

מיני פרוייקט בבסיסי נתונים

נושא: יקב

תת נושא: שירות לקוחות ביקב

מגישות: שיר באבאייב

214242216

ורננה חורי

326320868

תשפ"ה 2025

מחלקת שירות לקוחות ביקב

תיאור המחלקה:

המערכת מיועדת לנהל את כל המידע הקשור לשירות הלקוחות של היקב, תוך כדי שמירה על יעילות ואיכות השירות. היא מקיפה את כל המידע הנדרש לעיבוד הזמנות, תלונות, משלוחים, ומעקב אחר הלקוחות והעובדים.

הישויות המרכזיות במערכת:

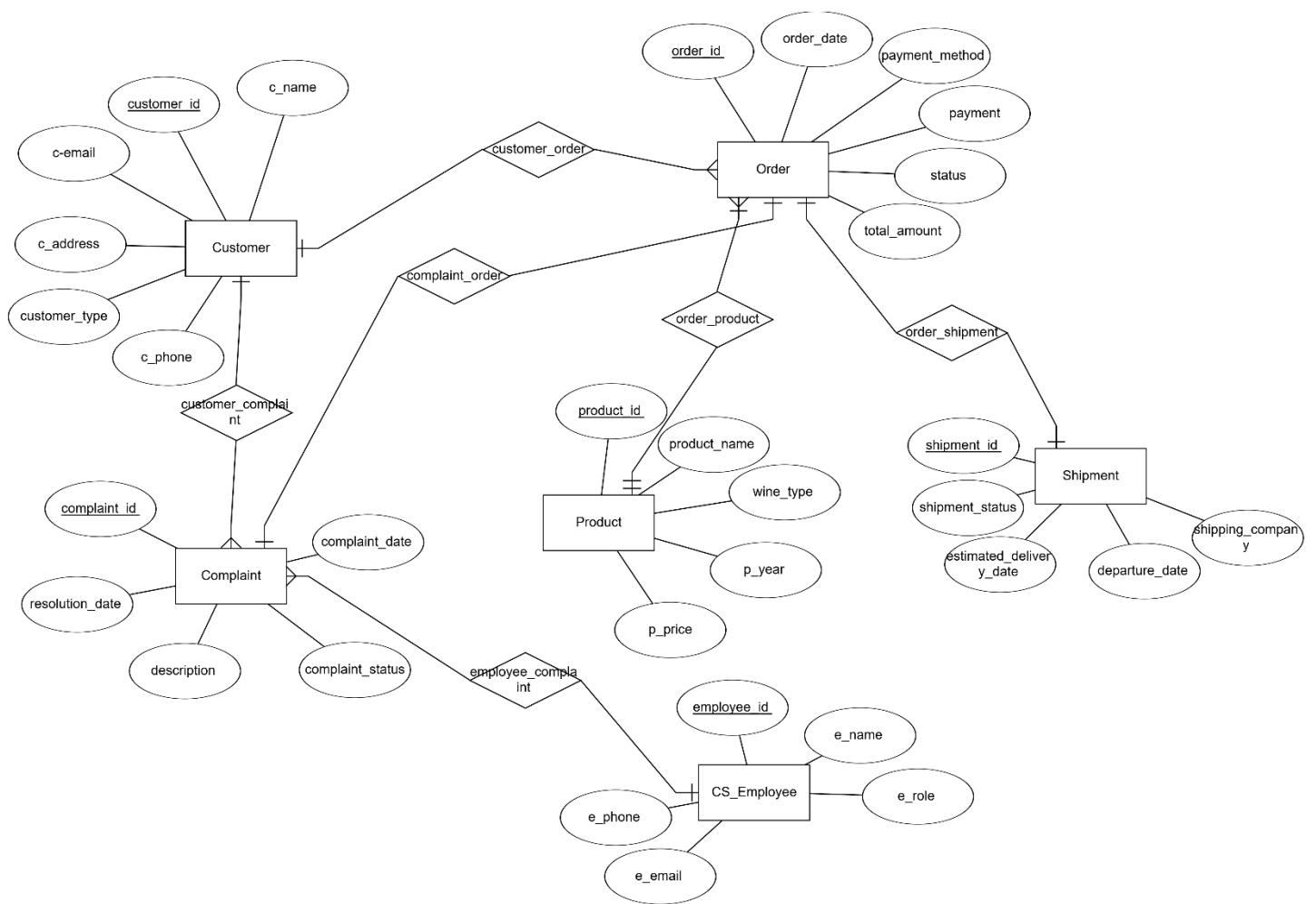
1. לקוח: מערכת זו שומרת את פרטי הלקוח, כולל שם, כתובת, פרטי קשר, וסוג הלקוח (כמו לקוח פרטי או עסקי). המידע הזה חיוני למעקב אחר כל לקוח והזמנותיו.
2. מוצר: כולל את המידע על המוצרים שנמכרים, כלומר יינות שונים המיוצרים על ידי היקב, פרטי המוצר כמו שם, סוג יין, שנת ייצור, ומחיר.
3. עובד: כל עובד המועסק בשירות הלקוחות של היקב, כולל שם, תפקיד, דוא"ל וטלפון. המידע הזה חשוב למעקב אחר העובדים המטפלים בתלונות ובשירות לקוחות.
4. תלונה: כל תלונה שמגיש לקוח בנוגע לשירות או למוצר, כולל תאריך הגשת התלונה, תיאור הבעיה, סטטוס התלונה ותאריך פתרונה, וכן המידע על הלקוח והעובד המטפל בתלונה. כל תלונה מתועדת ומנוהלת כדי לשמור על רמת שירות גבוהה.
5. הזמנה: כל הזמנה שמבצע לקוח, כולל תאריך ההזמנה, סטטוס ההזמנה, סך הסכום הכולל של ההזמנה, אמצעי התשלום, ומידע על המוצר המוזמן, הלקוח, המשלוח ואפשרות לקישור לתלונה (אם קיימת). המידע הזה מאפשר לעקוב אחרי כל הזמנה ולהבטיח שהלקוחות יקבלו את המוצרים באופן תקין ובזמן.
6. משלוח: מידע על המשלוחים, כולל שם חברת השילוח, תאריך היציאה והגעת המשלוח הצפוי, והסטטוס הנוכחי של המשלוח. זהו כלי חשוב לניהול ההפצה של המוצרים.

