CS434 – Data Base Theory and Design

Project #3

Team Database Application (TDA): Part 3 – Schema Creation and Testing

Team

Lipika Baniya | 800794205 | lbaniya@siue.edu

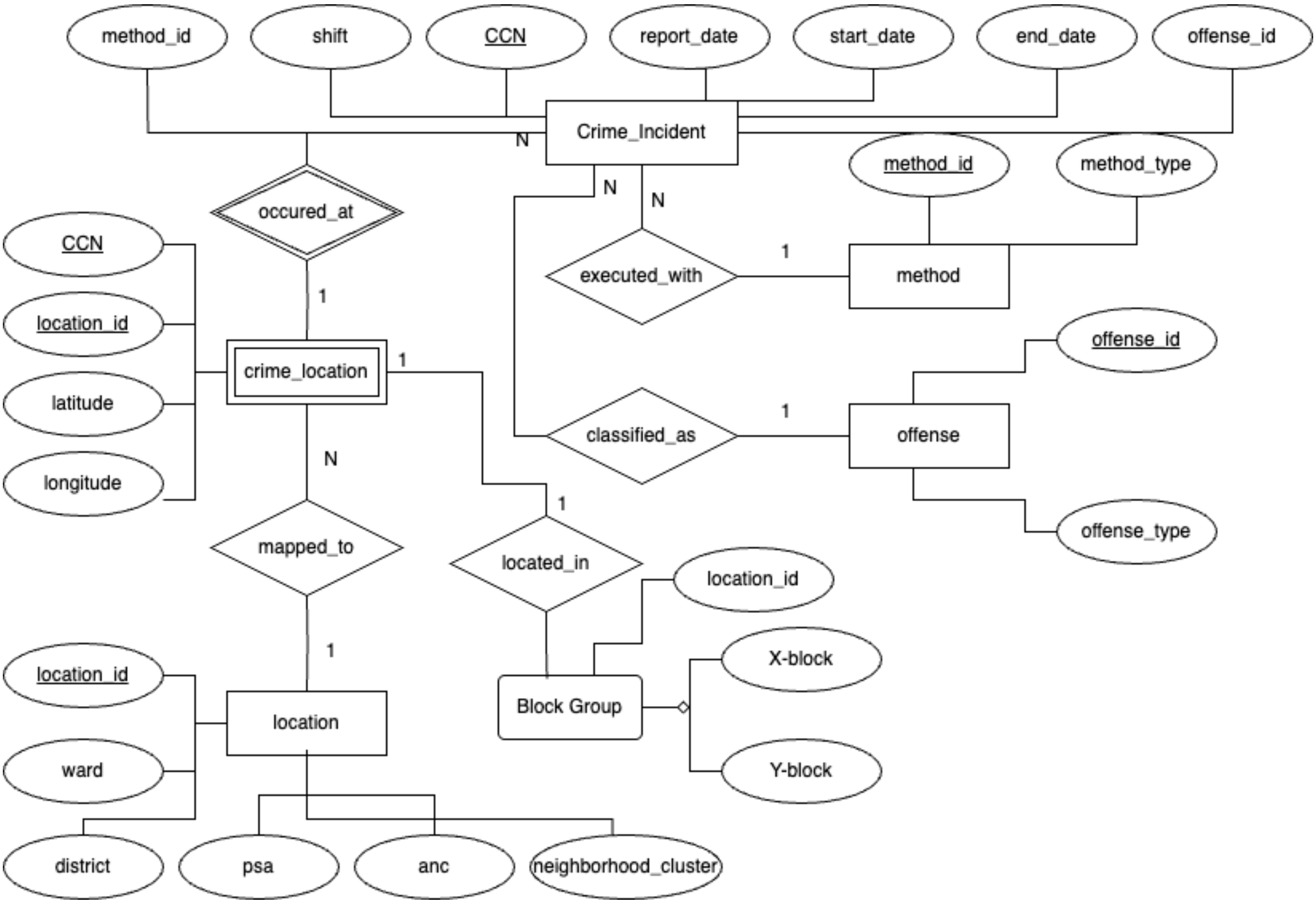
The domain I would like to manage with the TDA is **Washington DC Crime Datasets 2024** by the District of Columbia Metropolitan Police Department (MPD).

General Nature of application

The main goal of an Entity Relationship Diagram (ER Diagram) is to explain the relationship between entities; it is a structural design of the database. Through the help of specialized symbols, it helps to define the relationship between entities. It is based on three main principles entities, attributes and relationships, these help to design the database that would be required before implementing the database. It is a systematic process to design a database as it would require analyzing all requirements.

About Data

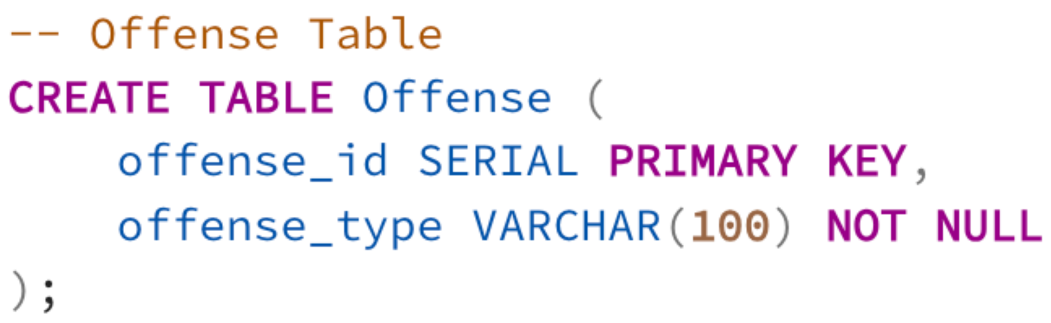
Washington, D.C. has been facing significant challenges in ensuring public safety due to the varying and growing crime rates in different neighborhoods and time periods. It is important for law enforcement agencies to understand when and where crimes occur so that it can respond efficiently and allocate limited resources wisely. Imagine a robust database system that is designed to handle this task effectively, because without a data-driven approach, policing efforts may remain reactive, which would result in delays or gaps in coverage in high-risk areas.



Tables:

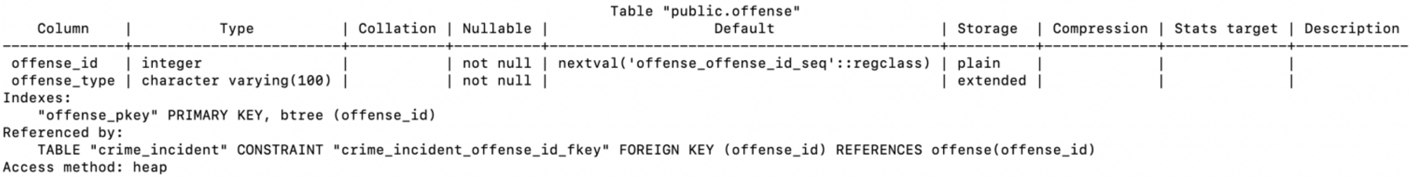
1. Offense

1.1.Command to Create Table



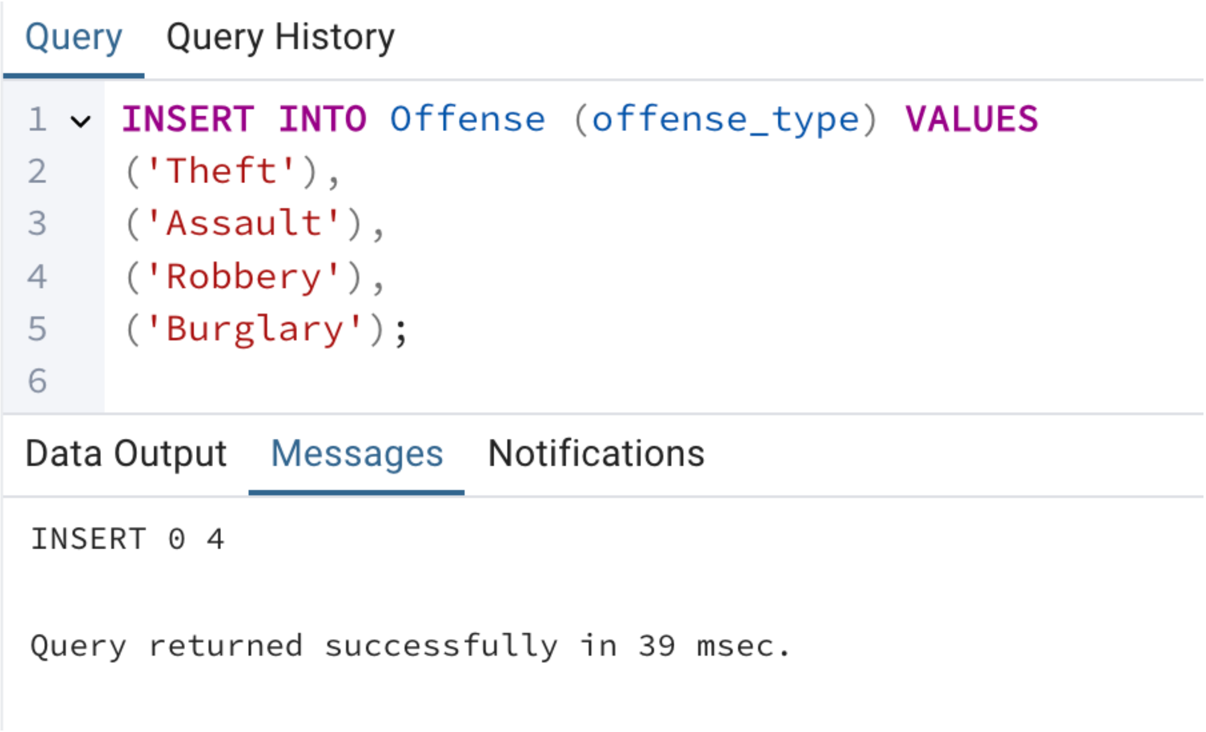
```
-- Offense Table
CREATE TABLE Offense (
    offense_id SERIAL PRIMARY KEY,
    offense_type VARCHAR(100) NOT NULL
);
```

