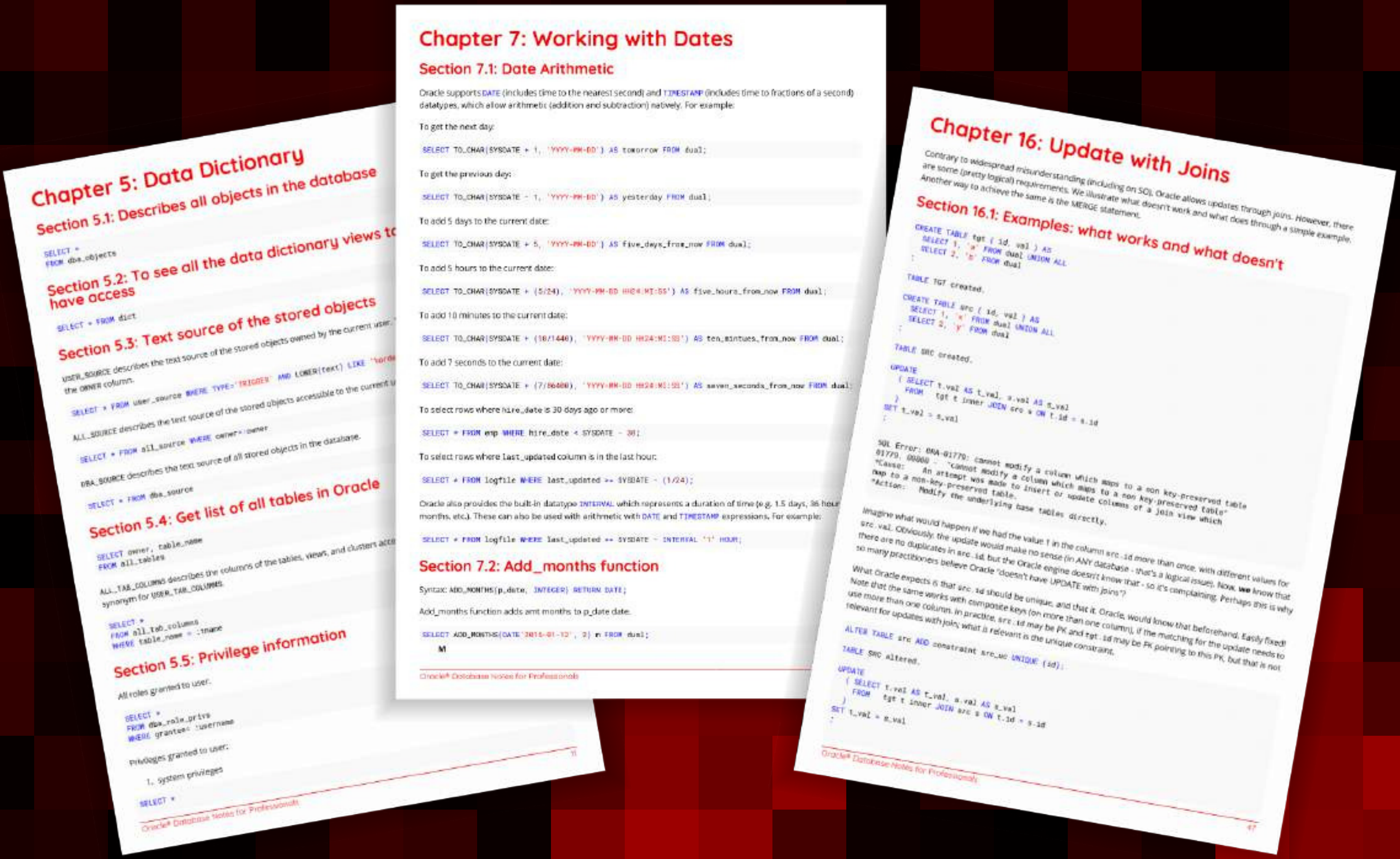
Oracle® Database

Notes for Professionals



Chapter 5: Data Dictionary

Section 5.1: Describes all objects in the database

```
SELECT *
FROM dba_objects
```

Section 5.2: To see all the data dictionary views to have access

```
SELECT * FROM dict
```

Section 5.3: Text source of the stored objects

USER_SOURCE describes the text source of the stored objects owned by the current user. This view does not display the OWNER column.

ALL_SOURCE describes the text source of the stored objects accessible to the current user.

