

5

Managing Data in Different Time Zones

Objectives

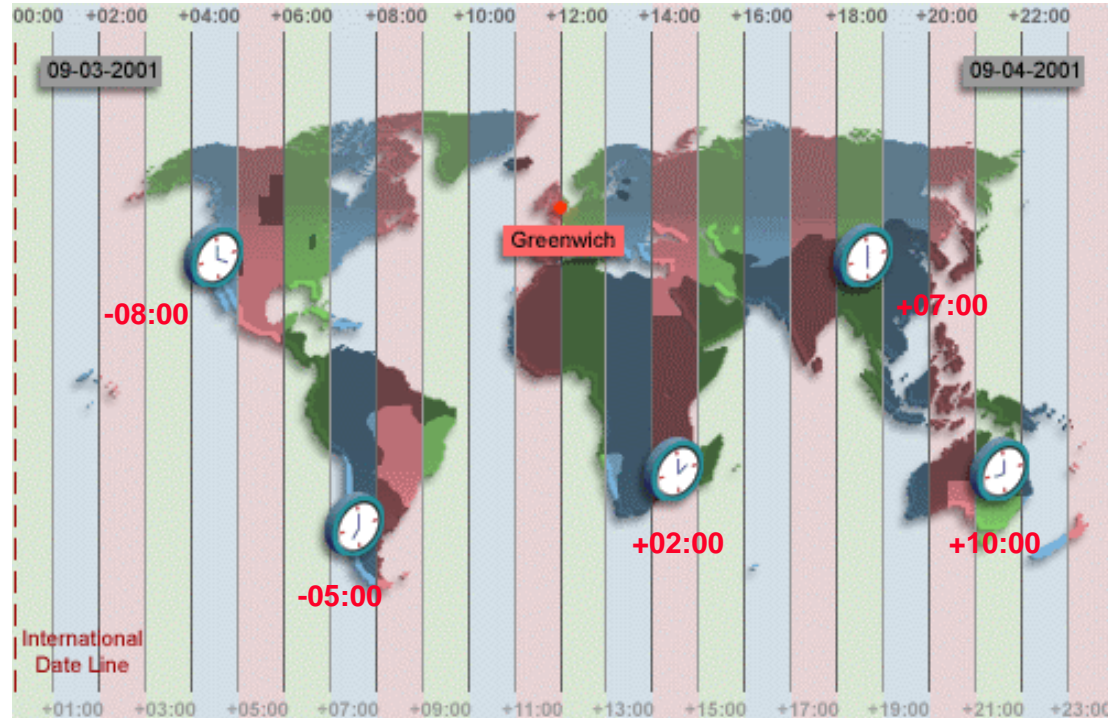
After completing this lesson, you should be able to do the following:

- Use data types similar to `DATE` that store fractional seconds and track time zones
- Use data types that store the difference between two datetime values
- Use the following datetime functions:
 - `CURRENT_DATE`
 - `CURRENT_TIMESTAMP`
 - `LOCALTIMESTAMP`
 - `DBTIMEZONE`
 - `SESSIONTIMEZONE`
 - `EXTRACT`
 - `TZ_OFFSET`
 - `FROM_TZ`
 - `TO_TIMESTAMP`
 - `TO_YMINTERVAL`
 - `TO_DSINTERVAL`

Lesson Agenda

- `CURRENT_DATE`, `CURRENT_TIMESTAMP`, and `LOCALTIMESTAMP`
- `INTERVAL` data types
- Using the following functions:
 - `EXTRACT`
 - `TZ_OFFSET`
 - `FROM_TZ`
 - `TO_TIMESTAMP`
 - `TO_YMINTERVAL`
 - `TO_DSINTERVAL`

Time Zones



The image represents the time for each time zone when Greenwich time is 12:00.

