player (p_id) on delete cascade,
    mts number(2) check (mts >= 0),
    batting_order number(2) check (batting_order > 0 or batting_order < 12),
    out_type varchar2(10) check (out_type in ('b', 'c', 'st', 'lbw', 'run out', 'not
out')),
    fow number(2) check (fow >= 0),
    nruns number(4) check (nruns >= 0),
    nballs number(3) check (nballs >= 0 and nballs < 300),
    fours number(3) check (fours >= 0 and fours <= 251),
    sixes number(3) check (sixes >= 0 and sixes <= 251)
);

```

```

SQL> create table batting (
2     match_id varchar2(10) references match(match_id),
3     p_id varchar2(10) references player (p_id),
4     mts number(2) check (mts >= 0),
5     batting_order number(2) check (batting_order > 0 or batting_order < 12),
6     out_type varchar2(10) check (out_type in ('b', 'c', 'st', 'lbw', 'run out', 'not out')),
7     fow number(2) check (fow >= 0),
8     nruns number(4) check (nruns >= 0),
9     nballs number(3) check (nballs >= 0 and nballs < 300),
10    fours number(3) check (fours >= 0 and fours <= 251),
11    sixes number(3) check (sixes >= 0 and sixes <= 251)
12 );

```

Table created.

```
SQL> desc batting;
```

Name	Null?	Type
MATCH_ID		VARCHAR2(10)
P_ID		VARCHAR2(10)
MTS		NUMBER(2)
BATTING_ORDER		NUMBER(2)
OUT_TYPE		VARCHAR2(10)
FOW		NUMBER(2)
NRUNS		NUMBER(4)
NBALLS		NUMBER(3)
FOURS		NUMBER(3)
SIXES		NUMBER(3)

insert all

```

    INTO batting VALUES ('M001', 'P001', 99, 3, 'b', 2, 75, 95, 8, 1)
    INTO batting VALUES ('M001', '27001', 90, 4, 'c', 3, 105, 90, 6, 5)
    INTO batting VALUES ('M003', 'P003', 96, 2, 'lbw', 1, 88, 102, 10, 0)
    INTO batting VALUES ('M003', 'P004', 85, 5, 'not out', null, 25, 35, 1, 2)
    INTO batting VALUES ('M004', 'P007', 81, 2, 'run out', 6, 98, 123, 3, 5)
    INTO batting VALUES ('M005', 'P005', 77, 1, 'b', 1, 0, 1, 0, 0)
    INTO batting VALUES ('M002', 'P005', 77, 1, 'b', 1, 0, 1, 0, 0)
    INTO batting VALUES ('M006', '27001', 90, 1, 'c', 3, 62, 70, 6, 2)
    INTO batting VALUES ('M001', 'P006', 93, 5, 'c', 5, 153, 100, 10, 9)
    INTO batting VALUES ('M007', 'P006', 93, 5, 'c', 5, 53, 22, 5, 2)
    INTO batting VALUES ('M007', 'P001', 99, 3, 'not out', null, 123, 110, 4, 9)

```

```
select * from dual;
```

```

SQL>
SQL> insert all
2 INTO batting VALUES ('M001', 'P001', 99, 3, 'b', 2, 75, 95, 8, 1)
3 INTO batting VALUES ('M001', '27001', 90, 4, 'c', 3, 105, 90, 6, 5)
4 INTO batting VALUES ('M003', 'P003', 96, 2, 'lbw', 1, 88, 102, 10, 0)
5 INTO batting VALUES ('M003', 'P004', 85, 5, 'not out', null, 25, 35, 1, 2)
6 INTO batting VALUES ('M004', 'P007', 81, 2, 'run out', 6, 98, 123, 3, 5)
7 INTO batting VALUES ('M005', 'P005', 77, 1, 'b', 1, 0, 1, 0, 0)
8 INTO batting VALUES ('M002', 'P005', 77, 1, 'b', 1, 0, 1, 0, 0)
9 INTO batting VALUES ('M006', '27001', 90, 1, 'c', 3, 62, 70, 6, 2)
10 INTO batting VALUES ('M001', 'P006', 93, 5, 'c', 5, 153, 100, 10, 9)
11 INTO batting VALUES ('M007', 'P006', 93, 5, 'c', 5, 53, 22, 5, 2)
12 INTO batting VALUES ('M007', 'P001', 99, 3, 'not out', null, 123, 110, 4, 9)
13 select * from dual;

```

11 rows created.