DBA_SOURCE describes the text source of all stored objects in the database.

```
SELECT * FROM dba_source
```

Section 5.4: Get list of all tables in Oracle

```
SELECT owner, table_name
FROM all_tables
```

ALL_TAB_COLUMNS describes the columns of the tables, views, and clusters accessible

Section 5.5: Privilege information

All roles granted to user.

```
SELECT *
FROM dba_role_privs
WHERE grantee= :username
```

Privileges granted to user:

1. system privileges

Oracle® Database Notes for Professionals

Chapter 7: Working with Dates

Section 7.1: Date Arithmetic

Oracle supports DATE (includes time to the nearest second) and TIMESTAMP (includes time to fractions of a second) datatypes, which allow arithmetic (addition and subtraction) natively. For example:

To get the next day:

```
SELECT TO_CHAR(SYSDATE + 1, 'YYYY-MM-DD') AS tomorrow FROM dual;
```

To get the previous day:

```
SELECT TO_CHAR(SYSDATE - 1, 'YYYY-MM-DD') AS yesterday FROM dual;
```

To add 5 days to the current date:

```
SELECT TO_CHAR(SYSDATE + 5, 'YYYY-MM-DD') AS five_days_from_now FROM dual;
```

To add 5 hours to the current date:

```
SELECT TO_CHAR(SYSDATE + (5/24), 'YYYY-MM-DD HH24:MI:SS') AS five_hours_from_now FROM dual;
```

To add 10 minutes to the current date:

```
SELECT TO_CHAR(SYSDATE + (10/1440), 'YYYY-MM-DD HH24:MI:SS') AS ten_mintues_from_now FROM dual;
```

To add 7 seconds to the current date:

```
SELECT TO_CHAR(SYSDATE + (7/86400), 'YYYY-MM-DD HH24:MI:SS') AS seven_seconds_from_now FROM dual;
```

To select rows where hire_date is 30 days ago or more:

```
SELECT * FROM emp WHERE hire_date < SYSDATE - 30;
```

To select rows where last_updated column is in the last hour:

```
SELECT * FROM logfile WHERE last_updated >= SYSDATE - (1/24);
```

Oracle also provides the built-in datatype INTERVAL which represents a duration of time (e.g. 1.5 days, 36 hours, 2 months, etc.). These can also be used with arithmetic with DATE and TIMESTAMP expressions. For example:

```
SELECT * FROM logfile WHERE last_updated >= SYSDATE - INTERVAL '1' HOUR;
```

Section 7.2: Add_months function

Syntax: ADD_MONTHS(p_date, INTEGER) RETURN DATE;

Add_months function adds amt months to p_date date.

```
SELECT ADD_MONTHS(DATE'2015-01-12', 2) m FROM dual;
M
```

Oracle® Database Notes for Professionals

Chapter 16: Update with Joins

Contrary to widespread misunderstanding (including on SO), Oracle allows updates through joins. However, there are some (pretty logical) requirements. We illustrate what doesn't work and what does through a simple example. Another way to achieve the same is the MERGE statement.

Section 16.1: Examples: what works and what doesn't

```
CREATE TABLE tgt ( id, val ) AS
  SELECT 1, 'a' FROM dual UNION ALL
  SELECT 2, 'b' FROM dual
;

TABLE TGT created.

CREATE TABLE src ( id, val ) AS
  SELECT 1, 'x' FROM dual UNION ALL
  SELECT 2, 'y' FROM dual
;

TABLE SRC created.

UPDATE
  ( SELECT t.val AS t_val, s.val AS s_val
    FROM   tgt t inner JOIN src s ON t.id = s.id
  )
SET t_val = s_val
;
```

```
SQL Error: ORA-01779: cannot modify a column which maps to a non key-preserved table
01779. 00000 - "cannot modify a column which maps to a non key-preserved table"
*Cause:    An attempt was made to insert or update columns of a join view which
           map to a non-key-preserved table.
*Action:   Modify the underlying base tables directly.
```

Imagine what would happen if we had the value 1 in the column src.id more than once, with different values for src.val. Obviously, the update would make no sense (in ANY database - that's a logical issue). Now, we know that there are no duplicates in src.id, but the Oracle engine doesn't know that - so it's complaining. Perhaps this is why so many practitioners believe Oracle "doesn't have UPDATE with joins"?

What Oracle expects is that src.id should be unique, and that it, Oracle, would know that beforehand. Easily fixed! Note that the same works with composite keys (on more than one column), if the matching for the update needs to use more than one column. In practice, src.id may be PK and tgt.id may be FK pointing to this PK, but that is not relevant for updates with join; what is relevant is the unique constraint.

```
ALTER TABLE src ADD constraint src_uc UNIQUE (id);

TABLE SRC altered.

UPDATE
  ( SELECT t.val AS t_val, s.val AS s_val
    FROM   tgt t inner JOIN src s ON t.id = s.id
  )
SET t_val = s_val
;
```

Oracle® Database Notes for Professionals

100+ pages

of professional hints and tricks

GoalKicker.com

Free Programming Books

Disclaimer
This is an unofficial free book created for educational purposes and is not affiliated with official Oracle® Database group(s) or company(s).
All trademarks and registered trademarks are the property of their respective owners

