

Institute of Technology, Nirma University

Olympics Management
System

Innovative Assignment
(Phase 2)

Database Management System
(2CS402)

Submitted By:

Mahek Parekh (19BCE154)

Dhruva Patel (19BCE162)

Esha Patel (19BCE169)

Creating Tables and Inserting Records:

1. Creating table country and inserting records:

Creating Table:

create table country (countrycode varchar2(5) primary key, countryname varchar2(15) not null);

Table created.

SQL> desc country

Name	Null?	Type
COUNTRYCODE	NOT NULL	VARCHAR2(5)
COUNTRYNAME	NOT NULL	VARCHAR2(15)

Inserting Records:

SQL> insert into country values ('IND12', 'India');

1 row created.

SQL> insert into country values ('CHI12', 'China');

1 row created.

SQL> insert into country values ('USA99', 'USA');

1 row created.

SQL> insert into country values ('JAP01', 'Japan');

1 row created.

SQL> insert into country values ('ENG47', 'England');

1 row created.

SQL> insert into country values ('RUS14', 'Russia');

1 row created.

SQL> insert into country values ('CAN06', 'Canada');

1 row created.

SQL> insert into country values ('BRA16', 'Brazil');

1 row created.

SQL> insert into country values ('AUS02', 'Australia');

1 row created.

SQL> select * from country;

COUNT COUNTRYNAME

IND12 India

CHI12 China

USA99 USA

JAP01 Japan

ENG47 England

RUS14 Russia

CAN06 Canada

BRA16 Brazil

AUS02 Australia

9 rows selected.

2. Creating table athlete and inserting records:

Creating Table:

SQL> create table athlete (athleteid varchar2(5) primary key, first_name varchar2(15) not null, last_name varchar2(15), dob date, gender char check(gender='F' or gender='M'), countrycode varchar2(5) references country(countrycode));

Table created.

SQL> desc athlete

Name	Null?	Type
-----	-----	-----
ATHLETEID	NOT NULL	VARCHAR2(5)
FIRST_NAME	NOT NULL	VARCHAR2(15)
LAST_NAME		VARCHAR2(15)

DOB	DATE
GENDER	CHAR(1)
COUNTRYCODE	VARCHAR2(5)

Inserting Records:

```
SQL> insert into athlete values('AIND1', 'Deepa', 'Karmakar', '09-AUG-93', 'F', 'IND12');
```

1 row created.

```
SQL> insert into athlete values('AIND2', 'Sakshi', 'Malik', '03-SEP-92', 'F', 'IND12');
```

1 row created.

```
SQL> insert into athlete values('AIND3', 'Vinesh', 'Phogat', '25-AUG-94', 'F', 'IND12');
```

1 row created.

```
SQL> insert into athlete values('AIND4', 'Leander', 'Paes', '17-JUN-73', 'M', 'IND12');
```

1 row created.

```
SQL> insert into athlete values('AUSA1', 'Serena', 'Williams', '26-Sep-81', 'F', 'USA99');
```

1 row created.

```
SQL> insert into athlete values('ACHI1', 'Lin', 'Dan', '14-OCT-83', 'M', 'CHI12');
```

1 row created.

```
SQL> insert into athlete values('AJAP1', 'Naomi', 'Osaka', '14-OCT-83', 'M', 'JAP01');
```

1 row created.

```
SQL> insert into athlete values('AENG1', 'David', 'Beckham', '02-MAY-75', 'M', 'ENG47');
```

1 row created.

```
SQL> insert into athlete values('ARUS1', 'Maria', 'Sharapova', '19-APR-87', 'F', 'RUS14');
```

1 row created.

```
SQL> insert into athlete values('ACAN1', 'Donovan', 'Bailey', '16-DEC-67', 'F', 'CAN06');
```

1 row created.

```
SQL> insert into athlete values('ABRA1', 'Rafaela', 'Silva', '24-APR-92', 'F', 'BRA16');
```

1 row created.

```
SQL> insert into athlete values('AAUS1', 'Ben', 'Simmons', '20-JUL-96', 'M', 'AUS02');
```

1 row created.

```
SQL> insert into athlete values('AIND5', 'Ashvini', 'Ponnappa', '20-AUG-1992', 'F', 'IND12');
```

1 row created.

```
SQL> insert into athlete values('AIND6','Jwala','Gutta','30-JUL-1992','F','IND12');
```

