Data Definition Language Quick Reference

Contents

[**Creating a TABLE** 1](#_Toc82092648)

[**Creating a table with CONSTRAINTS** 2](#_Toc82092649)

[**Creating a PRIMARY KEY (single column)** 3](#_Toc82092650)

[**Creating a PRIMARY KEY (composite)** 3](#_Toc82092651)

[**Creating a FOREIGN KEY** 4](#_Toc82092652)

[**How to ALTER table after it’s built** 4](#_Toc82092653)

[**How to delete (DROP) a table** 5](#_Toc82092654)

[**How to purge (TRUNCATE) a table** 5](#_Toc82092655)

[**Speed up queries using INDEXES** 5](#_Toc82092656)

[**Create auto-incrementing IDs with a SEQUENCE** 6](#_Toc82092657)

[**More resources** 6](#_Toc82092658)

# **Creating a TABLE**

|  |  |
| --- | --- |
| **Syntax** | CREATE TABLE table\_name (     column1 datatype,     column2 datatype,     column3 datatype,    .... ); |
| **Example** | CREATE TABLE vendors  (  vendor\_id NUMBER,  vendor\_name VARCHAR(50)  ); |
| **More info** | <https://www.w3schools.com/sql/sql_create_table.asp> |

# **Creating a table with CONSTRAINTS**

|  |  |
| --- | --- |
| **Syntax** | CREATE TABLE table\_name (     column1 datatype *constraint*,     column2 datatype *constraint*,     column3 datatype *constraint*,     .... ); |
| **Example** | --Example 1 – Defining NOT NULL, UNIQUE, DEFAULT  CREATE TABLE invoices  (  invoice\_id NUMBER NOT NULL UNIQUE,  vendor\_id NUMBER NOT NULL,  invoice\_number VARCHAR2(50) NOT NULL,  invoice\_date DATE DEFAULT SYSDATE,  invoice\_total NUMBER(9,2) NOT NULL,  payment\_total NUMBER(9,2) DEFAULT 0  );  --Example 2a – Defining a CHECK constraint at column-level  CREATE TABLE invoices  (  invoice\_id NUMBER PRIMARY KEY,  invoice\_total NUMBER(9,2) NOT NULL CHECK (invoice\_total >= 0),  payment\_total NUMBER(9,2) DEFAULT 0 CHECK (payment\_total >= 0)  );    --Example 2a – Defining a CHECK constraint at table-level  CREATE TABLE invoices  (  invoice\_id NUMBER PRIMARY KEY,  invoice\_total NUMBER(9,2) NOT NULL,  payment\_total NUMBER(9,2) DEFAULT 0,  CONSTRAINT invoices\_ck CHECK (invoice\_total >= 0 AND payment\_total >= 0)  ); |
| **More info** | <https://www.w3schools.com/sql/sql_constraints.asp>  The following constraints are commonly used in SQL:   * [NOT NULL](https://www.w3schools.com/sql/sql_notnull.asp) - Ensures that a column cannot have a NULL value * [UNIQUE](https://www.w3schools.com/sql/sql_unique.asp) - Ensures that all values in a column are different * [PRIMARY KEY](https://www.w3schools.com/sql/sql_primarykey.asp) - A combination of a NOT NULL and UNIQUE. Uniquely identifies each row in a table * [FOREIGN KEY](https://www.w3schools.com/sql/sql_foreignkey.asp) - Prevents actions that would destroy links between tables * [CHECK](https://www.w3schools.com/sql/sql_check.asp) - Ensures that the values in a column satisfies a specific condition * [DEFAULT](https://www.w3schools.com/sql/sql_default.asp) - Sets a default value for a column if no value is specified |

# **Creating a PRIMARY KEY (single column)**

|  |  |
| --- | --- |
| **Syntax** | Syntax-wise, there are multiplate ways to define a primary key in SQL. Most common is at a column level or a table-level. There is NO different or advantage to either except your preference. At a table level you use the syntax: **CONSTRAINT *constraint\_name* PRIMARY KEY (column name)** |
| **Example** | **--Option 1 – Defining PK at column-level**  CREATE TABLE vendors  (  vendor\_id NUMBER PRIMARY KEY,  vendor\_name VARCHAR2(50) NOT NULL UNIQUE  );  **--Option 2 – Defining PK at table-level**  CREATE TABLE vendors  (  vendor\_id NUMBER,  vendor\_name VARCHAR2(50) NOT NULL,  CONSTRAINT vendors\_pk PRIMARY KEY (vendor\_id),  CONSTRAINT vendor\_name\_unq UNIQUE (vendor\_name)  ); |
| **More info** | <https://www.w3schools.com/sql/sql_primarykey.asp> |

# **Creating a PRIMARY KEY (composite)**

|  |  |
| --- | --- |
| **Syntax** | To define a composite PK that uses 2 or more columns, you have to define it a table level like below. At a table level you use the syntax: **CONSTRAINT *constraint\_name* PRIMARY KEY (column 1 name, column 2 name, …)** |
| **Example** | CREATE TABLE invoice\_line\_items  (  invoice\_id NUMBER NOT NULL,  invoice\_sequence NUMBER NOT NULL,  line\_item\_description VARCHAR2(100) NOT NULL,  CONSTRAINT line\_items\_pk PRIMARY KEY (invoice\_id, invoice\_sequence)  ); |
| **More info** | <https://www.w3schools.com/sql/sql_primarykey.asp> |

