

More SQL in Oracle

CS157A

Chris Pollett

Oct. 26, 2005.

Outline

- Insert/Delete/Update
- Selects
- String, Number, Date functions
- Views
- Commit and Rollback
- Sequences
- Data Dictionary
- Types

Insert

- The basic syntax in Oracle is essentially the same as what we have already seen in the standard:
- Here are some examples:
 - insert into components values('f96', 1031, 'exam1', 100, 30);
 - insert into courses(term, lineno, cno) values ('f96', 1037, 'csc326');
- One can also do inserts using selects:
 - insert into cheap_parts
 - select * from parts where price <= 20.0;

Update

- Again, the syntax of update is very similar to the standard. Here are some examples:

```
update employee  
set salary = salary*100  
where fname='Chris';
```

```
update parts set qoh= (select max(qoh) from parts) where  
qoh < 100;
```

```
update parts set qoh = 2*qoh where 3 <= (select sum(qty)  
from odetails where odetails.pno = parts.pno);
```

Delete

- Similar once more to the standard. Here are some examples:

`delete from customers;`

`delete from zip in`

`(select zip from zipcodes where city='Fort Hays');`

Selects

- As we we've already observed Oracle does not use the keyword `AS` for aliases:
 `select E.SSN from EMPLOYEE E;`
- `AS` can be used to rename column outputs but is optional:
 `select SSN as SOCIAL from EMPLOYEE;`
- Oracle uses `MINUS` rather than `EXCEPT` for set difference.
- Oracle only supports `UNION ALL` for bag's.
- Most other aspects of the standard are supported: i.e., subselect, `IN` keyword, `EXISTS`, `op ALL`, aggregates, `GROUP BY` and `HAVING`.

String, Number, Date functions

- Recall the select line of a select-from-where query in the standard is allowed to have expressions in the standard:

```
select 1.1*salary as raise_salary from  
EMPLOYEE;
```
- This is allowed in Oracle. In addition, Oracle has strong support for functions to manipulate Strings, Numbers, and Dates.

String, Number, Date Functions

II

- String functions:
 - to_char(1234), || -- string concatenation, lpad(string, length, ['char']), rpad(string, length,['char']), ltrim(string, ['char']), rtrim(string, ['char']), lower(string), upper(string), initcap(string), length(string), substr(string, start, [n]), instr(string, 'char' [, start [,n])
- Numeric functions:
 - +, -, *, /, abs, ceil, floor, mod, power, sqrt. Aggregates: max, min, sum, avg. In addition there are least, greatest work on a list of values.
- Date functions:
 - +, -, sysdate, next_day(d, day), add_months(d, count), last_day(d), months_between(d2,d1), least(d1, d2, d3,...), greatest(d1,d2,...), trunc(d), round(d), to_char(d, format), to_date(s,format)

Views

- A view is a table that is derived from other views and base tables via a query.
- The syntax to create a view is:
`create view <viewname> as <select-statement>;`
- For example,
`create view EMP as select LNAME, FNAME, SALARY from EMPLOYEE;`
- Once created we can use a view much like any other table. For example, we can do:
`select * from EMP; -- to see rows in this view`
- To get rid of a view one does the command drop view:
`drop view EMP;`

Commit and Rollback

- Sometimes it is necessary to reverse the changes done by insert, delete, and update statements.
- Oracle has two commands for this purpose: **commit** and **rollback**.
- **commit** makes a sequence of changes done in a session or transaction permanent.
- **rollback** undoes a sequence of changes done since the last commit or the start of session or transaction.
- Some operations cause an auto-commit to be done: exit a session, create table, drop table, alter table

Sequences

- A **sequence** is an object that consists of an integer value initialized to a particular value at its definition time and which permits a next value operation.
- In Oracle one can create a sequence with:
create sequence custseq start with 1000;
- It could then be used in statements like insert:
insert into customers values (custseq.nextval, 'Jones',
 '123 Main St.', 67226, '111-11-1111');
- One can also specify in the definition of a sequence an **increment by** clause, **maxvalue** and **minvalue** clauses, and a **cycle/no cycle** clause

Data Dictionary

- Here are some useful tables and columns from Oracle's data dictionary:
 - `user_catalog(table_name, table_type)` -- can be abbreviated `cat`
 - `user_objects(object_name, object_id, object_type, created, last_ddl_time, timestamp, status)`
 - `user_tables` -- can list its columns with `describe user_tables` --it can also be abbreviated `tab`
 - `user_tab_columns` -- has statistical information about columns of each table
 - `user_views(view_name, text_length, text)`

Types

- Oracle allows you to define new datatypes:

```
create type address_type as object(  
    street varchar2(30),  
    city varchar2(30),  
    state varchar2(20),  
    zip number(5)  
);
```

- This type could then be used for instance if one wanted to create a table:

```
create table name_address (name varchar2(10), address  
    address_type);
```