1 row created.

```
SQL> insert into athlete values('AJAP2','Yuki','Fukushima','23-APR-1992','F','JAP01');
```

1 row created.

```
SQL> insert into athlete values('AJAP3','Sayaka','Hirota','26-APR-1992','F','JAP01');
```

1 row created.

```
SQL> select * from athlete;
```

ATHLE	FIRST_NAME	LAST_NAME	DOB	G	COUNT
-------	------------	-----------	-----	---	-------

AIND1	Deepa	Karmakar	09-AUG-93	F	IND12
AIND2	Sakshi	Malik	03-SEP-92	F	IND12
AIND3	Vinesh	Phogat	25-AUG-94	F	IND12
AIND4	Leander	Paes	17-JUN-73	M	IND12
AUSA1	Serena	Williams	26-SEP-81	F	USA99
ACHI1	Lin	Dan	14-OCT-83	M	CHI12
AJAP1	Naomi	Osaka	14-OCT-83	M	JAP01
AENG1	David	Beckham	02-MAY-75	M	ENG47
ARUS1	Maria	Sharapova	19-APR-87	F	RUS14
ACAN1	Donovan	Bailey	16-DEC-67	F	CAN06
ABRA1	Rafaela	Silva	24-APR-92	F	BRA16

ATHLE	FIRST_NAME	LAST_NAME	DOB	G	COUNT
-------	------------	-----------	-----	---	-------

AAUS1	Ben	Simmons	20-JUL-96	M	AUS02
AIND5	Ashvini	Ponnapa	20-AUG-92	F	IND12
AIND6	Jwala	Gutta	30-JUL-92	F	IND12
AJAP2	Yuki	Fukushima	23-APR-92	F	JAP01
AJAP3	Sayaka	Hirota	26-APR-92	F	JAP01

16 rows selected.

3. Creating table referee and inserting records:

Creating Table:

```
SQL> create table referee (refid varchar2(5) primary key, first_name varchar2(15) not null,
last_name varchar2(15), gender char(1) check(gender='M' or gender='F'), countrycode varchar2(5)
references country(countrycode));
```

Table created.

```
SQL> desc referee
```

Name	Null?	Type

REFID	NOT NULL	VARCHAR2(5)
FIRST_NAME	NOT NULL	VARCHAR2(15)
LAST_NAME		VARCHAR2(15)
GENDER		CHAR(1)
COUNTRYCODE		VARCHAR2(5)

Inserting Records:

```
SQL> insert into referee values('RIND1', 'Pranjal', 'Banerjee', 'M', 'IND12');
```

1 row created.

```
SQL> insert into referee values('RCHI1', 'Ching', 'Pong', 'F', 'CHI12');
```

1 row created.

```
SQL> insert into referee values('RUSA1', 'Clinton', 'Dave', 'M', 'USA99');
```

1 row created.

```
SQL> insert into referee values('RJAP1', 'Hakuna', 'Matata', 'F', 'JAP01');
```

1 row created.

```
SQL> insert into referee values('RENG1', 'Thomas', 'George', 'M', 'ENG47');
```

1 row created.

```
SQL> insert into referee values('RRUS1', 'Gabriella', 'Johnson', 'F', 'RUS14');
```

1 row created.

```
SQL> insert into referee values('RCAN1', 'George', 'Kurien', 'M', 'CAN06');
```

1 row created.

```
SQL> insert into referee values('RBRA1', 'Wilson', 'Bond', 'F', 'BRA16');
```

1 row created.

```
SQL> select * from referee;
```

```
REFID FIRST_NAME  LAST_NAME  G COUNT
```

```
-----
```

RIND1	Pranjal	Banerjee	M	IND12
RCHI1	Ching	Pong	F	CHI12
RUSA1	Clinton	Dave	M	USA99
RJAP1	Hakuna	Matata	F	JAP01
RENG1	Thomas	George	M	ENG47
RRUS1	Gabriella	Johnson	F	RUS14
RCAN1	George	Kurien	M	CAN06
RBRA1	Wilson	Bond	F	BRA16

8 rows selected.

