# MySQL Differences with Oracle

# Documentation

Reference Manual:

<https://dev.mysql.com/doc/refman/8.0/en/>

# Tools

## Client Tools mysql (SQL\*Plus) and MySQLWorkbench ( SQL\*Developer)

Overview of all programs : <https://dev.mysql.com/doc/refman/8.0/en/programs-overview.html>'

mysql is a command line client very similar to sqlplus

MySQLWorkbench looks like Windows-only and similar to SQL\*Developer

**Mysqld – is database engine i.e. MySQL Server**

# mysql (i.e. SQL\*Plus)

## Help

To see mysql help:

|  |
| --- |
| mysql> help;  For information about MySQL products and services, visit:  http://www.mysql.com/  For developer information, including the MySQL Reference Manual, visit:  http://dev.mysql.com/  To buy MySQL Enterprise support, training, or other products, visit:  https://shop.mysql.com/  List of all MySQL commands:  Note that all text commands must be first on line and end with ';'  ? (\?) Synonym for `help'.  clear (\c) Clear the current input statement.  connect (\r) Reconnect to the server. Optional arguments are db and host.  delimiter (\d) Set statement delimiter.  ego (\G) Send command to mysql server, display result vertically.  exit (\q) Exit mysql. Same as quit.  go (\g) Send command to mysql server.  help (\h) Display this help.  notee (\t) Don't write into outfile.  print (\p) Print current command.  prompt (\R) Change your mysql prompt.  quit (\q) Quit mysql.  rehash (\#) Rebuild completion hash.  source (\.) Execute an SQL script file. Takes a file name as an argument.  status (\s) Get status information from the server.  system (\!) Execute a system shell command.  tee (\T) Set outfile [to\_outfile]. Append everything into given outfile.  use (\u) Use another database. Takes database name as argument.  charset (\C) Switch to another charset. Might be needed for processing binlog with multi-byte charsets.  warnings (\W) Show warnings after every statement.  nowarning (\w) Don't show warnings after every statement.  resetconnection(\x) Clean session context. |

## System (i.e. host)

Executes OS command

|  |
| --- |
| For server side help, type 'help contents'  mysql> system dir;  Volume in drive C is OS  Volume Serial Number is 0867-9728  Directory of C:\Program Files\MySQL\MySQL Server 8.0\bin |

## Source (i.e. @)

### From inside mysql

**Note "/" for Windows!**

source C:/git/mysql/commands/bind\_var.sql;

### From Command Line

Use redirect:

|  |
| --- |
| pushd "C:\Program Files\MySQL\MySQL Server 8.0\bin\"  mysql -u root -p join\_us < C:/git/mysql/commands/bind\_var.sql |

## Tee ( i.e. spool)

**Note "/" for Windows!**

|  |
| --- |
| tee C:/git/mysql/commands/spool.log;  source C:/git/mysql/commands/bind\_var.sql;  notee; |

Generates spool.log:

|  |
| --- |
| mysql> tee C:/git/mysql/commands/spool.log;source C:/git/mysql/commands/bind\_var.sql;notee;  Logging to file 'C:/git/mysql/commands/spool.log'  Query OK, 0 rows affected (0.00 sec)  +------------+  | @num\_users |  +------------+  | 0 |  +------------+  1 row in set (0.00 sec)  Query OK, 1 row affected (0.00 sec)  +------------+  | @num\_users |  +------------+  | 1517 |  +------------+  1 row in set (0.00 sec)  Outfile disabled. |

## Delimiters

## Scripts Error Handling

# STRICT MODE and Warnings

# SQL

## Key Minuses

* No FULL OUTER JOIN
* No simple data arithmetic

## Key Pluses

* Easy use of Boolean

select A=B; -- returns 0 or 1

* aSasA

## Comments

## Bind Variables

* Bind variable are prefixed by @
* SELECT INTO … @bind\_var does not have to be inside begin … end

|  |
| --- |
| set @num\_users := 0;  select @num\_users;  select count(\*) into @num\_users from users;  select @num\_users; |

## Issue with Parentheses

## Quotation

### Data

### Metadata

## Transactional Control

## IF EXISTS and IF NOT EXISTS

## LIMIT ( i.e ROWNUM )

# INFORMATION\_SCHEMA.(i.e. Data Dictionary)

Looks like data dictionary views equivalent to are ALL\_TABLES, ALL\_VIEWS and etc. listed in INFORMATION\_SCHEMA.tables

|  |
| --- |
| SELECT \* FROM INFORMATION\_SCHEMA.tables where table\_schema = 'INFORMATION\_SCHEMA'; |

For example this show all triggers in the current database

|  |
| --- |
| SELECT \* FROM INFORMATION\_SCHEMA.triggers t where t.EVENT\_OBJECT\_SCHEMA = database() ; |

# Databases (i.e. Schemas)

## Crete

## Drop

## Use Database

## Show Current Database

# NULLS

## Empty String IS NOT NULL

## IFNULL (i.e. NVL)

# DUAL (i.e. One Row SQL)

# INSERT

## INSERT SET

## INSERT IGNORE

# Aggregate Functions

# AUTO INCREMENT (i.e. Sequences )

# Primary/Unique Keys on Strings

# Boolean

Boolean expressions

# Dates

## Datatypes

## Functions

## Sysdate() VS. NOW()

## Casting

## Date Math

# Strings

Case Insensitive Comparison

## Strings Datatypes

## Strings Functions

# Compound Statements (i.e. PL/SQL)

Compound statements provide a procedural language somewhat similar to PL/SQL

## Key Limitations:

* No packages and package variables
* No DBMS\_OUTPUT equivalent
* No anonymous PL/SQL Blocks
* Signal Message cannot be an expression ( must be either variable or coenstant)

## SQLSTATE (i.e. SQLCODE)

SQLSTATE [VALUE] ***sqlstate\_value***: A 5-character string literal indicating an SQLSTATE value.

Similar to use to Oracle SQLCODE

## Signal and Resignal (i.e. RAISE and RAISE\_APPLICATION\_ERROR)

## Declare Inside

DECLARE commands are INSIDE begin… end

## Value Assignment

To assign value use

SET var := expression

## Exception handling

More info here:

<https://dev.mysql.com/doc/refman/8.0/en/declare-handler.html>

Exceptions are handled in declared handlers i.e.

|  |
| --- |
| DECLARE *handler\_action* HANDLER  FOR *condition\_value* [, *condition\_value*] ...  *statement*  *handler\_action*: {  CONTINUE  | EXIT  | UNDO  }  *condition\_value*: {  *mysql\_error\_code*  | SQLSTATE [VALUE] *sqlstate\_value*  | *condition\_name*  | SQLWARNING  | NOT FOUND  | SQLEXCEPTION  } |

For example:

## Triggers