1.2.Describing relation schemas



```
                                                     Table "public.offense"
    Column    |          Type          | Collation | Nullable |                   Default                   | Storage  | Compression | Stats target | Description
--------------+------------------------+-----------+----------+---------------------------------------------+----------+-------------+--------------+-------------
 offense_id   | integer                |           | not null | nextval('offense_offense_id_seq'::regclass) | plain    |             |              |
 offense_type | character varying(100) |           | not null |                                             | extended |             |              |
Indexes:
    "offense_pkey" PRIMARY KEY, btree (offense_id)
Referenced by:
    TABLE "crime_incident" CONSTRAINT "crime_incident_offense_id_fkey" FOREIGN KEY (offense_id) REFERENCES offense(offense_id)
Access method: heap
```

1.3.Inserting few records



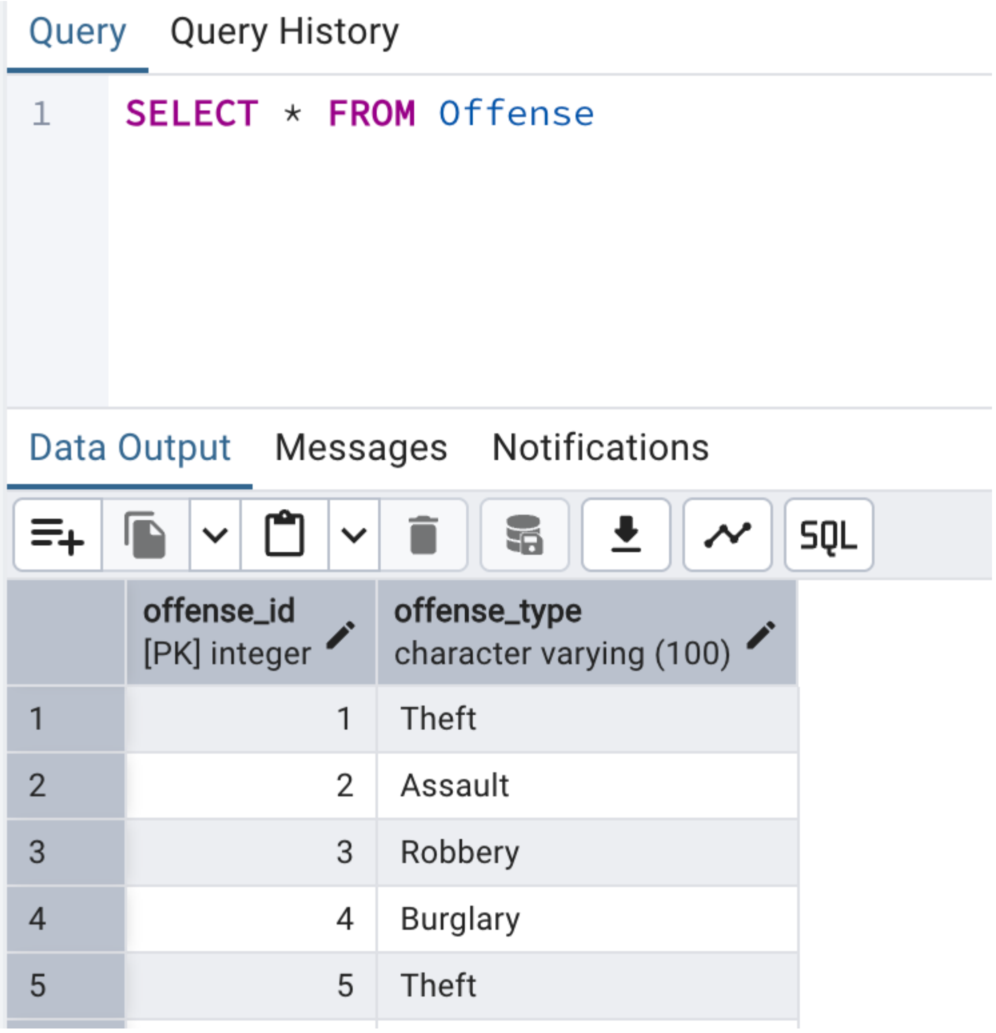
```
INSERT INTO Offense (offense_type) VALUES
('Theft'),
('Assault'),
('Robbery'),
('Burglary');

```

```
INSERT 0 4

Query returned successfully in 39 msec.
```

1.4.Listing the contents

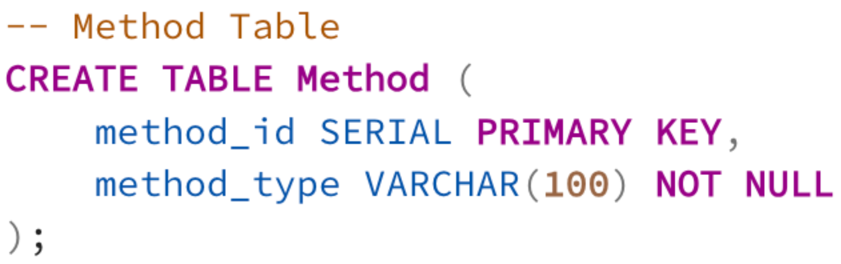


```
SELECT * FROM Offense
```

	offense_id [PK] integer	offense_type character varying (100)
1	1	Theft
2	2	Assault
3	3	Robbery
4	4	Burglary
5	5	Theft