TIME_ZONE Session Parameter

TIME_ZONE may be set to:

- An absolute offset
- Database time zone
- OS local time zone
- A named region

```
ALTER SESSION SET TIME_ZONE = '-05:00';  
ALTER SESSION SET TIME_ZONE = dbtimezone;  
ALTER SESSION SET TIME_ZONE = local;  
ALTER SESSION SET TIME_ZONE = 'America/New_York';
```

CURRENT_DATE, CURRENT_TIMESTAMP, and LOCALTIMESTAMP

- **CURRENT_DATE:**
 - Returns the current date from the user session
 - Has a data type of DATE
- **CURRENT_TIMESTAMP:**
 - Returns the current date and time from the user session
 - Has a data type of TIMESTAMP WITH TIME ZONE
- **LOCALTIMESTAMP:**
 - Returns the current date and time from the user session
 - Has a data type of TIMESTAMP

Comparing Date and Time in a Session's Time Zone

The `TIME_ZONE` parameter is set to `-5:00` and then `SELECT` statements for each date and time are executed to compare differences.

```
ALTER SESSION  
SET NLS_DATE_FORMAT = 'DD-MON-YYYY HH24:MI:SS';  
ALTER SESSION SET TIME_ZONE = '-5:00';
```

```
SELECT SESSIONTIMEZONE, CURRENT_DATE FROM DUAL;
```

1

```
SELECT SESSIONTIMEZONE, CURRENT_TIMESTAMP FROM DUAL;
```

2

```
SELECT SESSIONTIMEZONE, LOCALTIMESTAMP FROM DUAL;
```

3

Comparing Date and Time in a Session's Time Zone

Results of queries:

```
ALTER SESSION succeeded.
```

SESSIONTIMEZONE	CURRENT_DATE
1 -05:00	23-JUN-2009 01:34:52

1

SESSIONTIMEZONE	CURRENT_TIMESTAMP
1 -05:00	23-JUN-09 01.35.26.239882000 AM -05:00

2

SESSIONTIMEZONE	LOCALTIMESTAMP
1 -05:00	23-JUN-09 01.36.21.811798000 AM

3

DBTIMEZONE and SESSIONTIMEZONE

- Display the value of the database time zone:

```
SELECT DBTIMEZONE FROM DUAL;
```

	DBTIMEZONE
1	+00:00

- Display the value of the session's time zone:

```
SELECT SESSIONTIMEZONE FROM DUAL;
```

	SESSIONTIMEZONE
1	-05:00

TIMESTAMP Data Types

Data Type	Fields
TIMESTAMP	Year, Month, Day, Hour, Minute, Second with fractional seconds
TIMESTAMP WITH TIME ZONE	Same as the TIMESTAMP data type; also includes: TIMEZONE_HOUR, and TIMEZONE_MINUTE or TIMEZONE_REGION
TIMESTAMP WITH LOCAL TIME ZONE	Same as the TIMESTAMP data type; also includes a time zone offset in its value

TIMESTAMP Fields

Datetime Field	Valid Values
YEAR	–4712 to 9999 (excluding year 0)
MONTH	01 to 12
DAY	01 to 31
HOURL	00 to 23
MINUTE	00 to 59
SECOND	00 to 59.9(N) where 9(N) is precision
TIMEZONE_HOUR	–12 to 14
TIMEZONE_MINUTE	00 to 59

Difference Between DATE and TIMESTAMP

A

```
-- when hire_date is  
of type DATE
```

```
SELECT hire_date  
FROM employees;
```

	HIRE_DATE
1	21-JUN-99
2	13-JAN-00
3	17-SEP-87
4	17-FEB-96
5	17-AUG-97
6	07-JUN-94
7	07-JUN-94
8	07-JUN-94

B

```
ALTER TABLE employees  
MODIFY hire_date TIMESTAMP;
```

```
SELECT hire_date  
FROM employees;
```

	HIRE_DATE
1	21-JUN-99 12.00.00.000000000 AM
2	13-JAN-00 12.00.00.000000000 AM
3	17-SEP-87 12.00.00.000000000 AM
4	17-FEB-96 12.00.00.000000000 AM
5	17-AUG-97 12.00.00.000000000 AM
6	07-JUN-94 12.00.00.000000000 AM
7	07-JUN-94 12.00.00.000000000 AM
8	07-JUN-94 12.00.00.000000000 AM

...

