**Function: init\_db() – Creates the Invoice Database**

**🔹 Step 1: Connect to the Database**

* sqlite3.connect('invoice.db')
  + If invoice.db doesn't exist, it will **create a new database file**.
  + If it exists, it will **connect to the existing one**.
* cursor = conn.cursor()
  + Creates a **cursor object** to send SQL commands to the database.

**🔹 Step 2: Create customers Table**

sql

Copy code

CREATE TABLE IF NOT EXISTS customers (

id INTEGER PRIMARY KEY AUTOINCREMENT,

name TEXT NOT NULL,

contact TEXT,

address TEXT

)

* id: Auto-incremented customer ID (1, 2, 3…)
* name: Customer’s name (required)
* contact: Customer's phone/email
* address: Customer’s address

**🔹 Step 3: Create invoices Table**

sql

Copy code

CREATE TABLE IF NOT EXISTS invoices (

id INTEGER PRIMARY KEY AUTOINCREMENT,

customer\_id INTEGER,

date TEXT,

total\_amount REAL,

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

)

* id: Unique invoice ID
* customer\_id: Links the invoice to a customer (foreign key)
* date: Date/time of invoice creation
* total\_amount: Final bill total

**🔹 Step 4: Create items Table**

sql

Copy code

CREATE TABLE IF NOT EXISTS items (

id INTEGER PRIMARY KEY AUTOINCREMENT,

invoice\_id INTEGER,

item\_name TEXT,

qty INTEGER,

price REAL,

total REAL,

FOREIGN KEY(invoice\_id) REFERENCES invoices(id)

)

* id: Unique item ID
* invoice\_id: Links item to a specific invoice (foreign key)
* item\_name: Name of the product
* qty: Quantity bought
* price: Price per unit
* total: qty × price (calculated when adding item)

**🔹 Step 5: Save & Close**

python

Copy code

conn.commit()

conn.close()

* conn.commit(): Saves all changes permanently into invoice.db
* conn.close(): Closes the connection to free up memory

**🧠 In Simple Words:**

This function sets up the **entire structure** of your billing system by:

* Creating 3 tables (customers, invoices, items)
* Linking them using **foreign keys**
* Preparing the app to store real invoice data