1. **VARCHAR VS Varchar2**

**VARCHAR** is reserved by Oracle to support distinction between NULL and empty string in future, as ANSI standard prescribes. **VARCHAR2** does not distinguish between a NULL and empty string, and never will. If you rely on empty string and NULL being the same thing, you should use **VARCHAR2** .Jul 23, 2009

1. Can I use BOOLEAN in SELECT statement

|  |  |
| --- | --- |
| 17down vote | From [documentation](http://docs.oracle.com/cd/B19306_01/appdev.102/b14261/datatypes.htm#CJACJGBG):  You cannot insert the values TRUE and FALSE into a database column. You cannot select or fetch column values into a BOOLEAN variable. Functions called from a SQL query cannot take any BOOLEAN parameters. Neither can built-in SQL functions such as TO\_CHAR; to represent BOOLEAN values in output, you must use IF-THEN or CASE constructs to translate BOOLEANvalues into some other type, such as 0 or 1, 'Y' or 'N', 'true' or 'false', and so on. |

The BOOLEAN data type is a PL/SQL data type. Oracle does not provide an equivalent SQL data type (...) you can create a wrapper function which maps a SQL type to the BOOLEAN type.

Check this: <http://forums.datadirect.com/ddforums/thread.jspa?threadID=1771&tstart=0&messageID=5284>

1. Purpose of DUAL

The **DUAL** table is a special one-row, one-column [table](https://en.wikipedia.org/wiki/Table_(database)) present by default in [Oracle](https://en.wikipedia.org/wiki/Oracle_database) and other [database](https://en.wikipedia.org/wiki/Relational_database) installations

Oracle's [SQL](https://en.wikipedia.org/wiki/SQL) syntax requires the [FROM](https://en.wikipedia.org/wiki/From_(SQL)) clause but some queries don't require any tables - DUAL can be readily used in these cases.

**SELECT** 1+1

**from** dual;

1. IF NOT EXISTS In Function not present in ORACLE – alternatives

Exists condition can be used only in SQL statement, it cannot be used directly in PL/SQL. There are several options:

* Using case expression with exists condition inside a select statement:
* SQL> declare
* 2 l\_exists number(1);
* 3 begin
* 4 select case
* 5 when exists(select 1
* 6 from employees
* 7 where department\_id = 1)
* 8 then 1
* 9 else 0
* 10 end into l\_exists
* 11 from dual;
* 12
* 13 if (l\_exists = 1)
* 14 then
* 15 dbms\_output.put\_line('exists');
* 16 else
* 17 dbms\_output.put\_line(q'[doesn't exist]');
* 18 end if;
* 19 end;
* 20 /
* doesn't exist

PL/SQL procedure successfully completed

1. USER\_TABLES vs ALL\_TABLES vs DBA\_TABLES

The information available in data dictionary tables is very difficult to understand. So it provides views in a form that is easily  understood by users. All these views are owned by sys.  
  
Oracle provides three different type of views.  
  
USER\_XXXX -> Objects owned by the user.  
ALL\_XXXX   -> List of objects that user has access to  
DBA\_XXXX -> List of all objects available in database.

USER\_TABLES is tables which you own  
ALL\_TABLES is tables which own, and tables owner by other users, which you have been granted excplicit access to  
DBA\_TABLES is all tables in the database