הפונקציות העיקריות של המערכת:

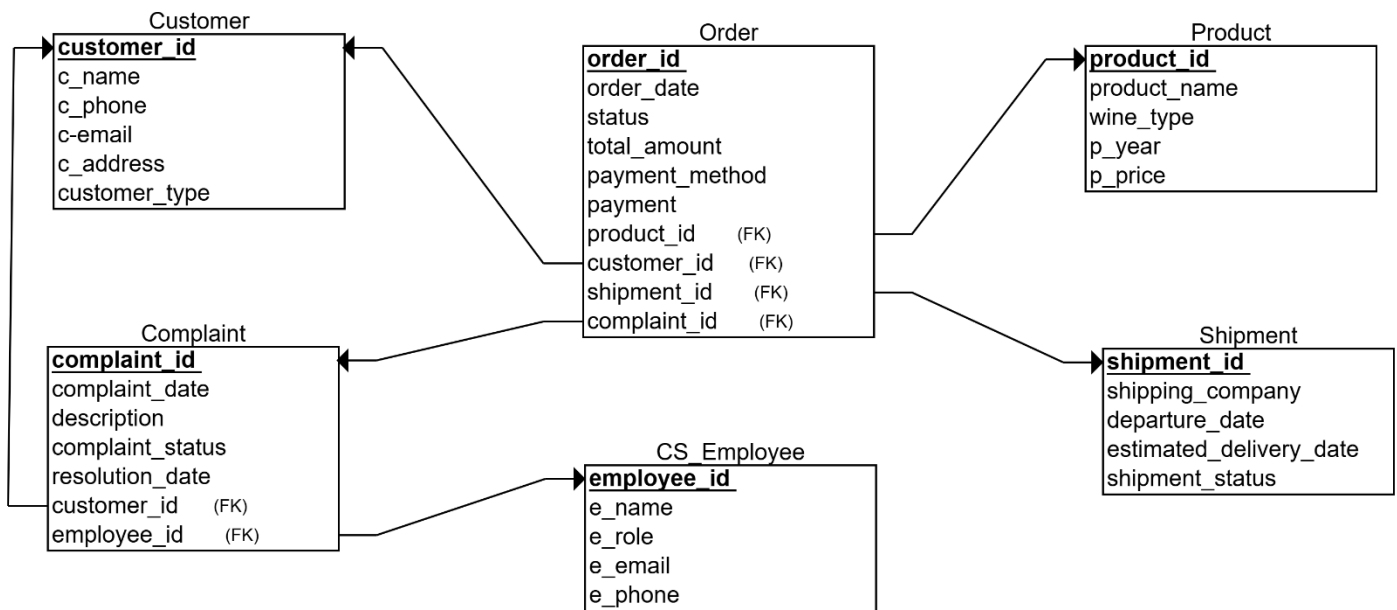
- ניהול הזמנות: המערכת מאפשרת לקבל ולעבד הזמנות מלקוחות, לעקוב אחרי הסטטוס של כל הזמנה, ולבצע תשלום ושילוח.
- טיפול בתלונות: המערכת מאפשרת לנהל את התלונות שמוגשות על ידי הלקוחות, לעקוב אחרי סטטוס התלונה, ולוודא כי כל תלונה מקבלת מענה הולם ופתרון בזמן.
- ניהול משלוחים: המערכת מאפשרת לעקוב אחרי מצב המשלוח ולוודא שהלקוחות מקבלים את המוצרים במועד שנקבע.