Comparing TIMESTAMP Data Types

```
CREATE TABLE web_orders  
(order_date TIMESTAMP WITH TIME ZONE,  
 delivery_time TIMESTAMP WITH LOCAL TIME ZONE);
```

```
INSERT INTO web_orders values  
(current_date, current_timestamp + 2);
```

```
SELECT * FROM web_orders;
```

	ORDER_DATE	DELIVERY_TIME
1	23-JUN-09 01.56.39.000000000 AM -05:00	25-JUN-09 01.56.39.000000000 AM

Lesson Agenda

- CURRENT_DATE, CURRENT_TIMESTAMP, and LOCALTIMESTAMP
- INTERVAL data types
- Using the following functions:
 - EXTRACT
 - TZ_OFFSET
 - FROM_TZ
 - TO_TIMESTAMP
 - TO_YMINTERVAL
 - TO_DSINTERVAL

INTERVAL Data Types

- INTERVAL data types are used to store the difference between two datetime values.
- There are two classes of intervals:
 - Year-month
 - Day-time
- The precision of the interval is:
 - The actual subset of fields that constitutes an interval
 - Specified in the interval qualifier

Data Type	Fields
INTERVAL YEAR TO MONTH	Year, Month
INTERVAL DAY TO SECOND	Days, Hour, Minute, Second with fractional seconds

INTERVAL Fields

INTERVAL Field	Valid Values for Interval
YEAR	Any positive or negative integer
MONTH	00 to 11
DAY	Any positive or negative integer
HOURL	00 to 23
MINUTE	00 to 59
SECOND	00 to 59.9(N) where 9(N) is precision

INTERVAL YEAR TO MONTH: Example

```
CREATE TABLE warranty
(prod_id number,  warranty_time INTERVAL YEAR(3) TO
MONTH);

INSERT INTO warranty VALUES (123, INTERVAL '8' MONTH);
INSERT INTO warranty VALUES (155, INTERVAL '200'
YEAR(3));
INSERT INTO warranty VALUES (678, '200-11');
SELECT * FROM warranty;
```

	PROD_ID	WARRANTY_TIME
1	123	0-8
2	155	200-0
3	678	200-11

INTERVAL DAY TO SECOND

Data Type: Example

```
CREATE TABLE lab
( exp_id number, test_time INTERVAL DAY(2) TO SECOND);

INSERT INTO lab VALUES (100012, '90 00:00:00');
INSERT INTO lab VALUES (56098,
    INTERVAL '6 03:30:16' DAY TO SECOND);
```

```
SELECT * FROM lab;
```

	EXP_ID	TEST_TIME
1	100012	90 0:0:0.0
2	56098	6 3:30:16.0

Lesson Agenda

- CURRENT_DATE, CURRENT_TIMESTAMP, and LOCALTIMESTAMP
- INTERVAL data types
- Using the following functions:
 - EXTRACT
 - TZ_OFFSET
 - FROM_TZ
 - TO_TIMESTAMP
 - TO_YMINTERVAL
 - TO_DSINTERVAL

EXTRACT

- Display the YEAR component from the SYSDATE.

```
SELECT EXTRACT (YEAR FROM SYSDATE) FROM DUAL;
```

	EXTRACT(YEARFROMSYSDATE)
1	2009

- Display the MONTH component from the HIRE_DATE for those employees whose MANAGER_ID is 100.

```
SELECT last name, hire date,  
       EXTRACT (MONTH FROM HIRE_DATE)  
FROM employees  
WHERE manager_id = 100;
```

	LAST_NAME	HIRE_DATE	EXTRACT(MONTHFROMHIRE_DATE)
1	Hartstein	17-FEB-1996 00:00:00	2
2	Kochhar	21-SEP-1989 00:00:00	9
3	De Haan	13-JAN-1993 00:00:00	1
4	Raphaely	07-DEC-1994 00:00:00	12
5	Weiss	18-JUL-1996 00:00:00	7

TZ_OFFSET

Display the time zone offset for the 'US/Eastern',
'Canada/Yukon' and 'Europe/London' time zones:


```
SELECT TZ_OFFSET('US/Eastern'),  
       TZ_OFFSET('Canada/Yukon'),  
       TZ_OFFSET('Europe/London')  
FROM DUAL;
```