4. Creating table olympicsite and inserting records:

Creating Table:

```
SQL> create table olympicsite (year number(4), season char(1) check(season='W' or season='S'), city  
varchar2(15) not null, countrycode varchar2(5) references country(countrycode), primary  
key(year,season));
```

Table created.

```
SQL> desc olympicsite
```

Name	Null?	Type
YEAR	NOT NULL	NUMBER(4)
SEASON	NOT NULL	CHAR(1)
CITY	NOT NULL	VARCHAR2(15)
COUNTRYCODE		VARCHAR2(5)

Inserting Records:

```
SQL> insert into olympicsite values(2016, 'S', 'Rio', 'BRA16');
```

1 row created.

```
SQL> insert into olympicsite values(2020, 'S', 'Tokyo', 'JAP01');
```

1 row created.

```
SQL> insert into olympicsite values(2012, 'S', 'London', 'ENG47');
```

1 row created.

```
SQL> insert into olympicsite values(2008, 'S', 'Beijing', 'CHI12');
```

1 row created.

```
SQL> insert into olympicsite values(2022, 'W', 'Beijing', 'CHI12');
```

1 row created.

```
SQL> select * from olympicsite;
```

YEAR	S	CITY	COUNT
2016	S	Rio	BRA16
2020	S	Tokyo	JAP01
2012	S	London	ENG47
2008	S	Beijing	CHI12
2022	W	Beijing	CHI12

5. Creating table partof and inserting records:

Creating Table:

```
SQL> create table partof (athleteid varchar2(5) primary key references athlete(athleteid), teamid  
varchar2(5));
```

Table created.

```
SQL> desc partof
```

Name	Null?	Type

ATHLETEID	NOT NULL VARCHAR2(5)
TEAMID	VARCHAR2(5)

Inserting Records:

SQL> insert into partof values('AIND6','BDWD1');

1 row created.

SQL> insert into partof values('AIND5','BDWD1');

1 row created.

SQL> insert into partof values('AJAP2','BDWD2');

1 row created.

SQL> insert into partof values('AJAP3','BDWD2');

1 row created.

SQL> select * from partof;

ATHLE TEAMI

AIND6 BDWD1

AIND5 BDWD1

AJAP2 BDWD2

AJAP3 BDWD2

6. Creating table judges and inserting records:

Creating Table:

SQL> create table judges (matchid varchar2(5) primary key, refid varchar(2) references referee(refid));

Table created.

SQL> desc judges

Name	Null?	Type
-----	-----	-----

MATCHID	NOT NULL VARCHAR2(5)
REFID	VARCHAR2(2)

Inserting Records:

SQL> insert into judges values('MBD01','RCHI1');

1 row created.

SQL> select * from judges;

MATCH REFID

MBD01 RCHI1

7. Creating table result and inserting records:

Creating Table:

SQL> create table result (matchid varchar2(5) references judges(matchid), athleteid varchar2(5) references athlete(athleteid), position number(2) check(position>0), primary key(matchid,athleteid));

Table created.

SQL> desc result

Name	Null?	Type
-----	-----	-----
MATCHID	NOT NULL	VARCHAR2(5)
ATHLETEID	NOT NULL	VARCHAR2(5)
POSITION		NUMBER(2)

Inserting Records:

SQL> insert into result values('MBD01','AIND5','');

1 row created.

SQL> insert into result values('MBD01','AIND6','');

1 row created.

```
SQL> insert into result values('MBD01','AJAP2','');
```

1 row created.

```
SQL> insert into result values('MBD01','AJAP3','');
```

1 row created.

```
SQL> select * from result;
```

MATCH ATHLE POSITION

MBD01 AIND5

MBD01 AIND6

MBD01 AJAP2

MBD01 AJAP3

8. Creating table teamparticipatedinevent and adding records:

Creating Table:

```
SQL> create table teamparticipatedinevent(teamid varchar2(5), eventid varchar2(5), sportname  
varchar2(15), eventname varchar2(15), primary key(teamid, eventid));
```

Table created.

```
SQL> desc teamparticipatedinevent
```

Name	Null?	Type
TEAMID	NOT NULL	VARCHAR2(5)
EVENTID	NOT NULL	VARCHAR2(5)
SPORTNAME		VARCHAR2(15)
EVENTNAME		VARCHAR2(15)

Inserting Records:

```
SQL> insert into teamparticipatedinevent values('BDWD1','BDWD1','Badminton','Women Doubles');
```