המערכת מבוססת על רישום ושימור מידע אודות כל הפעולות שביצע הלקוח, תלונותיו, הזמנותיו, תשלומים, ושירותי המשלוח, כדי לשפר את איכות השירות ולהגביר את שביעות רצון הלקוחות.

תרשים ERD:



תרשים DSD:



DSD מה שנוצר מהקובץ **createTables.sql**

```
CREATE TABLE Customer
(
  customer_id INT NOT NULL,
  c_name INT NOT NULL,
  c_phone_ INT NOT NULL,
  c_email INT NOT NULL,
  c_address_ INT NOT NULL,
  customer_type INT NOT NULL,
  PRIMARY KEY (customer_id)
);
```

```
CREATE TABLE Product
(
  product_id_ INT NOT NULL,
  product_name_ INT NOT NULL,
  wine_type_ INT NOT NULL,
  p_year_ INT NOT NULL,
  p_price INT NOT NULL,
  PRIMARY KEY (product_id_)
);
```

```
CREATE TABLE Complaint
(
  complaint_id_ INT NOT NULL,
  complaint_date_ INT NOT NULL,
  description_ INT NOT NULL,
  complaint_status INT NOT NULL,
  resolution_date_ INT NOT NULL,
  customer_id INT NOT NULL,
  employee_id_ INT NOT NULL,
  PRIMARY KEY (complaint_id_),
  FOREIGN KEY (customer_id) REFERENCES Customer(customer_id),
  FOREIGN KEY (employee_id_) REFERENCES CS_Employee(employee_id_)
);
```

```
CREATE TABLE Order
(
  order_id_ INT NOT NULL,
  order_date_ INT NOT NULL,
  status_ INT NOT NULL,
  total_amount_ INT NOT NULL,
  payment_method_ INT NOT NULL,
  payment INT NOT NULL,
  product_id_ INT NOT NULL,
  customer_id INT NOT NULL,
  shipment_id_ INT NOT NULL,
  complaint_id_ INT NOT NULL,
  PRIMARY KEY (order_id_),
  FOREIGN KEY (product_id_) REFERENCES Product(product_id_),
  FOREIGN KEY (customer_id) REFERENCES Customer(customer_id),
  FOREIGN KEY (shipment_id_) REFERENCES Shipment(shipment_id_),
  FOREIGN KEY (complaint_id_) REFERENCES Complaint(complaint_id_)
);
```

```
CREATE TABLE CS_Employee
(
  employee_id_ INT NOT NULL,
  e_name_ INT NOT NULL,
  e_role_ INT NOT NULL,
  e_email_ INT NOT NULL,
  e_phone_ INT NOT NULL,
  PRIMARY KEY (employee_id_)
);

CREATE TABLE Shipment
(
  shipment_id_ INT NOT NULL,
  shipping_company_ INT NOT NULL,
  departure_date_ INT NOT NULL,
  estimated_delivery_date_ INT NOT NULL,
  shipment_status INT NOT NULL,
  PRIMARY KEY (shipment_id_)
);
```

