

SQL

The **SELECT** statement

Orlando Karam
okaram@spsu.edu

SELECT

- SELECT fields or expressions (or *)
- FROM table
- WHERE conditions
- ORDER BY fields or expressions

- DISTINCT vs ALL
- Oracle - Dual

Fields and Expressions

- `SELECT *`
- `SELECT id, name`
- `SELECT id, age+5 AS "Age in 5 Years"`
 - ‘ vs “ (and autocorrect)
- `SELECT DISTINCT Age, Gender`

Numeric expressions

- The usual suspects (+, -, *, /)
- Numeric functions
 - abs, sign
 - sin, cos ...
 - ceil, floor, round, trunc
 - exp, ln, log
 - power
 - remainder, mod
 - width_bucket

String expressions

- Remember, 'single quotes'
- || for concatenation
- Useful Functions (tested in Oracle):
 - Lower, Upper, InitCap
 - LTrim, RTrim, Trim, LPad, RPad
 - Instr, Substr, Replace, Translate
 - Regexp_replace, regexp_substr, regexp_instr
 - Length

Date/Time expressions (Oracle)

- Constants can be strings, with current locale
 - 'Jan-1-2010'
- CURRENT_DATE, CURRENT_TIMESTAMP
- NumToYMInterval, NumToDSInterval
- To_Char, extract
 - extract(year from current_date)
- round, trunc
- add_months, months_between

- Takes a variable number of arguments, returns the first of them which is not null
 - Oracle defines nvl, nvl2, nullif, which are similar

CASE expressions

- Like a switch (if/else if...), but returning a value
- Syntax looks like a statement
- CASE gender WHEN 'M' THEN 'Male' WHEN 'F' THEN 'Female' ELSE 'Unknown' END
- CASE WHEN age<12 THEN 'Kid' WHEN Age>12 AND Age<20 'Teenager' ELSE 'Adult' END

- Table or View
- Aliasing (write an alias after the table name)
- Can select from more than one table (will see with joins)
- Example:
 - SELECT *
 - FROM Person P, Emails E

Conditions

- Comparison operators
 - = != > < <> >= etc
 - Any, some, all
- Combine conditions
 - AND, OR, NOT (Unknown, NULLs)
- IS NULL / IS NOT NULL
- IS NAN, IS INFINITE
- LIKE / NOT LIKE REGEXP_LIKE
- BETWEEN
- IN, EXISTS (subqueries explained later)