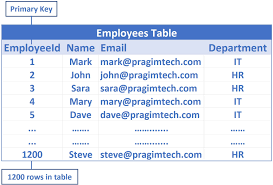
**Data Persistence (Databases):**

*Types;*

**SQL (Structural Query Language) DB:**

* They store data in a table form
* Example: 

**NoSQL DB:**

* They are pretty flexible as we can change the structure of the data without changing the entire database.

**SQL:**

**To create a table and insert data:**

* **Note:** <https://www.w3schools.com/sql/> is a guide for SQL where you can see everything related to SQL for reference to create a code.
* All the SQL commands are provided in the above table.
* https://sqliteonline.com/#fiddle-5bbdbaef7288bo2ajn2wly03 is a link for us to try SQL programs.

**Project on joining the order number with the name:**

**Code:**

-- create customer's table

CREATE TABLE customers (

id INT,

first\_name STRING,

last\_name STRING,

address STRING,

PRIMARY KEY (id)

);

-- insert some values

INSERT INTO customers VALUES (1, 'John', 'Doe', '32 Cherry Blvd');

INSERT INTO customers VALUES (2, 'Angela', 'Yu', '12 Sunset Drive');

-- fetch some values

SELECT \* FROM customers WHERE first\_name = 'John';

-- create the products table

CREATE TABLE products (

id INT NOT NULL,

name STRING,

price MONEY,

PRIMARY KEY (id)

);

--insert a product into products table

INSERT INTO products VALUES (1, 'Pen', 1.20);

-- select a product with id = 1

SELECT \* FROM products WHERE id = 1;

--insert a product with no price into products table

INSERT INTO products (id, name) VALUES (2, 'Pencil');

--select a product with id = 2

SELECT \* FROM products WHERE id = 2;

--update a row in products table

UPDATE products SET price=0.8 WHERE id = 2;

--select all records from products table

SELECT \* FROM products;

--add a new column - stock to the products table

ALTER TABLE products ADD stock INT;

--update a row in products table

UPDATE products SET stock=32 WHERE id = 1;

--select all records from products table

SELECT \* FROM products;

--update a row in products table

UPDATE products SET stock=12 WHERE id = 2;

--select all records from products table

SELECT \* FROM products;

--delete a row from products table

DELETE FROM products WHERE id = 2;

--insert back the deleted row into products table

INSERT INTO products VALUES (2, 'Pencil', 0.8, 12);

-- create the orders table

CREATE TABLE orders (

id INT NOT NULL,

order\_number INT,

customer\_id INT,

product\_id INT,

PRIMARY KEY (id),

FOREIGN KEY (customer\_id) REFERENCES customers(id),

FOREIGN KEY (product\_id) REFERENCES products(id)

);

--insert an order into orders table

INSERT INTO orders VALUES (1, 4362, 2, 1);

--join orders and customers table

SELECT orders.order\_number, customers.first\_name, customers.last\_name, customers.address

FROM orders

INNER JOIN customers ON orders.customer\_id = customers.id;

--join orders and products table

SELECT orders.order\_number, products.name, products.price, products.stock

FROM orders

INNER JOIN products ON product\_id = products.id;

**Output:**