לאחר השינויים:

```

CREATE TABLE Customer
(
  customer_id INT NOT NULL,
  c_name VARCHAR2(100) NOT NULL,
  c_phone_ VARCHAR2(20) NOT NULL,
  c_email VARCHAR2(100) NOT NULL,
  c_address_ VARCHAR2(255) NOT NULL,
  customer_type INT NOT NULL,
  PRIMARY KEY (customer_id)
);

CREATE TABLE Product
(
  product_id_ INT NOT NULL,
  product_name_ VARCHAR2(100) NOT NULL,
  wine_type_ VARCHAR2(50) NOT NULL,
  p_year_ INT NOT NULL,
  p_price NUMERIC(10,2) NOT NULL,
  PRIMARY KEY (product_id_)
);

CREATE TABLE CS_Employee
(
  employee_id_ INT NOT NULL,
  e_name_ VARCHAR2(100) NOT NULL,
  e_role_ VARCHAR2(50) NOT NULL,
  e_email_ VARCHAR2(100) NOT NULL,
  e_phone_ VARCHAR2(20) NOT NULL,
  PRIMARY KEY (employee_id_)
);

```

```

CREATE TABLE Shipment
(
  shipment_id_ INT NOT NULL,
  shipping_company_ VARCHAR2(100) NOT NULL,
  departure_date_ DATE NOT NULL,
  estimated_delivery_date_ DATE NOT NULL,
  shipment_status VARCHAR2(50) NOT NULL,
  PRIMARY KEY (shipment_id_)
);

CREATE TABLE Complaint
(
  complaint_id_ INT NOT NULL,
  complaint_date_ DATE NOT NULL,
  description_ VARCHAR2(500) NOT NULL,
  complaint_status VARCHAR2(50) NOT NULL,
  resolution_date_ DATE NOT NULL,
  customer_id INT NOT NULL,
  employee_id_ INT NOT NULL,
  PRIMARY KEY (complaint_id_),
  FOREIGN KEY (customer_id) REFERENCES Customer(customer_id),
  FOREIGN KEY (employee_id_) REFERENCES CS_Employee(employee_id_)
);

```

```

CREATE TABLE Order
(
  order_id_ INT NOT NULL,
  order_date_ DATE NOT NULL,
  status_ VARCHAR2(50) NOT NULL,
  total_amount_ NUMERIC(10,2) NOT NULL,
  payment_method_ VARCHAR2(50) NOT NULL,
  payment NUMERIC(10,2) NOT NULL,
  product_id_ INT NOT NULL,
  customer_id INT NOT NULL,
  shipment_id_ INT NOT NULL,
  complaint_id_ INT, -- **לא חובה**
  PRIMARY KEY (order_id_),
  FOREIGN KEY (product_id_) REFERENCES Product(product_id_),
  FOREIGN KEY (customer_id) REFERENCES Customer(customer_id),
  FOREIGN KEY (shipment_id_) REFERENCES Shipment(shipment_id_),
  FOREIGN KEY (complaint_id_) REFERENCES Complaint(complaint_id_)
);

```

dropTables.sql

```
DROP TABLE Order CASCADE CONSTRAINTS;  
DROP TABLE Complaint CASCADE CONSTRAINTS;  
DROP TABLE Shipment CASCADE CONSTRAINTS;  
DROP TABLE CS_Employee CASCADE CONSTRAINTS;  
DROP TABLE Product CASCADE CONSTRAINTS;  
DROP TABLE Customer CASCADE CONSTRAINTS;  
|
```

:selectAll.sql

```
-- הצגת כל הנתונים מטבלת Customer  
SELECT * FROM Customer;  
  
-- הצגת כל הנתונים מטבלת Product  
SELECT * FROM Product;  
  
-- הצגת כל הנתונים מטבלת CS_Employee  
SELECT * FROM CS_Employee;  
  
-- הצגת כל הנתונים מטבלת Shipment  
SELECT * FROM Shipment;  
  
-- הצגת כל הנתונים מטבלת Complaint  
SELECT * FROM Complaint;  
  
-- הצגת כל הנתונים מטבלת Order  
SELECT * FROM "Order";
```

הכנסת נתונים מקובץ EXCEL:

G	F	E	D	C	B	A
110	175	26/07/2025	Open	Community	20/12/2024	2
67	94	09/04/2025	Open	Small adul	10/03/2025	3
77	163	29/04/2025	Closed	Above old	10/08/2024	4
81	19	06/03/2026	In Progress	Research	22/02/2025	5
111	59	24/09/2025	Open	Less accor	25/04/2024	6
176	14	14/05/2025	Open	Risk decisi	20/01/2025	7
188	127	14/07/2025	Closed	Entire life	05/12/2024	8
190	110	30/04/2025	Resolved	Day full to	02/05/2024	9
155	58	13/09/2025	Resolved	The audier	07/11/2024	10
199	179	21/04/2025	Closed	Worry cut	11/10/2024	11
146	101	17/10/2025	Resolved	Ground m	09/02/2025	12
17	67	04/09/2025	Open	Know near	20/09/2024	13
101	185	02/05/2025	Closed	Idea born	07/09/2024	14
144	136	19/05/2025	Closed	Provide re	14/11/2024	15
150	22	14/09/2025	In Progress	Owner like	24/04/2024	16

יצירת רשומות באמצעות סקריפט בפייטון: (בחץ אדום מסומנים הקבצים שנוצרו בתיקייה בעקבות הרצת התוכנית).

```

1 import csv
2 import random
3 from datetime import datetime, timedelta
4
5 def write_csv(filename, data, headers):
6     with open(filename, mode='w', newline='', encoding='utf-8') as file:
7         writer = csv.writer(file)
8         writer.writerow(headers)
9         writer.writerows(data)
10
11 def generate_customers(n=200):
12     data = [(i + 300, f'Customer {i + 300}', f'05{random.randint(10000000, 99999999)}',
13             f'customer{i + 300}@example.com', f'Address {i + 300}', random.randint(1, 3)) for i in range(n)]
14     write_csv("customers.csv", data, ["customer_id", "c_name", "c_phone_", "c_email", "c_address_", "customer_type"])
15
16 def generate_products(n=200):
17     wine_types = ['Red', 'White', 'Rose', 'Sparkling']
18     data = [(i + 300, f'Wine {i + 300}', random.choice(wine_types), random.randint(1990, 2023), round(random.uniform(50, 500), 2))
19             for i in range(n)]
20     write_csv("products.csv", data, ["product_id_", "product_name_", "wine_type_", "p_year_", "p_price"])
21
22 def generate_employees(n=200):
23     roles = ['Support', 'Manager', 'Sales', 'Tech']

```



```

def generate_employees(n=200):
    roles = ['Support', 'Manager', 'Sales', 'Tech']
    data = [(i + 300, f'Employee {i + 300}', random.choice(roles), f'employee{i + 300}@company.com',
               f'05{random.randint(10000000, 99999999)}') for i in range(n)]
    write_csv("employees.csv", data, ["employee_id_", "e_name_", "e_role_", "e_email_", "e_phone_"])

def generate_shipments(n=200):
    status_list = ['In Transit', 'Delivered', 'Pending']
    data = [(i + 300, f'Company {i + 300}', (datetime.now() - timedelta(days=random.randint(1, 30))).date(),
               (datetime.now() - timedelta(days=random.randint(1, 30)) + timedelta(days=random.randint(2, 10))).date(),
               random.choice(status_list)) for i in range(n)]
    write_csv("shipments.csv", data, ["shipment_id_", "shipping_company_", "departure_date_", "estimated_delivery_date_", "shipment_status_"])

def generate_complaints(n=200):
    status_list = ['Open', 'Closed', 'In Progress']
    data = [(i + 300, (datetime.now() - timedelta(days=random.randint(1, 100))).date(), f'Complaint {i + 300}', random.choice(status_list),
               (datetime.now() - timedelta(days=random.randint(1, 100)) + timedelta(days=random.randint(1, 30))).date(),
               random.randint(300, 500), random.randint(300, 500)) for i in range(n)]
    write_csv("complaints.csv", data, ["complaint_id_", "complaint_date_", "description_", "complaint_status_", "resolution_date_", "customer_id_"])

```

```

def generate_orders(n=200):
    status_list = ['Processing', 'Shipped', 'Completed', 'Cancelled']
    payment_methods = ['Credit Card', 'PayPal', 'Bank Transfer', 'Cash']
    data = [(i + 300, (datetime.now() - timedelta(days=random.randint(1, 60))).date(), random.choice(status_list),
               round(random.uniform(100, 2000), 2), random.choice(payment_methods),
               round(random.uniform(100, 2000), 2) if random.random() > 0.2 else round(random.uniform(50, 2000), 2),
               random.randint(300, 500), random.randint(300, 500), random.randint(300, 500), random.choice([None, random.randint(300, 500)]))
               for i in range(n)]
    write_csv("orders.csv", data, ["order_id_", "order_date_", "status_", "total_amount_", "payment_method_", "payment", "product_id_", "customer_id_"])

# יצירת קבצי CSV
generate_customers()
generate_products()
generate_employees()
generate_shipments()
generate_complaints()
generate_orders()

print("Data generation completed. CSV files created successfully!")