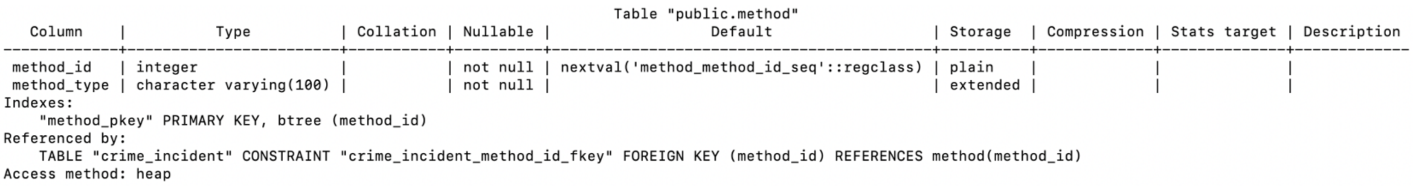
2. Method

2.1. Command to Create Table



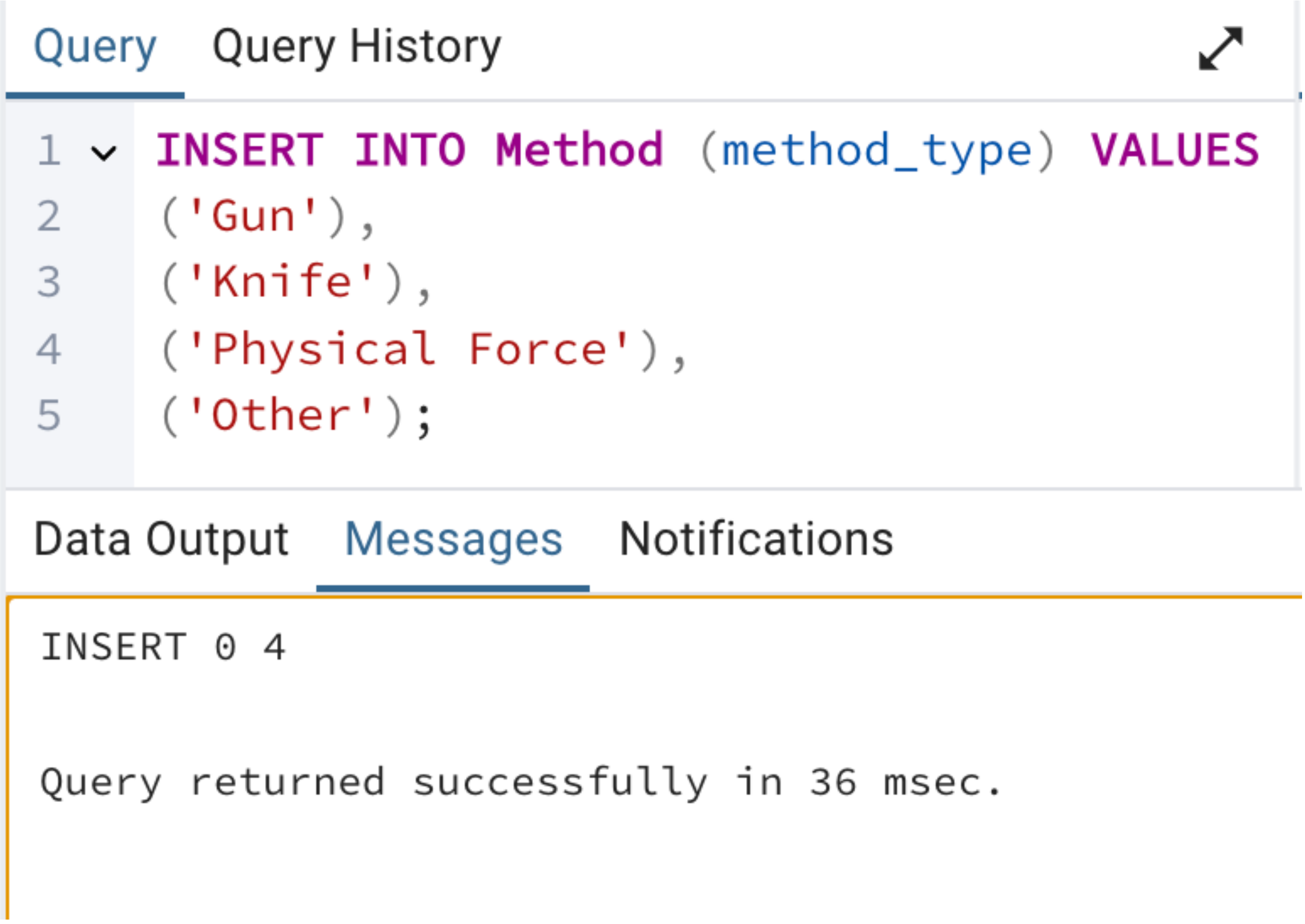
```sql
-- Method Table
CREATE TABLE Method (
    method_id SERIAL PRIMARY KEY,
    method_type VARCHAR(100) NOT NULL
);
```

2.2.Describing relation schemas



```
                                                           Table "public.method"
   Column    |          Type          | Collation | Nullable |                    Default                    | Storage  | Compression | Stats target | Description
-------------+------------------------+-----------+----------+-----------------------------------------------+----------+-------------+--------------+-------------
 method_id   | integer                |           | not null | nextval('method_method_id_seq'::regclass) | plain    |             |              |
 method_type | character varying(100) |           | not null |                                               | extended |             |              |
Indexes:
    "method_pkey" PRIMARY KEY, btree (method_id)
Referenced by:
    TABLE "crime_incident" CONSTRAINT "crime_incident_method_id_fkey" FOREIGN KEY (method_id) REFERENCES method(method_id)
Access method: heap
```

2.3.Inserting few records

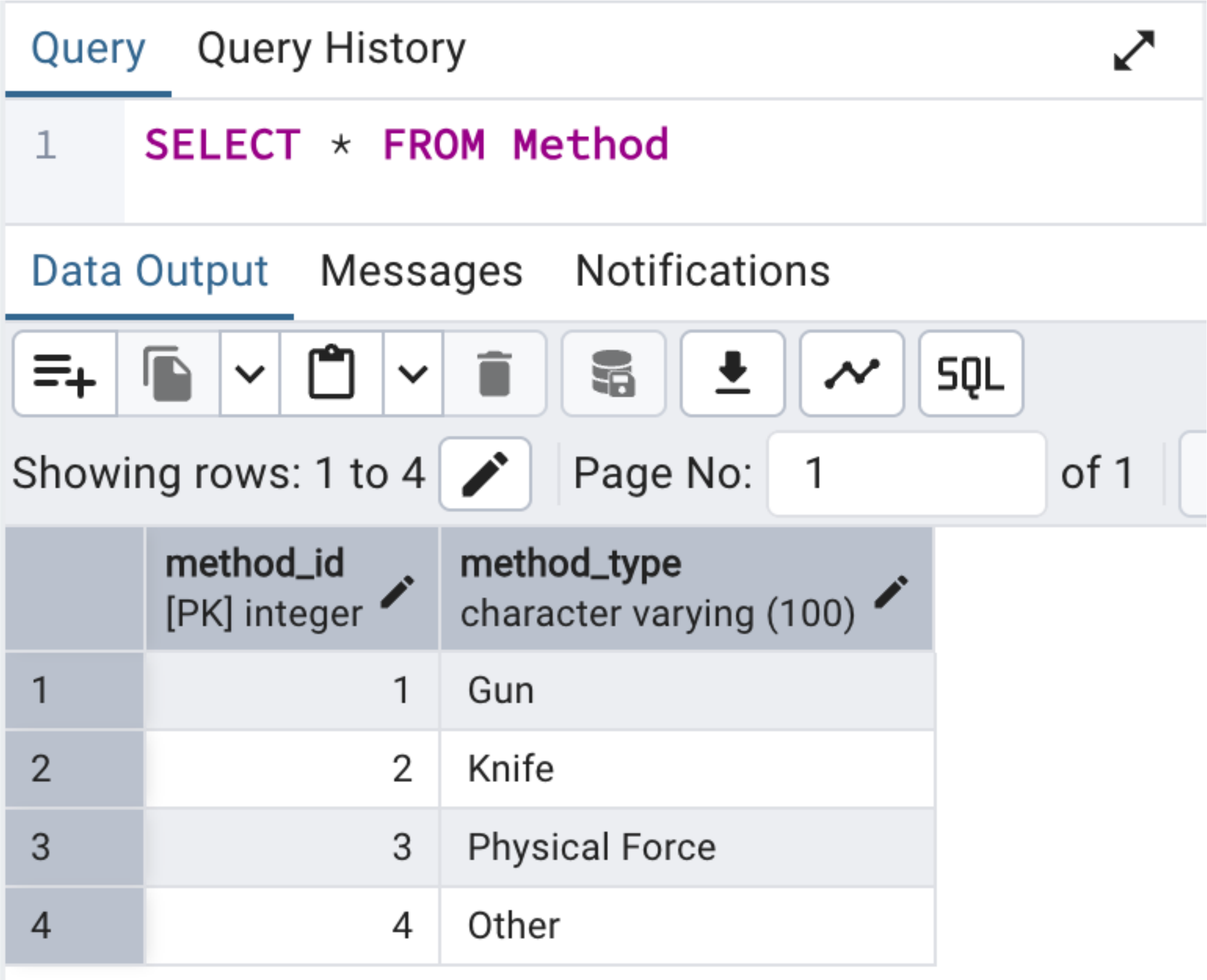


```sql
INSERT INTO Method (method_type) VALUES
('Gun'),
('Knife'),
('Physical Force'),
('Other');
```

```
INSERT 0 4

Query returned successfully in 36 msec.
```