```
SQL> select * from batting;
```

MATCH_ID	P_ID	MTS	BATTING_ORDER	OUT_TYPE	FOW	NRUNS	NBALLS	FOURS	SIXES
M001	P001	99	3	b	2	75	95	8	1
M001	27001	90	4	c	3	105	90	6	5
M003	P003	96	2	lbw	1	88	102	10	0
M003	P004	85	5	not out		25	35	1	2
M004	P007	81	2	run out	6	98	123	3	5
M005	P005	77	1	b	1	0	1	0	0
M002	P005	77	1	b	1	0	1	0	0
M006	27001	90	1	c	3	62	70	6	2
M001	P006	93	5	c	5	153	100	10	9
M007	P006	93	5	c	5	53	22	5	2
M007	P001	99	3	not out		123	110	4	9

11 rows selected.

d) BOWLING

create table bowling (

match_id varchar2 (10) references match(match_id) on delete cascade,

p_id varchar2 (10) references player (p_id) on delete cascade,

novers number (3, 1) check (novers >= 0 and novers < 10),

maidens number (2) check (maidens >= 0 and maidens < 10),

nruns number (3) check (nruns >= 0 and nruns <= 360),

nwickets number (2) check (nwickets >= 0 and nwickets <= 10)

);

```

SQL> create table bowling(
2     match_id varchar2(10) references match(match_id),
3     p_id varchar2(10) references player (p_id),
4     novers number(3, 1) check (novers >= 0 and novers < 10),
5     maidens number(2) check (maidens >= 0 and maidens < 10),
6     nruns number(3) check (nruns >= 0 and nruns <= 360),
7     nwickets number(2) check (nwickets >= 0 and nwickets <= 10)
8 );

```

Table created.

```
SQL> desc bowling;
```

Name	Null?	Type
MATCH_ID		VARCHAR2(10)
P_ID		VARCHAR2(10)
NOVERS		NUMBER(3,1)
MAIDENS		NUMBER(2)
NRUNS		NUMBER(3)
NWICKETS		NUMBER(2)

insert all

```

    INTO bowling VALUES ('M001', 'P001', 8.4, 2, 45, 3)
    INTO bowling VALUES ('M002', '27001', 9.0, 1, 50, 2)
    INTO bowling VALUES ('M003', 'P003', 6.5, 0, 39, 1)
    INTO bowling VALUES ('M004', 'P004', 7.2, 1, 60, 4)
    INTO bowling VALUES ('M005', 'P005', 5.0, 0, 35, 0)
select * from dual;

SQL>
SQL> insert all
  2      INTO bowling VALUES('M001', 'P001', 8.4, 2, 45, 3)
  3      INTO bowling VALUES('M002', '27001', 9.0, 1, 50, 2)
  4      INTO bowling VALUES('M003', 'P003', 6.5, 0, 39, 1)
  5      INTO bowling VALUES('M004', 'P004', 7.2, 1, 60, 4)
  6      INTO bowling VALUES('M005', 'P005', 5.0, 0, 35, 0)
  7  select * from dual;

5 rows created.

SQL> select * from bowling;

```

MATCH_ID	P_ID	NOVERS	MAIDENS	NRUNS	NWICKETS
M001	P001	8.4	2	45	3
M002	27001	9	1	50	2
M003	P003	6.5	0	39	1
M004	P004	7.2	1	60	4
M005	P005	5	0	35	0

e) M_TEST

```

create table m_test(
    match_id varchar2(10) references match(match_id) on delete cascade,
    primary key(match_id),
    mt_played number(5,1),
    mt_res varchar2(30)
);

```

```

insert all
    into m_test values ('M002', 190, 'Draw')
select * from dual;
select * from m_test;

```



```

SQL>
SQL> create table m_test(
2     match_id varchar2(10) references match(match_id) on delete cascade
3     primary key(match_id),
4     mt_played number(5,1),
5     mt_res varchar2(30)
6 );

```

Table created.