```

יצירת קבצי CSV עם רשומות באמצעות האתר mockaroo:

Field Name

Type

Options

customer_id

Row Number

blank: 0 %

Σ

×

c_name

First Name

blank: 0 %

Σ

×

c_email

Email Address

blank: 0 %

Σ

×

c_address

Street Address

blank: 0 %

Σ

×

customer_type

Number

min: 1 max: 5 decimals: 0 blank: 0 %

Σ

×

c_phone

Phone

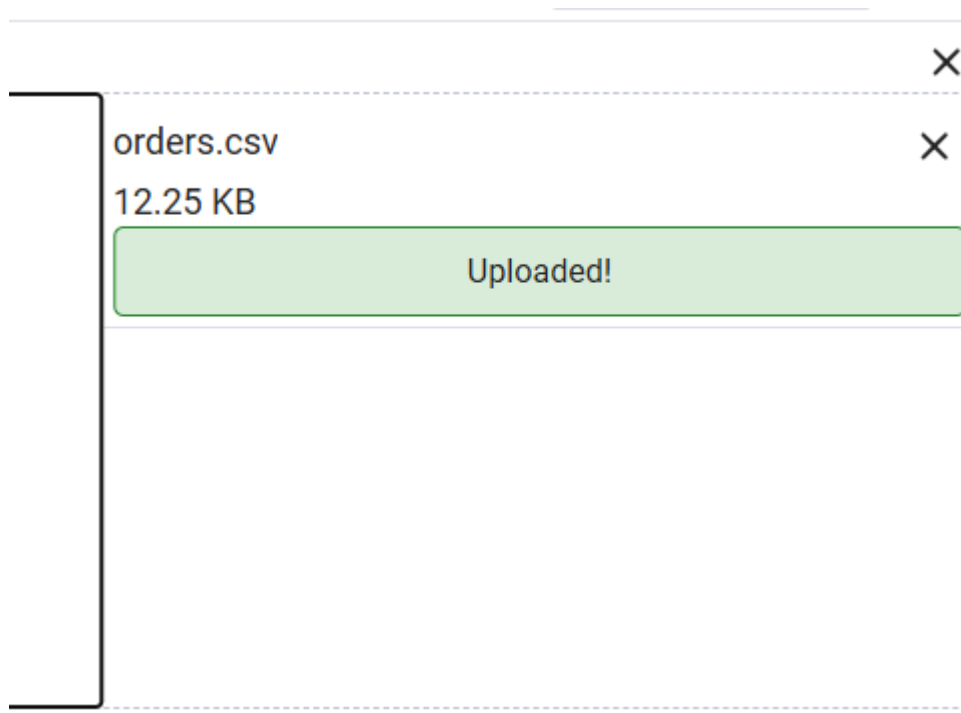
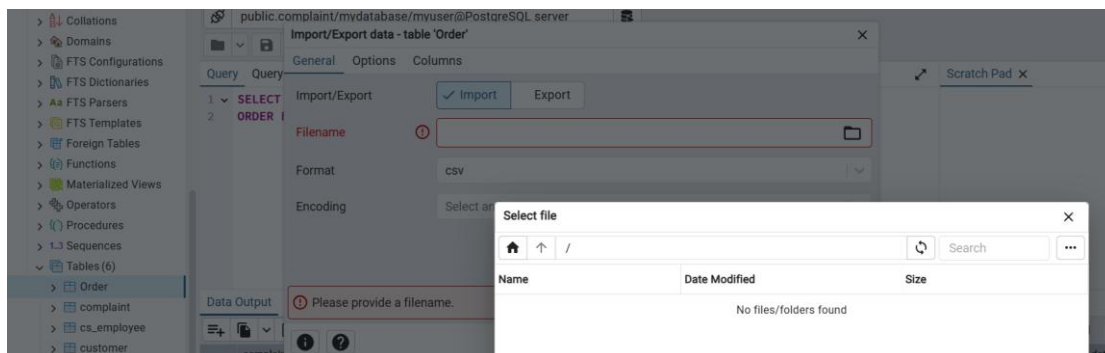
format: ###-###-#### blank: 0 %