2.4.Listing the contents

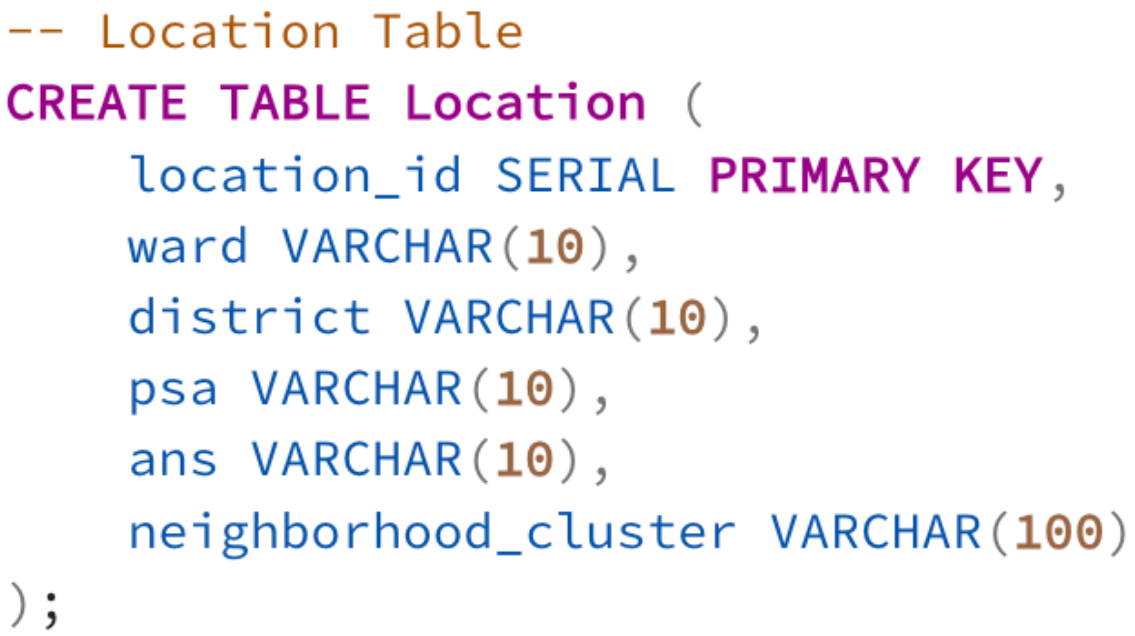


```
SELECT * FROM Method
```

	method_id [PK] integer	method_type character varying (100)
1	1	Gun
2	2	Knife
3	3	Physical Force
4	4	Other

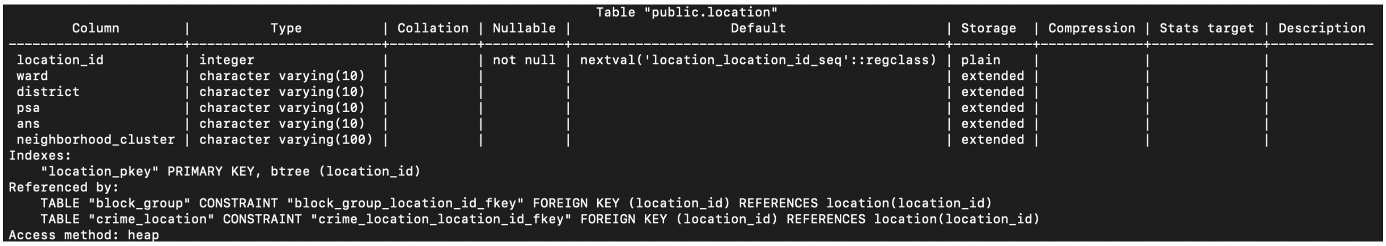
3. Location

3.1. Command to Create Table



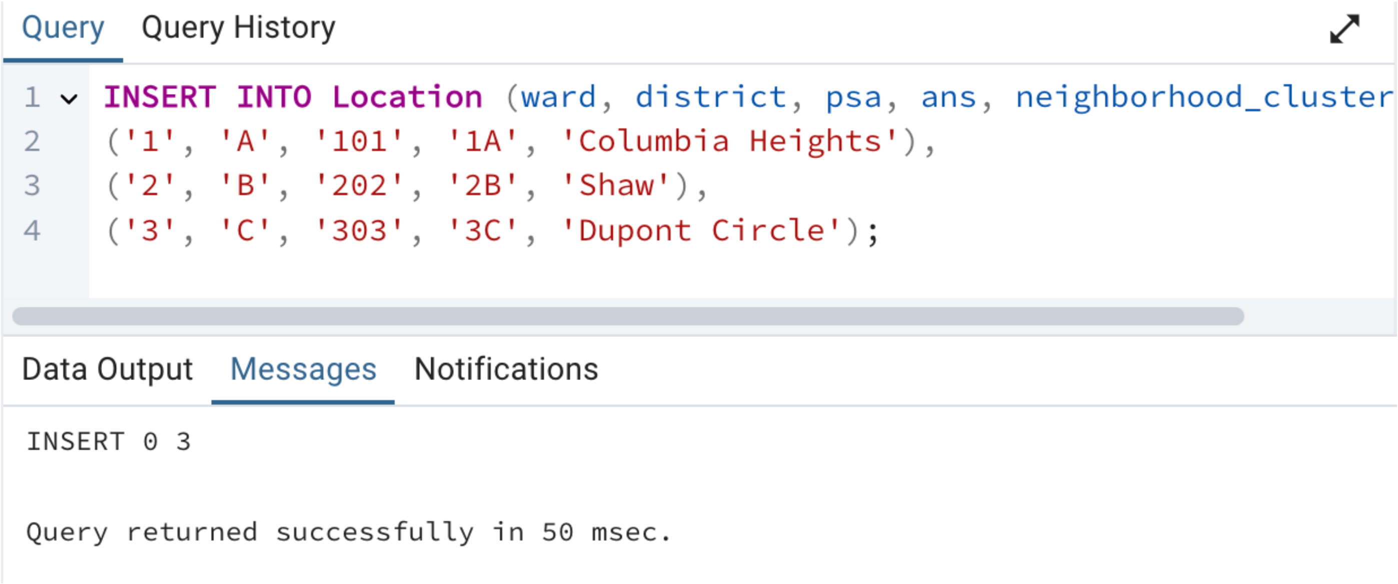
```
-- Location Table
CREATE TABLE Location (
    location_id SERIAL PRIMARY KEY,
    ward VARCHAR(10),
    district VARCHAR(10),
    psa VARCHAR(10),
    ans VARCHAR(10),
    neighborhood_cluster VARCHAR(100)
);
```

3.2. Describing relation schemas



```
                                                                  Table "public.location"
        Column        |          Type          | Collation | Nullable |                    Default                    | Storage  | Compression | Stats target | Description
----------------------+------------------------+-----------+----------+-----------------------------------------------+----------+-------------+--------------+-------------
 location_id          | integer                |           | not null | nextval('location_location_id_seq'::regclass) | plain    |             |              |
 ward                 | character varying(10)  |           |          |                                               | extended |             |              |
 district             | character varying(10)  |           |          |                                               | extended |             |              |
 psa                  | character varying(10)  |           |          |                                               | extended |             |              |
 ans                  | character varying(10)  |           |          |                                               | extended |             |              |
 neighborhood_cluster | character varying(100) |           |          |                                               | extended |             |              |
Indexes:
    "location_pkey" PRIMARY KEY, btree (location_id)
Referenced by:
    TABLE "block_group" CONSTRAINT "block_group_location_id_fkey" FOREIGN KEY (location_id) REFERENCES location(location_id)
    TABLE "crime_location" CONSTRAINT "crime_location_location_id_fkey" FOREIGN KEY (location_id) REFERENCES location(location_id)
Access method: heap
```

3.3. Inserting few records

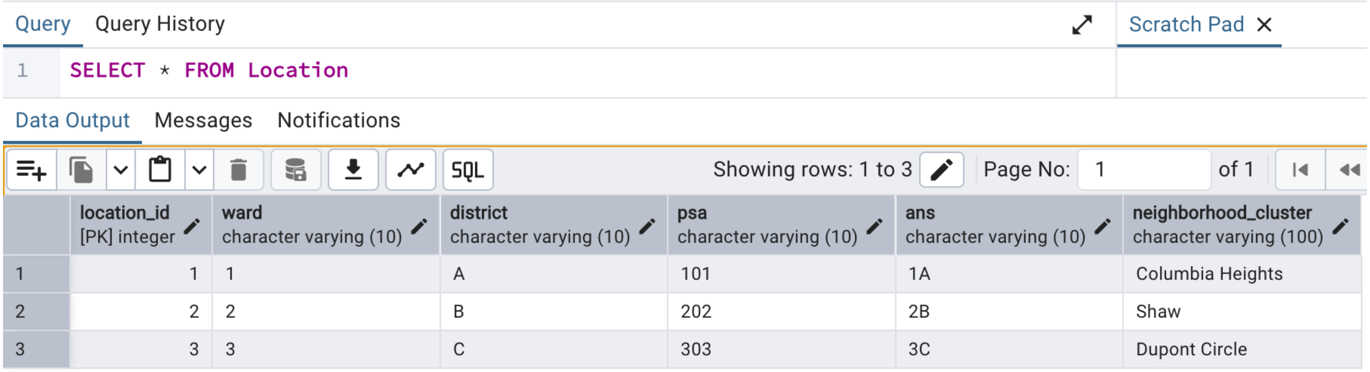


```
INSERT INTO Location (ward, district, psa, ans, neighborhood_cluster
('1', 'A', '101', '1A', 'Columbia Heights'),
('2', 'B', '202', '2B', 'Shaw'),
('3', 'C', '303', '3C', 'Dupont Circle');
```

```
INSERT 0 3

Query returned successfully in 50 msec.
```

