

SQL Commands Cheat Sheet		
SELECT	* / TOP / Field / DISTINCT Field	AGGREGATE COUNT SUM AVG (Average) MIN MAX Distinct - no duplicates
	MIN() / MAX() / AVG() / SUM (Field)	
	COUNT (*) / COUNT (DISTINCT Field)	
	ROUND (AVG (Field), 2)	Round(Field, # of decimal places)
	Field1 ', ' Field2 AS "NewFieldName"	Concatenate/ Join Fields
	Field + / - / * / / / x	Arithmetic
FROM	TABLES_name	Always Plural
INNER/OUTER LEFT/ RIGHT JOIN	Select t1.Field1, t2.Field1, t1.Field2 FROM Table1 t1 INNER JOIN Table2 t2 ON t1.PK = t2.FK	(INNER) JOIN: ALL match in BOTH LEFT (OUTER) JOIN: ALL records from left table that match right RIGHT (OUTER) JOIN: ALL records from right table that match left FULL JOIN: ALL records with match EITHER from left or right
WHERE // select individual records 'r%' starts with r '%r' ends with r '%r%' has r in middle 'r%' starts with r and ends with c % zero or more characters _ single character [] single character within [] ^ or ! DOES NOT - any char w/in range	Field = "Value"; NOT Field = "Value"; Field LIKE 'r%' / '%r' / '%r%' / 'r%c' Field LIKE '_r%' / '[rcs]%' / '[r-z]%' / '![abc]%' Field NOT LIKE 'a%' Field = "Value" AND / OR Field = "Value"; Field BETWEEN VAL1 AND VAL2 Field NOT BETWEEN VAL1 AND VAL2 Length (Field) >/</<=>/>= /<= x; Field IN / NOT IN (x,y,z) Field IS NULL	WHERE Predicates using OPERATORS *** use with numeric or alphabetic order = equals <> does not equal > is greater than < is less than >= greater than or equal to <= is less than or equal to WHERE Predicates using BOOLEAN AND OR IN (x,y,z) NOT BETWEEN...AND IS NULL
GROUP BY	Field	
HAVING	AVG (Salary) >= 5000 ONLY with AGGREGATES because WHERE doesn't apply to aggregates	Always before ORDER BY // select at a group level
ORDER BY	COUNT(Field) ASC / DESC or Field ASC / DESC	Sorting NVL -replace NULL =0

CONSTRAINTS	CREATE & DELETE DATABASE DDL	ALTER DATABASE DDL
NOT NULL DEFAULT UNIQUE Primary Key (NOT NULL, UNIQUE) Foreign Key (Links data between 2 tables)	CREATE DATABASE database_name; DROP DATABASE database_name;	ALTER DATABASE database_name MODIFY NAME = new_db_name;
CREATE TABLE DDL	ALTER TABLE DDL	INSERT DATA DML
CREATE TABLE table_name (Field_name data_type (size), Person_ID int, FirstName varchar(255), LastName varchar(255));	ALTER TABLE table_name [ADD/ MODIFY/ DROP] Field Field_name data_type [num/ char/date];	INSERT INTO table_name (Field1, Field2) VALUES (value1, value2);
UPDATE DATA DML	DELETE DATA IN TABLE DDL	DELETE TABLE DML
UPDATE table_name SET Field_name = value WHERE conditions;	TRUNCATE TABLE table_name;	DROP TABLE table_name;

DML - Data Manipulation Language - Retrieve, Update, add and delete DATA in database // SELECT, INSERT, UPDATE, DELETE
 DDL - Data Definition Language - Create and Modify database // CREATE, DROP, ALTER, TRUNCATE
 DCL - Data Control Language - maintains security in the database // GRANT, REVOKE

SQL Commands Cheat Sheet		
SELECT	* / TOP / Field / DISTINCT Field	AGGREGATTE COUNT SUM AVG (Average) MIN MAX Distinct - no duplicates Round(Field, # of decimal places) Concatenate/ Join Fields Arithmetic
	MIN() / MAX() / AVG() / SUM (Field)	
	COUNT (*) / COUNT (DISTINCT Field)	
	ROUND (AVG (Field), 2)	
	Field1 ', ' Field2 AS "NewFieldName"	
	Field + / - / * / / / x	
FROM	TABLE S _name	Always Plural
INNER/OUTER LEFT/ RIGHT JOIN	Select t1.Field1, t2.Field1, t1.Field2 FROM Table1 t1 INNER JOIN Table2 t2 ON t1.PK = t2.FK	(INNER) JOIN : ALL match in BOTH LEFT (OUTER) JOIN : ALL records from left table that match right RIGHT (OUTER) JOIN : ALL records from right table that match left FULL JOIN : ALL records with match EITHER from left or right
WHERE // select individual records 'r%' starts with r '%r' ends with r '%r%' has rin middle 'r%' starts with r and ends with c % zero or more characters _ single character [] single character within [] ^ or ! DOES NOT - any char w/in range	Field = "Value"; NOT Field = "Value"; Field LIKE 'r%' / '%r' / '%r%' / 'r%c' Field LIKE '_r%' / '[rcs]%' / '[r-z]%' / '[!abc]%' Field NOT LIKE 'a%' Field = "Value" AND / OR Field = "Value"; Field BETWEEN VAL1 AND VAL2 Field NOT BETWEEN VAL1 AND VAL2 Length (Field) > / < / <= / = / >= / <= x; Field IN / NOT IN (x,y,z) Field IS NULL	WHERE Predicates using OPERATORS *** use with numeric or alphabetic order = equals <> does not equal > is greater than < is less than >= greater than or equal to <= is less than or equal to WHERE Predicates using BOOLEAN AND OR IN (x,y,z) NOT BETWEEN...AND IS NULL
GROUP BY	Field	
HAVING	AVG (Salary) >= 5000 ONLY with AGGREGATES because WHERE doesn't apply to aggregates	Always before ORDER BY // select at a group level
ORDER BY	COUNT(Field) ASC / DESC or Field ASC / DESC	Sorting NVL -replace NULL =0

CONSTRAINTS	CREATE & DELETE DATABASE DDL	ALTER DATABASE DDL
NOT NULL DEFAULT UNIQUE Primary Key (NOT NULL, UNIQUE) Foreign Key (Links data between 2 tables)	CREATE DATABASE database_name; DROP DATABASE database_name;	ALTER DATABASE database_name MODIFY NAME = new_db_name;
CREATE TABLE DDL	ALTER TABLE DDL	INSERT DATA DML
CREATE TABLE table_name (Field_name data_type (size), Person_ID int, FirstName varchar(255), LastName varchar(255));	ALTER TABLE table_name [ADD/ MODIFY/ DROP] Field Field_name data_type [num/ char/date] ;	INSERT INTO table_name (Field1, Field2) VALUES (value1, value2);
UPDATE DATA DML	DELETE DATA IN TABLE DDL	DELETE TABLE DML
UPDATE table_name SET Field_name = value WHERE conditions;	TRUNCATE TABLE table_name;	DROP TABLE table_name;

DML - Data Manipulation Language - Retrieve, Update, add and delete DATA in database // **SELECT, INSERT, UPDATE, DELETE**

DDL - Data Definition Language - Create and Modify database // **CREATE, DROP, ALTER, TRUNCATE**

DCL - Data Control Language - maintains security in the database // **GRANT, REVOKE**