Explanation of the Code in Simple Steps (main.py)

This Python script integrates **webcam image capture, face detection, database storage, and a Streamlit web app**. Below is a **step-by-step breakdown** of how the code works:

---

1. Importing Required Libraries

The script imports several essential libraries:

- **numpy & pandas** – For data handling.
- **streamlit** – To create a web-based interface.
- **cv2 (OpenCV)** – For image processing and face detection.
- **sqlite3** – To store user data in a database.
- **os** – To handle file operations.

---

2. Capturing an Image from the Webcam

Function: capture_image(save_path)

- Opens the webcam using OpenCV (cv2.VideoCapture(0)).
- Displays a live feed with a message: **"Press 'Space' to Capture"**.
- If the user presses **Space (" ")**, it saves the captured image as **"captured_image.jpg"** in the specified folder.
- Pressing **'q'** exits the webcam window.

Use Case: This is used to take a picture of a person (e.g., a customer in a hotel system).

---

3. Creating and Managing the SQLite Database

- **create_connection(db_file)** – Establishes a connection to the SQLite database.
- **create_table(conn)** – Creates a table user_data with fields (id, name, age, email) if it doesn't exist.
- **insert_data(conn, user_data)** – Inserts customer details into the database.

Use Case: This is used to store customer details in a **local database** for future reference.

---

4. Streamlit Web App for User Interaction

The **main()** function builds a simple web app with the following features:

- **Title:** "Webcam Image Capture and Database Storage"
- **Button to Capture an Image:** Saves the captured image.

- **Input Fields for Customer Details:** Name, Room Number, Age, and Email.
- **"Store in Database" Button:** Saves the entered details in the SQLite database.
- **"Classify" Button:** Detects faces, eyes, and smiles using OpenCV's **Haar Cascade classifiers**.
- **"Generate Bill" Button:** Placeholder for a billing system.

---

5. Face, Eye, and Smile Detection

If the **"Classify" button** is clicked:

- Loads **pre-trained Haar cascade models** for face, eye, and smile detection.
- Opens the webcam and detects faces, eyes, and smiles in real time.
- Draws bounding boxes around detected features.
- Saves detected features in **"detected_features.csv" and "detected_features.xlsx"**.
- Stops when the **'q' key** is pressed.

Use Case: Used for **face-based customer verification** in a hotel or other service-based systems.

---

6. Running the Application

The last part:

```
if __name__ == "__main__":
    main()
```

- Ensures the **main() function runs** when the script is executed.
- Launches the **Streamlit web interface**.

Summary of the Code's Functionality

✓ Takes pictures using a webcam and saves them
✓ Stores customer details in a database
✓ Detects faces, eyes, and smiles in real time
✓ Provides a web-based interface (Streamlit) for user interaction
✓ Allows bill generation (Placeholder for now)