3.4.Listing the contents

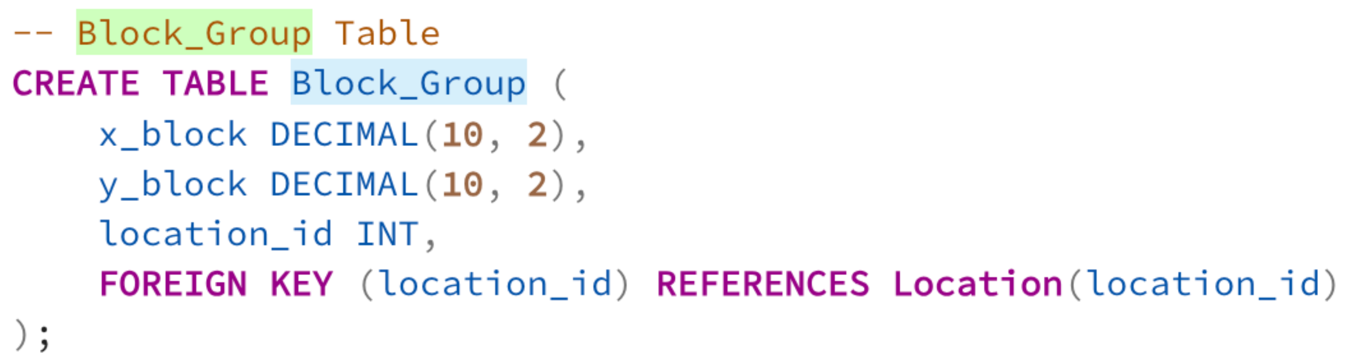


```
SELECT * FROM Location
```

	location_id [PK] integer	ward character varying (10)	district character varying (10)	psa character varying (10)	ans character varying (10)	neighborhood_cluster character varying (100)
1	1	1	A	101	1A	Columbia Heights
2	2	2	B	202	2B	Shaw
3	3	3	C	303	3C	Dupont Circle

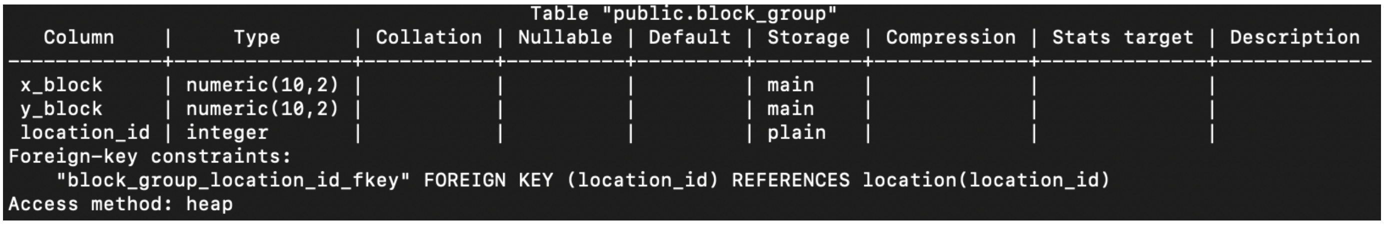
4. Block Group

4.1.Command to Create Table



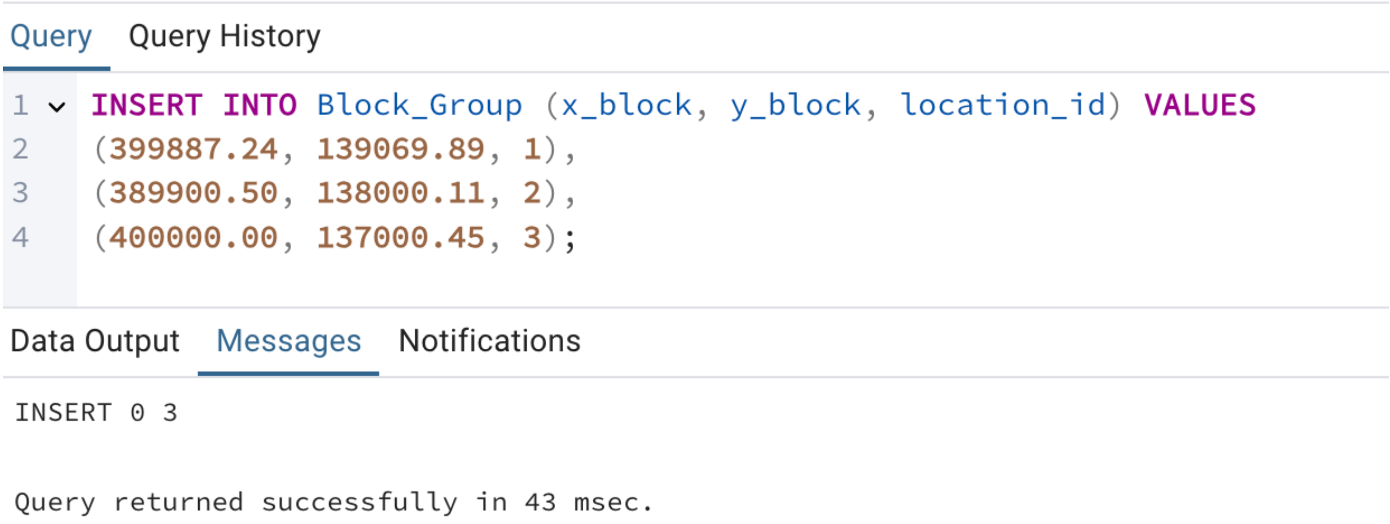
```sql
-- Block_Group Table
CREATE TABLE Block_Group (
    x_block DECIMAL(10, 2),
    y_block DECIMAL(10, 2),
    location_id INT,
    FOREIGN KEY (location_id) REFERENCES Location(location_id)
);
```

4.2.Describing relation schemas



```
                                          Table "public.block_group"
   Column    |     Type      | Collation | Nullable | Default | Storage | Compression | Stats target | Description
-------------+---------------+-----------+----------+---------+---------+-------------+--------------+-------------
 x_block     | numeric(10,2) |           |          |         | main    |             |              |
 y_block     | numeric(10,2) |           |          |         | main    |             |              |
 location_id | integer       |           |          |         | plain   |             |              |
Foreign-key constraints:
    "block_group_location_id_fkey" FOREIGN KEY (location_id) REFERENCES location(location_id)
Access method: heap
```

4.3.Inserting few records

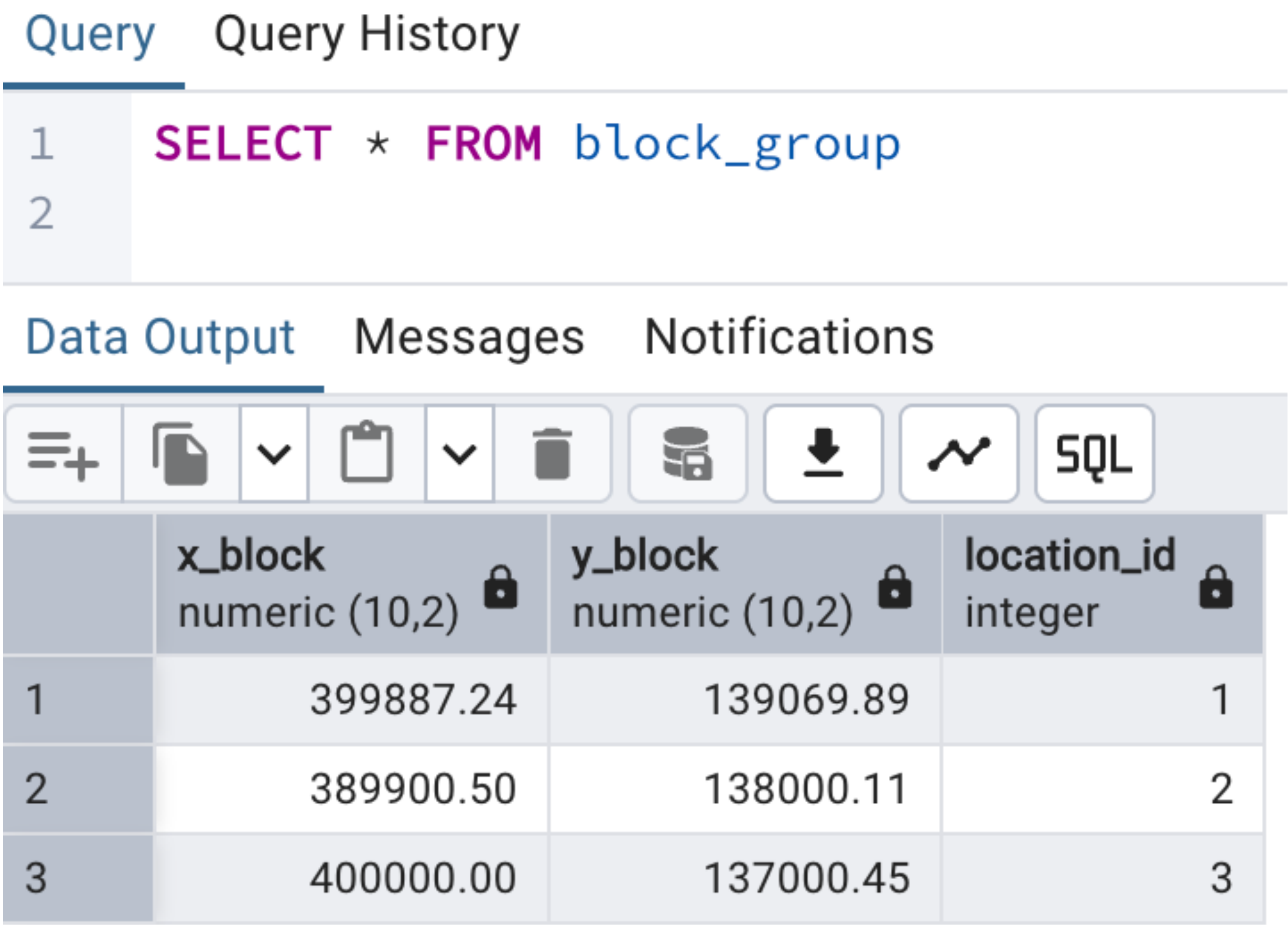


```sql
INSERT INTO Block_Group (x_block, y_block, location_id) VALUES
(399887.24, 139069.89, 1),
(389900.50, 138000.11, 2),
(400000.00, 137000.45, 3);
```

```
INSERT 0 3

Query returned successfully in 43 msec.
```