# **Creating a FOREIGN KEY**

|  |  |
| --- | --- |
| **Syntax** | Syntax-wise, there are multiplate ways to define a primary key in SQL. Most common is at a column level or a table-level. There is NO different or advantage to either except your preference. At a table level you use the syntax: **CONSTRAINT *constraint\_name* FOREIGN KEY (column name)** |
| **Example** | --For this example create a primary key table called VENDOR  CREATE TABLE vendors  (  vendor\_id NUMBER PRIMARY KEY,  vendor\_name VARCHAR2(50) NOT NULL UNIQUE  );  --Option 1: FK defined at column level  CREATE TABLE invoices  (  invoice\_id NUMBER PRIMARY KEY,  vendor\_id NUMBER REFERENCES vendors (vendor\_id),  invoice\_number VARCHAR2(50) NOT NULL UNIQUE  );  --Option 2: FK defined at table level  CREATE TABLE invoices  (  invoice\_id NUMBER PRIMARY KEY,  vendor\_id NUMBER NOT NULL,  invoice\_number VARCHAR2(50) NOT NULL UNIQUE,  CONSTRAINT invoices\_fk\_vendors FOREIGN KEY (vendor\_id) REFERENCES vendors (vendor\_id)  ); |
| **More info** | <https://www.w3schools.com/sql/sql_foreignkey.asp> |

# **How to ALTER table after it’s built**

|  |  |
| --- | --- |
| **Syntax** | --Syntax to add, change, or remove a column  ALTER TABLE table\_name ADD/MODIFY/DROP column\_name datatype;  --You can also alter a table to add, modify, or drop a constraint  ALTER TABLE table\_name ADD/MODIFY/DROP constraint constraint\_name; |
| **Example** | --Adds a column to VENDORS  ALTER TABLE vendors  ADD last\_transaction\_date DATE;  --Removes a column to VENDORS  ALTER TABLE vendors  DROP COLUMN last\_transaction\_date;  --Changes datatype of column on VENDORS  ALTER TABLE vendors  MODIFY vendor\_name VARCHAR2(100); |
| **More info** | <https://www.w3schools.com/sql/sql_alter.asp> |

# **How to delete (DROP) a table**

|  |  |
| --- | --- |
| **Syntax** | DROP TABLE *table\_name*; |
| **Example** | DROP TABLE invoice;  DROP TABLE vendor;  TRUNCATE TABLE invoice;  TRUNCATE TABLE vendor; |
| **More info** | <https://www.w3schools.com/sql/sql_drop_table.asp> |

# **How to purge (TRUNCATE) a table**

|  |  |
| --- | --- |
| **Syntax** | TRUNCATE TABLE *table\_name*; |
| **Example** | TRUNCATE TABLE invoice;  TRUNCATE TABLE vendor; |
| **More info** | <https://www.w3schools.com/sql/sql_drop_table.asp> |

# **Speed up queries using INDEXES**

|  |  |
| --- | --- |
| **Syntax** | CREATE INDEX index\_name ON table\_name (column1, column2, ...); |
| **Example** | --Example 1: Create an index on VENDORS table for VENDOR\_PHONE column  CREATE INDEX vendors\_vendor\_phone\_ix  ON vendors (vendor\_phone);  --Example 2: Create a descending index on invoice\_total which makes the more recently create invoices return quicker than older invoices  CREATE INDEX invoices\_invoice\_total\_ix  ON invoices (invoice\_total DESC); |
| **More info** | Best practice is to give indexes a clear naming standard like *table\_column\_ix*. See above.  <https://www.w3schools.com/sql/sql_create_index.asp> |

# **Create auto-incrementing IDs with a SEQUENCE**

|  |  |
| --- | --- |
| **Syntax** | CREATE SEQUENCE *sequence\_name* MINVALUE *min\_value\_allowed*  MAXVALUE max*\_value\_allowed*  START WITH *starting\_value*  INCREMENT BY *number\_to\_increment\_by* CACHE *number\_to\_cache\_in\_server\_memory*; |
| **Example** | --Create a sequence that by default starts at and increments by 1  CREATE SEQUENCE vendor\_id\_seq;  -- use a squence as default value for the column  CREATE TABLE vendors  (  vendor\_id NUMBER DEFAULT vendor\_id\_seq.NEXTVAL PRIMARY KEY,  vendor\_name VARCHAR2(50) NOT NULL UNIQUE  ); |
| **More info** | The first line of syntax is the only required line. The remaining settings are all optional. Also, syntax to create auto-incrementing numbers is different on MySQL vs SQL Server. See more here: <https://www.w3schools.com/sql/sql_autoincrement.asp> |

# **More resources**

W3 Schools - <https://www.w3schools.com/sql/>

Oracle Documentation

* Managing Table - <https://docs.oracle.com/cd/B28359_01/server.111/b28310/tables003.htm#ADMIN11634>
* Constraints - <https://docs.oracle.com/cd/B10501_01/appdev.920/a96590/adg05itg.htm>