1 row created.

```
SQL> insert into teamparticipatedinevent values('BDWD2','BDWD1','Badminton','Women Doubles');
```

1 row created.

```
SQL> select * from teamparticipatedinevent;
```

TEAM	EVENT	SPORTNAME	EVENTNAME
BDWD1	BDWD1	Badminton	Women Doubles
BDWD2	BDWD1	Badminton	Women Doubles

9. Creating table matchinevent and inserting records:

Creating Table:

```
SQL> create table matchinevent(matchid varchar2(5) primary key references judges(matchid),  
eventid varchar2(5), sportname varchar2(15), eventname varchar2(15), matchdate date, time  
number(2,2) check(time>=00.00 and time<=23.59), venue varchar2(15));
```

Table created.

```
SQL> desc matchinevent
```

Name	Null?	Type
MATCHID	NOT NULL	VARCHAR2(5)
EVENTID		VARCHAR2(5)
SPORTNAME		VARCHAR2(15)
EVENTNAME		VARCHAR2(15)
MATCHDATE		DATE
TIME		NUMBER(2,2)
VENUE		VARCHAR2(15)

Inserting Records:

```
SQL> insert into matchinevent values('MBD01','BDWD1','Badminton','Women Doubles','20-AUG-21',.10,'JNS');
```

1 row created.

```
SQL> select * from matchinevent;
```

MATCH	EVENT	SPORTNAME	EVENTNAME	MATCHDATE	TIME	VENUE
MBD01	BDWD1	Badminton	Women Doubles	20-AUG-21	.1	JNS

Queries Solved:

- To identify age of athlete which is derived attribute for athlete entity:

Query: SQL> select athleteid,floor(months_between(sysdate,dob)/12) from athlete;

Output:

ATHLE FLOOR(MONTHS_BETWEEN(SYSDATE,DOB)/12)

AIND1	27
AIND2	28
AIND3	26
AIND4	47
AUSA1	39
ACHI1	37
AJAP1	37
AENG1	45
ARUS1	34
ACAN1	53
ABRA1	29

ATHLE FLOOR(MONTHS_BETWEEN(SYSDATE,DOB)/12)

AAUS1	24
AIND5	28
AIND6	28
AJAP2	29
AJAP3	29

16 rows selected.

- **To identify country of the team which is a derived attribute for team entity:**

Query: SQL> select distinct teamid,countrycode from partof inner join athlete on athlete.athleteid=partof.athleteid;

Output:

```
TEAMID COUNT
-----
BDWD1 IND12
BDWD2 JAP01
```

- **To display number of athletes from the different countries:**

Query: SQL> select count(*),countrycode from athlete group by countrycode;

Output:

```
COUNT(*) COUNT
-----
6 IND12
1 RUS14
1 CAN06
1 BRA16
1 AUS02
3 JAP01
1 USA99
1 CHI12
1 ENG47
```

9 rows selected.

- **To display which athlete is associated with which sport and which event:**

Query: SQL> select athlete.athleteid,eventname,sportname from (athlete inner join result on athlete.athleteid=result.athleteid) inner join machinevent on machinevent.matchid=result.matchid;

Output:

```
ATHLE EVENTNAME SPORTNAME
-----
```

AIND5 Women Doubles Badminton
 AIND6 Women Doubles Badminton
 AJAP2 Women Doubles Badminton
 AJAP3 Women Doubles Badminton

- **To display names of the country which have hosted both Summer as well as winter Olympics:**

Query: SQL> select a.countrycode from olympicsite a cross join olympicsite b where a.countrycode=b.countrycode and a.season='S' and b.season='W';

Output:

COUNT

 CHI12

- **As it is rule in Olympics that the country of the referee cannot be similar to the countries of any of the participants. So, identify the details of referee who are eligible to judge a particular match :**

Query: SQL> select * from referee where countrycode not in (select countrycode from result inner join athlete on athlete.athleteid=result.athleteid);

Output:

REFID	FIRST_NAME	LAST_NAME	G	COUNT
RCHI1	Ching	Pong	F	CHI12
RUSA1	Clinton	Dave	M	USA99
RENG1	Thomas	George	M	ENG47
RRUS1	Gabriella	Johnson	F	RUS14
RCAN1	George	Kurien	M	CAN06
RBRA1	Wilson	Bond	F	BRA16

6 rows selected.