4.4.Listing the contents



```
SELECT * FROM block_group
```

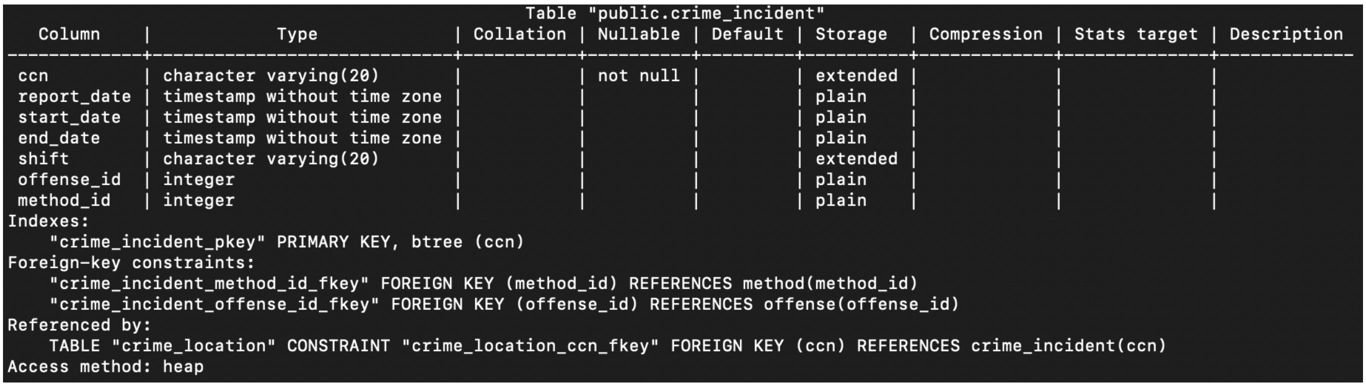
	x_block numeric (10,2)	y_block numeric (10,2)	location_id integer
1	399887.24	139069.89	1
2	389900.50	138000.11	2
3	400000.00	137000.45	3

5. Crime_Incident

5.1.Command to Create Table

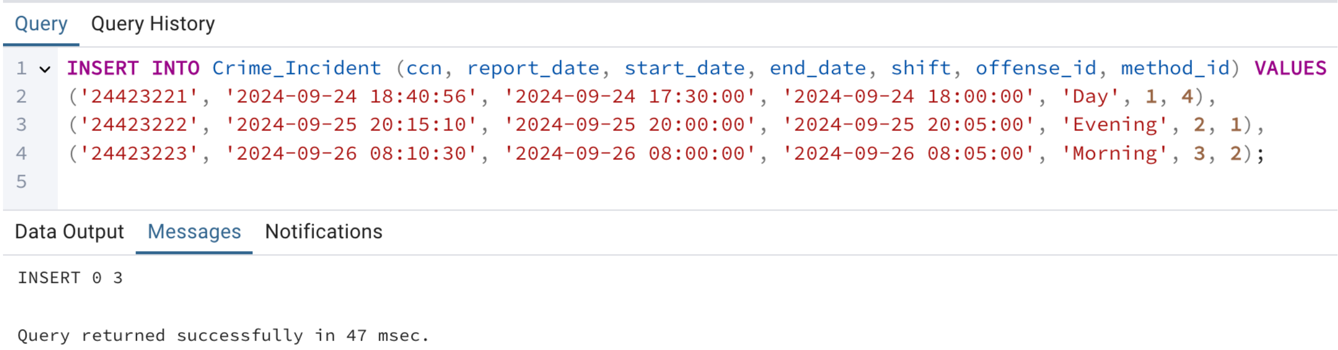
```
-- Crime_Incident Table
CREATE TABLE Crime_Incident (
    ccn VARCHAR(20) PRIMARY KEY,
    report_date TIMESTAMP,
    start_date TIMESTAMP,
    end_date TIMESTAMP,
    shift VARCHAR(20),
    offense_id INT,
    method_id INT,
    FOREIGN KEY (offense_id) REFERENCES Offense(offense_id),
    FOREIGN KEY (method_id) REFERENCES Method(method_id)
);
```

5.2.Describing the relation schemas



```
                                               Table "public.crime_incident"
   Column    |            Type             | Collation | Nullable | Default | Storage  | Compression | Stats target | Description
-------------+-----------------------------+-----------+----------+---------+----------+-------------+--------------+-------------
 ccn         | character varying(20)       |           | not null |         | extended |             |              |
 report_date | timestamp without time zone |           |          |         | plain    |             |              |
 start_date  | timestamp without time zone |           |          |         | plain    |             |              |
 end_date    | timestamp without time zone |           |          |         | plain    |             |              |
 shift       | character varying(20)       |           |          |         | extended |             |              |
 offense_id  | integer                     |           |          |         | plain    |             |              |
 method_id   | integer                     |           |          |         | plain    |             |              |
Indexes:
    "crime_incident_pkey" PRIMARY KEY, btree (ccn)
Foreign-key constraints:
    "crime_incident_method_id_fkey" FOREIGN KEY (method_id) REFERENCES method(method_id)
    "crime_incident_offense_id_fkey" FOREIGN KEY (offense_id) REFERENCES offense(offense_id)
Referenced by:
    TABLE "crime_location" CONSTRAINT "crime_location_ccn_fkey" FOREIGN KEY (ccn) REFERENCES crime_incident(ccn)
Access method: heap
```

5.3.Inserting few records

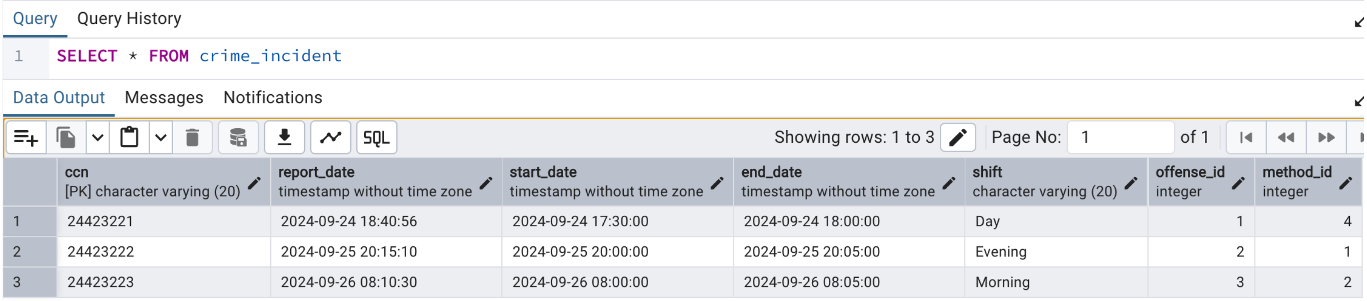


```
INSERT INTO Crime_Incident (ccn, report_date, start_date, end_date, shift, offense_id, method_id) VALUES
('24423221', '2024-09-24 18:40:56', '2024-09-24 17:30:00', '2024-09-24 18:00:00', 'Day', 1, 4),
('24423222', '2024-09-25 20:15:10', '2024-09-25 20:00:00', '2024-09-25 20:05:00', 'Evening', 2, 1),
('24423223', '2024-09-26 08:10:30', '2024-09-26 08:00:00', '2024-09-26 08:05:00', 'Morning', 3, 2);
```

```
INSERT 0 3

Query returned successfully in 47 msec.
```