Contents

About 1
Chapter 1: Getting started with Oracle Database 2
Section 1.1: Hello World 2
Section 1.2: SQL Query 2
Section 1.3: Hello world! from table 2
Section 1.4: Hello World from PL/SQL 3
Chapter 2: Getting started with PL/SQL 4
Section 2.1: Hello World 4
Section 2.2: Definition of PL/SQL 4
Section 2.3: Difference between %TYPE and %ROWTYPE 5
Section 2.4: Create or replace a view 6
Section 2.5: Create a table 6
Section 2.6: About PL/SQL 6
Chapter 3: Anonymous PL/SQL Block 8
Section 3.1: An example of an anonymous block 8
Chapter 4: PL/SQL procedure 9
Section 4.1: Syntax 9
Section 4.2: Hello World 9
Section 4.3: In/Out Parameters 9
Chapter 5: Data Dictionary 11
Section 5.1: Describes all objects in the database 11
Section 5.2: To see all the data dictionary views to which you have access 11
Section 5.3: Text source of the stored objects 11
Section 5.4: Get list of all tables in Oracle 11
Section 5.5: Privilege information 11
Section 5.6: Oracle version 12
Chapter 6: Dates 13
Section 6.1: Date Arithmetic - Difference between Dates in Days, Hours, Minutes and/or Seconds 13
Section 6.2: Setting the Default Date Format Model 14
Section 6.3: Date Arithmetic - Difference between Dates in Months or Years 14
Section 6.4: Extract the Year, Month, Day, Hour, Minute or Second Components of a Date 15
Section 6.5: Generating Dates with No Time Component 16
Section 6.6: Generating Dates with a Time Component 16
Section 6.7: The Format of a Date 17
Section 6.8: Converting Dates to a String 17
Section 6.9: Changing How SQL/Plus or SQL Developer Display Dates 18
Section 6.10: Time Zones and Daylight Savings Time 18
Section 6.11: Leap Seconds 19
Section 6.12: Getting the Day of the Week 19
Chapter 7: Working with Dates 20
Section 7.1: Date Arithmetic 20
Section 7.2: Add_months function 20
Chapter 8: DUAL table 22
Section 8.1: The following example returns the current operating system date and time 22
Section 8.2: The following example generates numbers between start_value and end_value 22
Chapter 9: JOINS 23

Section 9.1: CROSS JOIN 23
Section 9.2: LEFT OUTER JOIN 24
Section 9.3: RIGHT OUTER JOIN 25
Section 9.4: FULL OUTER JOIN 27
Section 9.5: ANTIJOIN 28
Section 9.6: INNER JOIN 29
Section 9.7: JOIN 30
Section 9.8: SEMIJOIN 30
Section 9.9: NATURAL JOIN 31

Chapter 10: Handling NULL values 33

Section 10.1: Operations containing NULL are NULL, except concatenation 33
Section 10.2: NVL2 to get a different result if a value is null or not 33
Section 10.3: COALESCE to return the first non-NULL value 33
Section 10.4: Columns of any data type can contain NULLs 33
Section 10.5: Empty strings are NULL 33
Section 10.6: NVL to replace null value 34

Chapter 11: String Manipulation 35

Section 11.1: INITCAP 35
Section 11.2: Regular expression 35
Section 11.3: SUBSTR 35
Section 11.4: Concatenation: Operator || or concat() function 36
Section 11.5: UPPER 36
Section 11.6: LOWER 37
Section 11.7: LTRIM / RTRIM 37

Chapter 12: IF-THEN-ELSE Statement 38

Section 12.1: IF-THEN 38
Section 12.2: IF-THEN-ELSE 38
Section 12.3: IF-THEN-ELSIF-ELSE 38

Chapter 13: Limiting the rows returned by a query (Pagination) 39

Section 13.1: Get first N rows with row limiting clause 39
Section 13.2: Get row N through M from many rows (before Oracle 12c) 39
Section 13.3: Get N numbers of Records from table 39
Section 13.4: Skipping some rows then taking some 40
Section 13.5: Skipping some rows from result 40
Section 13.6: Pagination in SQL 40

Chapter 14: Recursive Sub-Query Factoring using the WITH Clause (A.K.A. Common Table Expressions) 42

Section 14.1: Splitting a Delimited String 42
Section 14.2: A Simple Integer Generator 42

Chapter 15: Different ways to update records 44

Section 15.1: Update using Merge 44
Section 15.2: Update Syntax with example 44
Section 15.3: Update Using Inline View 44
Section 15.4: Merge with sample data 45