```

SQL>
SQL> insert all
2     into m_test values ('M002', 190, 'Draw')
3     select * from dual;

```

1 row created.

```
SQL> select * from m_test;
```

MATCH_ID	MT_PLAYED	MT_RES
M002	190	Draw

f) M_ODI

```

create table m_odi(
    match_id varchar2(10) references match(match_id) on delete cascade,
    primary key(match_id),
    mo_played number(5,1),
    mto_res varchar2(30)
);

```

```

SQL>
SQL> create table m_odi(
2     match_id varchar2(10) references match(match_id) on delete cascade,
3     primary key(match_id),
4     mo_played number(5,1),
5     mto_res varchar2(30)
6 );

```

Table created.

insert all

```

into m_odi values ('M001', 100, 'India won by 20 runs')
into m_odi values ('M003', 90.4, 'New Zealand won by 5 runs')
into m_odi values ('M004', 95.3, 'Bangladesh won by 3 wickets')
into m_odi values ('M005', 100, 'England won by 1 wickets')
into m_odi values ('M006', 80.5, 'India won by 150 runs')
into m_odi values ('M007', 99.2, 'India won by 10 runs')

```

```
select * from dual;
```

```
select * from m_odi;
```

```

SQL>
SQL> insert all
  2      into m_odi values ('M001', 100, 'India won by 20 runs')
  3      into m_odi values ('M003', 90.4, 'New Zealand won by 5 runs')
  4      into m_odi values ('M004', 95.3, 'Bangladesh won by 3 wickets')
  5      into m_odi values ('M005', 100, 'England won by 1 wickets')
  6      into m_odi values ('M006', 80.5, 'India won by 150 runs')
  7      into m_odi values ('M007', 99.2, 'India won by 10 runs')
  8  select * from dual;

```

6 rows created.

```
SQL> select * from m_odi;
```

MATCH_ID	MO_PLAYED	MTO_RES
M001	100	India won by 20 runs
M003	90.4	New Zealand won by 5 runs
M004	95.3	Bangladesh won by 3 wickets
M005	100	England won by 1 wickets
M006	80.5	India won by 150 runs
M007	99.2	India won by 10 runs

6 rows selected.

SQL QUERIES: -

- i) **Find match ids of those matches in which player 27001 bats and makes more runs than he made at every match he played at Sydney.**

```

➤ select match_id from batting where nruns > (
    select max(nruns) from (
        select * from batting where match_id in (
            select match_id from match where ground='Sydney'
        )
    )
    and p_id = '27001'
)
)

```

and p_id = '27001';

```

SQL> select match_id from batting where nruns > (
  2      select max(nruns) from (
  3          select * from batting where match_id in (
  4              select match_id from match where ground='Sydney'
  5          )
  6      )
  7      and p_id = '27001'
  8  )
  9  and p_id = '27001';

```

MATCH_ID
M001

- ii) Find player ids of players who have scored more than 30 in every ODI match that they have batted.

➤ select distinct p_id from batting where p_id not in (
 select p_id from batting where nruns <= 30 and match_id in (
 select match_id from m_odi
)
);

```
SQL> select distinct p_id from batting where p_id not in (  
2       select p_id from batting where nruns <= 30 and match_id in (  
3           select match_id from m_odi  
4       )  
5  );
```

```
P_ID  
-----  
P001  
27001  
P003  
P007  
P006
```

- iii) Find the ids of players that had a higher average score than the average score for all players when they played in Sri Lanka.

➤ select p_id from (
 select p_id, avg(nruns) as avg_runs from batting group by p_id
) where avg_runs > (
 select avg(nruns) from batting where match_id in (
 select match_id from match where ground = 'Sri Lanka'
)
);

```
SQL> SP2-0042: unknown command "s" - rest of line ignored.  
SQL> select p_id from (  
2       select p_id, avg(nruns) as avg_runs from batting group by p_id  
3  ) where avg_runs > (  
4       select avg(nruns) from batting where match_id in (  
5           select match_id from match where ground = 'Sri Lanka'  
6       )  
7  );
```

```
P_ID  
-----  
P001  
P007  
P006
```