5.4.Listing the contents

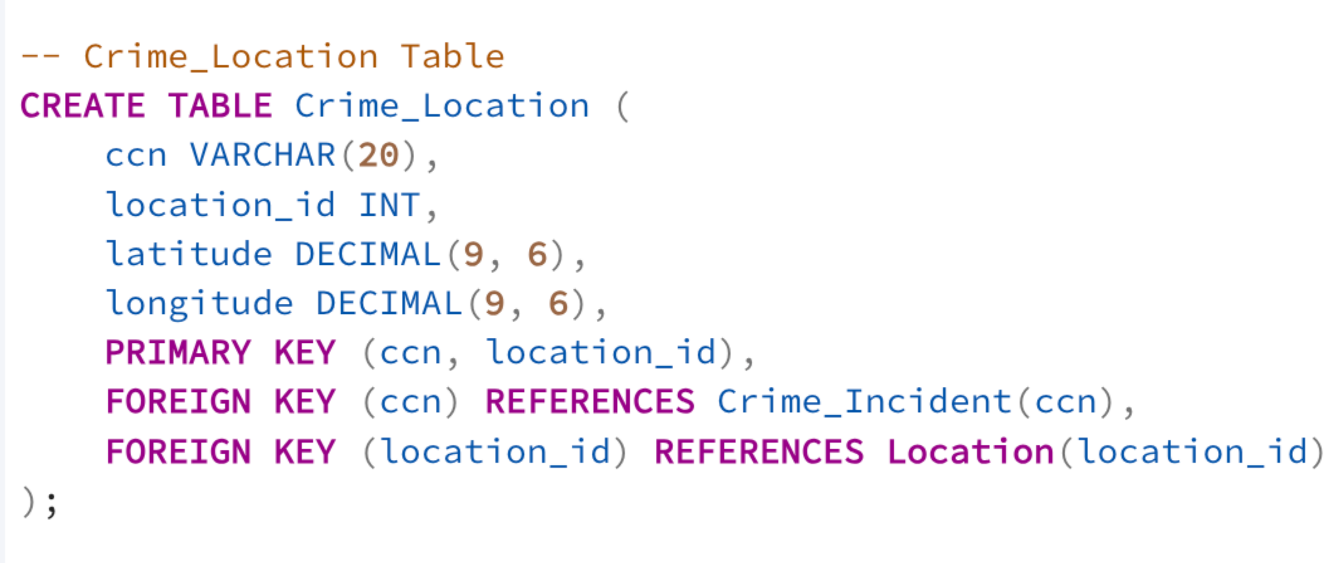


```
SELECT * FROM crime_incident
```

	ccn [PK] character varying (20)	report_date timestamp without time zone	start_date timestamp without time zone	end_date timestamp without time zone	shift character varying (20)	offense_id integer	method_id integer
1	24423221	2024-09-24 18:40:56	2024-09-24 17:30:00	2024-09-24 18:00:00	Day	1	4
2	24423222	2024-09-25 20:15:10	2024-09-25 20:00:00	2024-09-25 20:05:00	Evening	2	1
3	24423223	2024-09-26 08:10:30	2024-09-26 08:00:00	2024-09-26 08:05:00	Morning	3	2

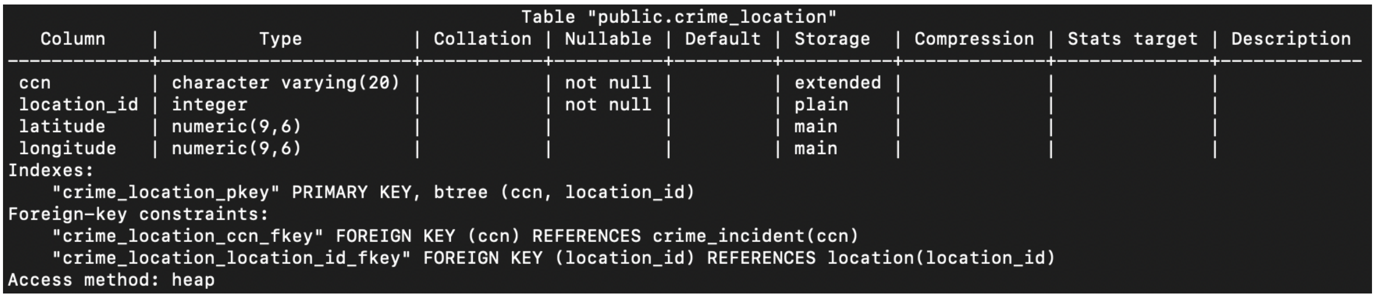
6. Crime_Location

6.1.Command to Create Table



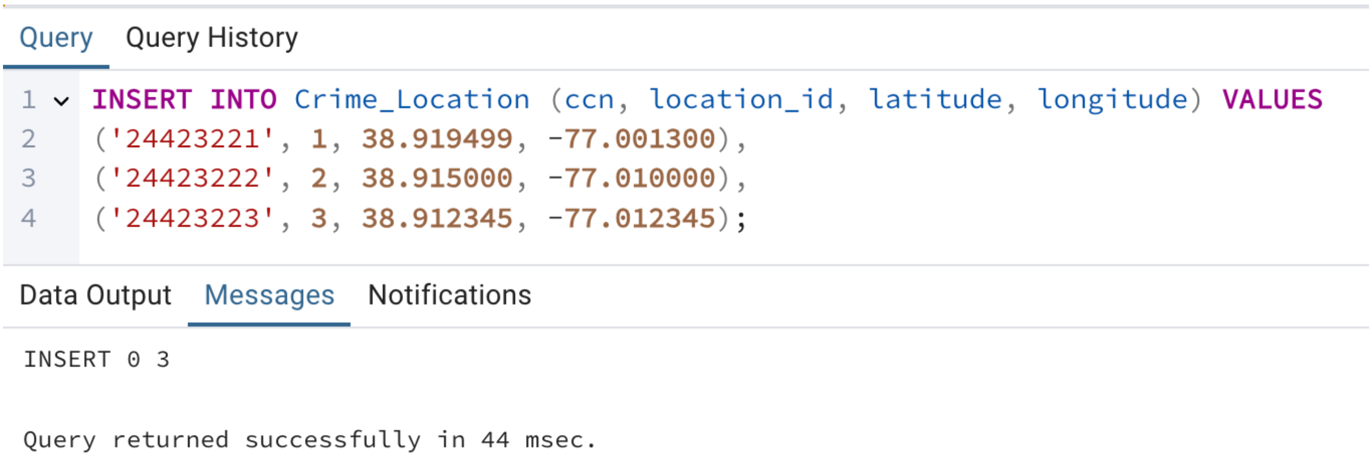
```
-- Crime_Location Table
CREATE TABLE Crime_Location (
    ccn VARCHAR(20),
    location_id INT,
    latitude DECIMAL(9, 6),
    longitude DECIMAL(9, 6),
    PRIMARY KEY (ccn, location_id),
    FOREIGN KEY (ccn) REFERENCES Crime_Incident(ccn),
    FOREIGN KEY (location_id) REFERENCES Location(location_id)
);
```

6.2.Describing the relational schemas



```
                                           Table "public.crime_location"
   Column    |         Type          | Collation | Nullable | Default | Storage  | Compression | Stats target | Description
-------------+-----------------------+-----------+----------+---------+----------+-------------+--------------+-------------
 ccn         | character varying(20) |           | not null |         | extended |             |              |
 location_id | integer               |           | not null |         | plain    |             |              |
 latitude    | numeric(9,6)          |           |          |         | main     |             |              |
 longitude   | numeric(9,6)          |           |          |         | main     |             |              |
Indexes:
    "crime_location_pkey" PRIMARY KEY, btree (ccn, location_id)
Foreign-key constraints:
    "crime_location_ccn_fkey" FOREIGN KEY (ccn) REFERENCES crime_incident(ccn)
    "crime_location_location_id_fkey" FOREIGN KEY (location_id) REFERENCES location(location_id)
Access method: heap
```