Σ

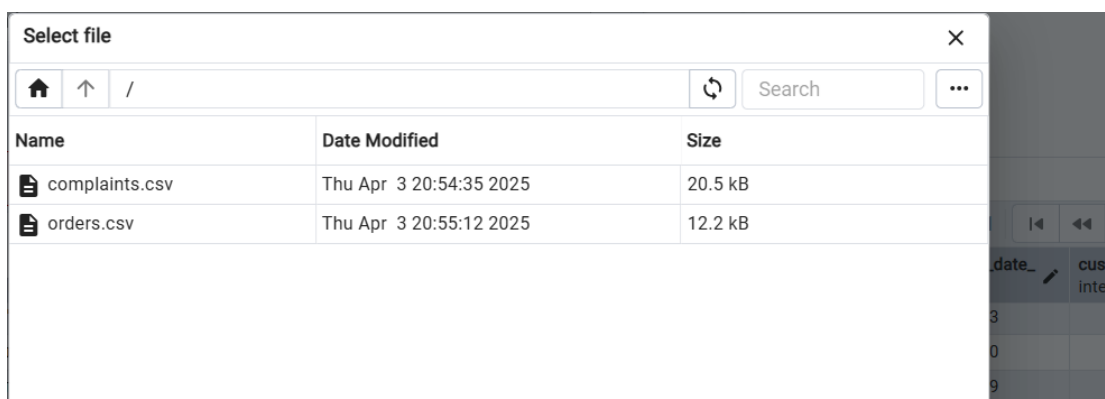
×

TABLE					RAW				
order_id	order_date	status	total_amount	payment_method	payment	product_id	customer_id	shipment_id	complaint_id
1023	01/01/2025	Processing	3	העברה בנקאית	189.99	1023	37485923	SH12349	13748
5236	16/08/2024	Cancelled	3	אשראי	149.99	2847	58237491	SH12347	61582
4123	15/03/2025	Failed	4	אשראי	149.99	3951	91827364	SH12346	20347
5236	25/01/2025	Failed	5	העברה בנקאית	189.99	4738	10293847	SH12347	13748
4123	09/11/2024	Pending	3	אשראי	99.99	5820	56473829	SH12347	72904
2045	28/04/2024	Processing	6	אשראי	399.99	6194	29384756	SH12348	13748
5236	11/04/2024	Shipped	3	כיס	299.99	7265	84736291	SH12349	20347

העלאת קבצים לpgAdmin:



הכנסת הרשומות לטבלאות שיצרנו באתר:



Import/Export data - table 'Order'

General
Options
Columns

Import/Export

✓ Import

Export

Filename
/orders.csv

Format
csv

Encoding
Select an item...

Close
Reset
OK

חלק מהרשומות שנוספו לטבלת 'תלונה':

complaint_id_	complaint_date_	description_	complaint_status	resolution_date_	customer_id	
[PK] integer	date	character varying (500)	character varying (50)	date	integer	
413	78105	2024-12-09	הבקבוק נשבר במהלך המשלוח והיין נפגע	פתוחה	2024-09-13	
414	82035	2024-04-03	המחיר גבוה לעומת איכות ירודה	פתוחה	2024-08-10	
415	85673	2024-12-21	האיכות הייתה גרועה והמשלוח היה באיחור	פתוחה	2025-01-09	
416	90283	2024-05-12	האיכות הייתה ירודה והמכירה מאכזבת	פתוחה	2024-10-14	
417	90376	2024-10-21	קיבלתי בקבוק שבור והיין נפגם	פתוחה	2025-02-17	
418	94721	2024-10-24	משלוח מאוחר והיין לא תאם את הציפיות	בטיפול	2025-03-14	

גיבוי:

Backup (Table: complaint)

General
Data Options
Query Options
Table Options
Options
Objects

FilenameDB_Backup_02-04-2025

FormatTar

Compression ratio

EncodingSelect an item...

Number of jobs

Role nameSelect an item...

שחזור הנתונים: השחזור עבד בהצלחה!

Restore (Database: postgres)

General
Data Options
Query Options
Table Options
Options

FormatCustom or tar

Filename/DB_Backup_02-04-2025

Number of jobs

Role nameSelect an item...

i
?

Close
Reset
Restore

Process completed

Restoring backup on the server 'PostgreSQL server (172.17.0.2:5432)'

[View Processes](#)**Process started**

Restoring backup on the server 'PostgreSQL server (172.17.0.2:5432)'

[View Processes](#)