	TZ_OFFSET('US/EASTERN')	TZ_OFFSET('CANADA/YUKON')	TZ_OFFSET('EUROPE/LONDON')
1	-04:00	-07:00	+01:00

FROM_TZ

Display the `TIMESTAMP` value `'2000-03-28 08:00:00'` as a `TIMESTAMP WITH TIME ZONE` value for the `'Australia/North'` time zone region.

```
SELECT FROM_TZ(TIMESTAMP
                '2000-07-12 08:00:00', 'Australia/North')
FROM DUAL;
```

 FROM_TZ(TIMESTAMP'2000-07-1208:00:00','AUSTRALIA/NORTH')
1 12-JUL-00 08.00.00.000000000 AM AUSTRALIA/NORTH

TO_TIMESTAMP

Display the character string '2007-03-06 11:00:00'
as a TIMESTAMP value:

```
SELECT TO_TIMESTAMP ('2007-03-06 11:00:00',  
                     'YYYY-MM-DD HH:MI:SS')  
FROM DUAL;
```

```
TO_TIMESTAMP('2007-03-0611:00:00','YYYY-MM-DDHH:MI:SS')  
06-MAR-07 11.00.00.000000000
```

TO_YMINTERVAL

Display a date that is one year and two months after the hire date for the employees working in the department with the DEPARTMENT_ID 20.

```
SELECT hire_date,  
       hire_date + TO_YMINTERVAL('01-02') AS  
       HIRE_DATE_YMININTERVAL  
FROM   employees  
WHERE  department_id = 20;
```

	HIRE_DATE	HIRE_DATE_YMININTERVAL
1	17-FEB-1996 00:00:00	17-APR-1997 00:00:00
2	17-AUG-1997 00:00:00	17-OCT-1998 00:00:00

TO_DSINTERVAL

Display a date that is 100 days and 10 hours after the hire date for all the employees.

```
SELECT last_name,  
       TO_CHAR(hire_date, 'mm-dd-yy:hh:mi:ss') hire_date,  
       TO_CHAR(hire_date +  
               TO_DSINTERVAL('100 10:00:00'),  
               'mm-dd-yy:hh:mi:ss') hiredate2  
FROM employees;
```

	LAST_NAME	HIRE_DATE	HIREDATE2
1	OConnell	06-21-99:12:00:00	09-29-99:10:00:00
2	Grant	01-13-00:12:00:00	04-22-00:10:00:00
3	Whalen	09-17-87:12:00:00	12-26-87:10:00:00
4	Hartstein	02-17-96:12:00:00	05-27-96:10:00:00
5	Fay	08-17-97:12:00:00	11-25-97:10:00:00
6	Mavris	06-07-94:12:00:00	09-15-94:10:00:00
7	Baer	06-07-94:12:00:00	09-15-94:10:00:00
8	Higgins	06-07-94:12:00:00	09-15-94:10:00:00

...

Daylight Saving Time

- First Sunday in April
 - Time jumps from 01:59:59 AM to 03:00:00 AM.
 - Values from 02:00:00 AM to 02:59:59 AM are not valid.
- Last Sunday in October
 - Time jumps from 02:00:00 AM to 01:00:01 AM.
 - Values from 01:00:01 AM to 02:00:00 AM are ambiguous because they are visited twice.

Quiz

The `TIME_ZONE` session parameter may be set to:

1. A relative offset
2. Database time zone
3. OS local time zone
4. A named region

Summary

In this lesson, you should have learned how to use the following functions:

- `CURRENT_DATE`
- `CURRENT_TIMESTAMP`
- `LOCALTIMESTAMP`
- `DBTIMEZONE`
- `SESSIONTIMEZONE`
- `EXTRACT`
- `TZ_OFFSET`
- `FROM_TZ`
- `TO_TIMESTAMP`
- `TO_YMINTERVAL`
- `TO_DSINTERVAL`

Practice 5: Overview

This practice covers using the datetime functions.