Chapter 16: Update with Joins 47

Section 16.1: Examples: what works and what doesn't 47

Chapter 17: Functions 49

Section 17.1: Calling Functions 49

Chapter 18: Statistical functions 50

Section 18.1: Calculating the median of a set of values 50
Chapter 19: Window Functions 51
Section 19.1: Ratio_To_Report 51
Chapter 20: Creating a Context 52
Section 20.1: Create a Context 52
Chapter 21: Splitting Delimited Strings 53
Section 21.1: Splitting Strings using a Hierarchical Query 53
Section 21.2: Splitting Strings using a PL/SQL Function 53
Section 21.3: Splitting Strings using a Recursive Sub-query Factoring Clause 54
Section 21.4: Splitting Strings using a Correlated Table Expression 55
Section 21.5: Splitting Strings using CROSS APPLY (Oracle 12c) 56
Section 21.6: Splitting Strings using XMLTable and FLWOR expressions 57
Section 21.7: Splitting Delimited Strings using XMLTable 57
Chapter 22: Collections and Records 59
Section 22.1: Use a collection as a return type for a split function 59
Chapter 23: Object Types 60
Section 23.1: Accessing stored objects 60
Section 23.2: BASE_TYPE 60
Section 23.3: MID_TYPE 61
Section 23.4: LEAF_TYPE 62
Chapter 24: Loop 64
Section 24.1: Simple Loop 64
Section 24.2: WHILE Loop 64
Section 24.3: FOR Loop 64
Chapter 25: Cursors 67
Section 25.1: Parameterized "FOR loop" Cursor 67
Section 25.2: Implicit "FOR loop" cursor 67
Section 25.3: Handling a CURSOR 67
Section 25.4: Working with SYS_REFCURSOR 68
Chapter 26: Sequences 69
Section 26.1: Creating a Sequence: Example 69
Chapter 27: Indexes 71
Section 27.1: b-tree index 71
Section 27.2: Bitmap Index 71
Section 27.3: Function Based Index 71
Chapter 28: Hints 72
Section 28.1: USE_NL 72
Section 28.2: APPEND HINT 72
Section 28.3: Parallel Hint 72
Section 28.4: USE_HASH 73
Section 28.5: FULL 73
Section 28.6: Result Cache 74
Chapter 29: Packages 75
Section 29.1: Define a Package header and body with a function 75
Section 29.2: Overloading 75
Section 29.3: Package Usage 76
Chapter 30: Exception Handling 78
Section 30.1: Syntax 78

Section 30.2: User defined exceptions ... 78
Section 30.3: Internally defined exceptions ... 79
Section 30.4: Predefined exceptions ... 80
Section 30.5: Define custom exception, raise it and see where it comes from ... 81
Section 30.6: Handling connexion error exceptions ... 82
Section 30.7: Exception handling ... 83
Chapter 31: Error logging ... 84
Section 31.1: Error logging when writing to database ... 84
Chapter 32: Database Links ... 85
Section 32.1: Creating a database link ... 85
Section 32.2: Create Database Link ... 85
Chapter 33: Table partitioning ... 87
Section 33.1: Select existing partitions ... 87
Section 33.2: Drop partition ... 87
Section 33.3: Select data from a partition ... 87
Section 33.4: Split Partition ... 87
Section 33.5: Merge Partitions ... 87
Section 33.6: Exchange a partition ... 87
Section 33.7: Hash partitioning ... 88
Section 33.8: Range partitioning ... 88
Section 33.9: List partitioning ... 88
Section 33.10: Truncate a partition ... 89
Section 33.11: Rename a partition ... 89
Section 33.12: Move partition to different tablespace ... 89
Section 33.13: Add new partition ... 89
Chapter 34: Oracle Advanced Queuing (AQ) ... 90
Section 34.1: Simple Producer/Consumer ... 90
Chapter 35: constraints ... 94
Section 35.1: Update foreign keys with new value in Oracle ... 94
Section 35.2: Disable all related foreign keys in oracle ... 94
Chapter 36: Autonomous Transactions ... 95
Section 36.1: Using autonomous transaction for logging errors ... 95
Chapter 37: Oracle MAF ... 96
Section 37.1: To get value from Binding ... 96
Section 37.2: To set value to binding ... 96
Section 37.3: To invoke a method from binding ... 96
Section 37.4: To call a javaScript function ... 96
Chapter 38: level query ... 97
Section 38.1: Generate N Number of records ... 97
Section 38.2: Few usages of Level Query ... 97
Chapter 39: Hierarchical Retrieval With Oracle Database 12C ... 98
Section 39.1: Using the CONNECT BY Caluse ... 98
Section 39.2: Specifying the Direction of the Query From the Top Down ... 98
Chapter 40: Data Pump ... 99
Section 40.1: Monitor Datapump jobs ... 99
Section 40.2: Step 3/6 : Create directory ... 99
Section 40.3: Step 7 : Export Commands ... 99
Section 40.4: Step 9 : Import Commands ... 100
Section 40.5: Datapump steps ... 101

Click here to download full PDF material