6.3.Inserting few records

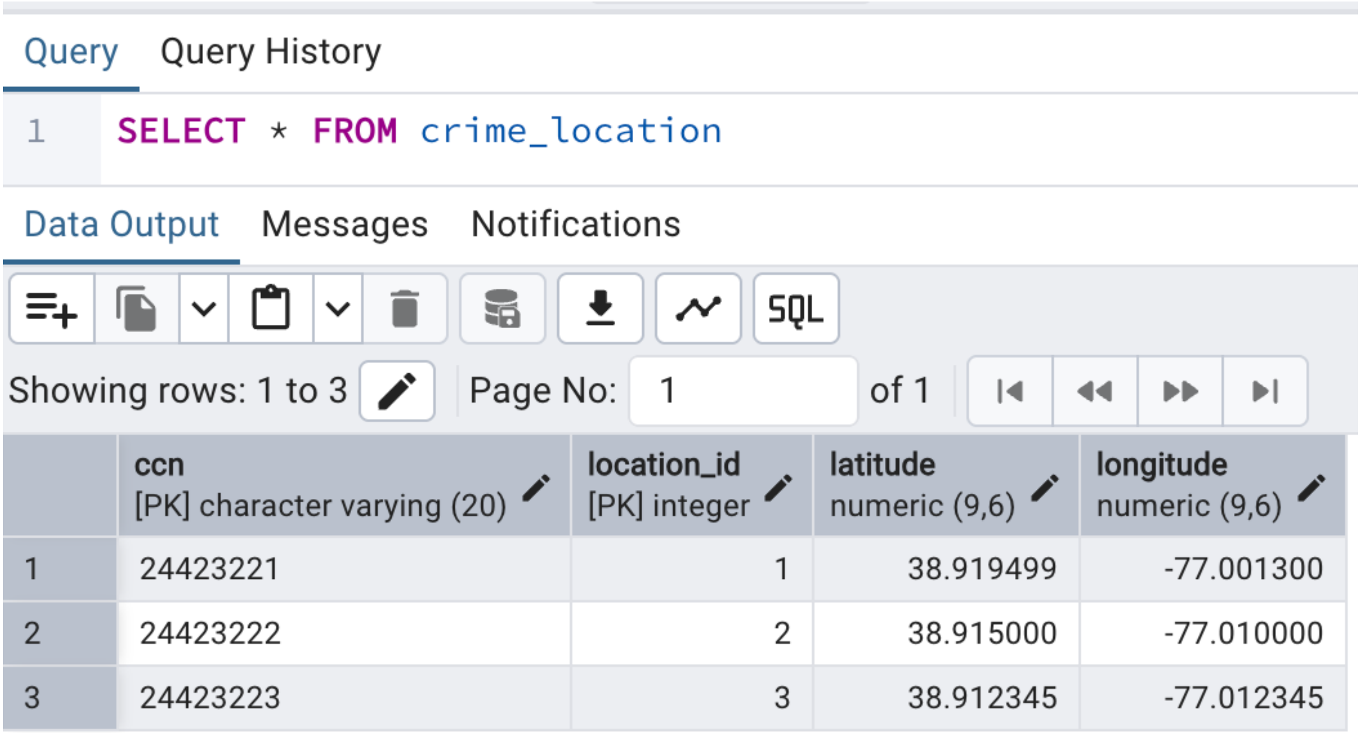


```sql
INSERT INTO Crime_Location (ccn, location_id, latitude, longitude) VALUES
('24423221', 1, 38.919499, -77.001300),
('24423222', 2, 38.915000, -77.010000),
('24423223', 3, 38.912345, -77.012345);
```

```
INSERT 0 3

Query returned successfully in 44 msec.
```

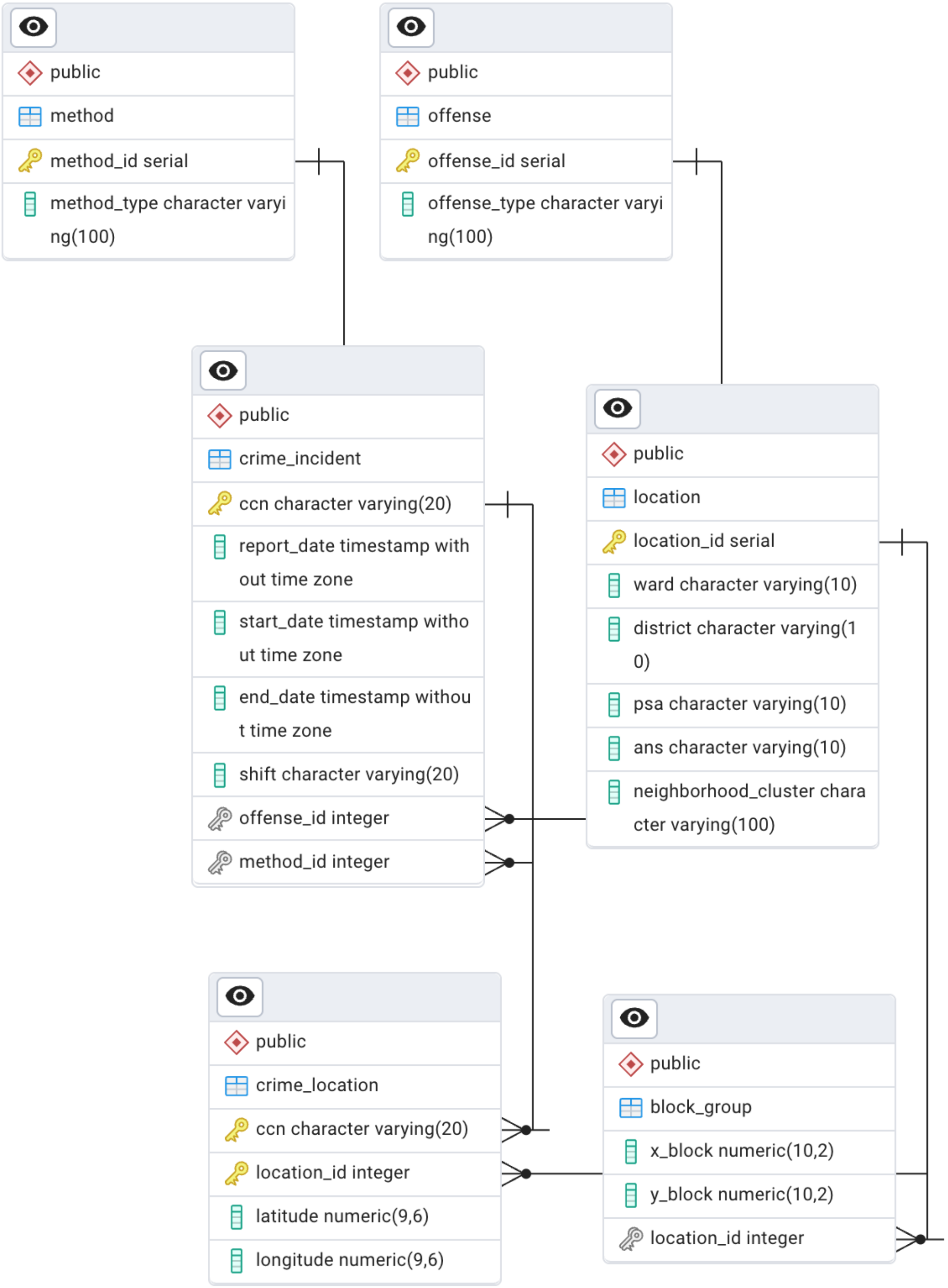
6.4.Listing the contents



```sql
SELECT * FROM crime_location
```

	ccn [PK] character varying (20)	location_id [PK] integer	latitude numeric (9,6)	longitude numeric (9,6)
1	24423221	1	38.919499	-77.001300
2	24423222	2	38.915000	-77.010000
3	24423223	3	38.912345	-77.012345

7. Schemas From PostgreSQL


public
method
method_id serial
method_type character varying(100)
public
offense
offense_id serial
offense_type character varying(100)
public
crime_incident
ccn character varying(20)
report_date timestamp without time zone
start_date timestamp without time zone
end_date timestamp without time zone
shift character varying(20)
offense_id integer
method_id integer
public
location
location_id serial
ward character varying(10)
district character varying(10)
psa character varying(10)
ans character varying(10)
neighborhood_cluster character varying(100)
public
crime_location
ccn character varying(20)
location_id integer
latitude numeric(9,6)
longitude numeric(9,6)
public
block_group
x_block numeric(10,2)
y_block numeric(10,2)
location_id integer

8. Relationships

- crime_incident → offense: Many-to-One (many incidents can be of the same offense type).
- crime_incident → method: Many-to-One (many incidents may use the same method).
- crime_incident → crime_location: One-to-Many (an incident can happen in multiple locations).
- crime_location → location: Many-to-One (multiple crime locations can belong to one location).
- location → block_group: One-to-One (each location has one block group with coordinates).