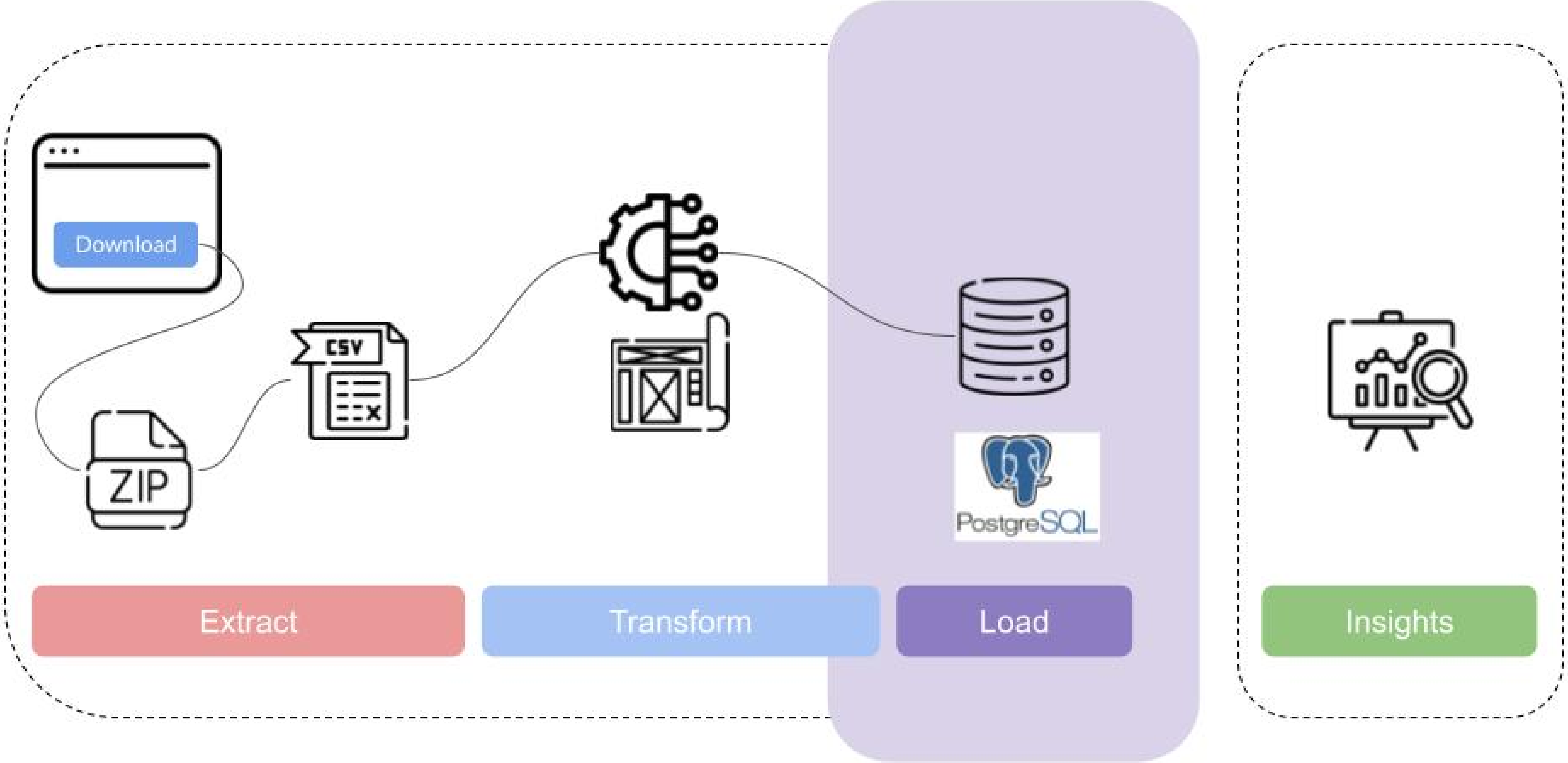
Unique key definition and clean table

ETL IN PYTHON

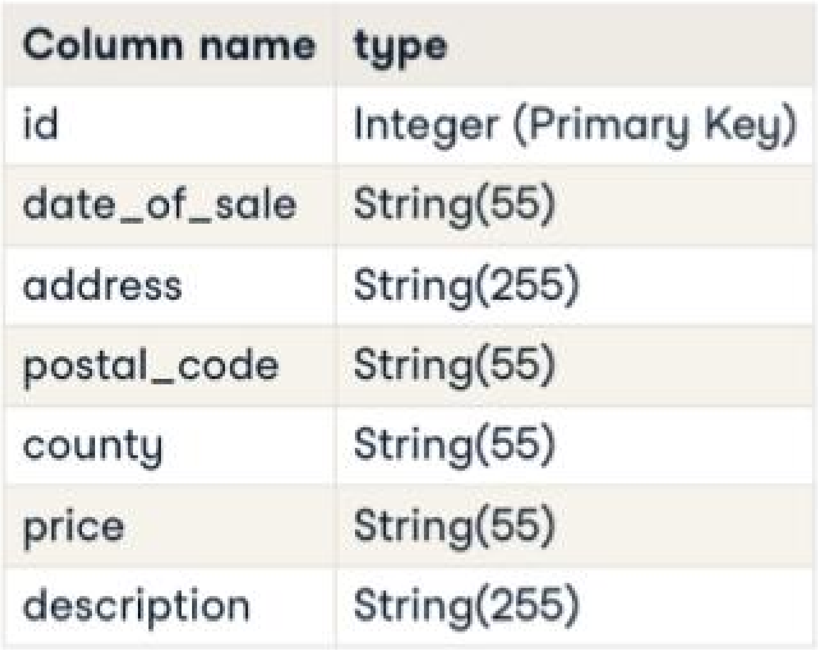
Stefano Francavilla
CEO - Geowox

Where we are in the pipeline



What it looks like

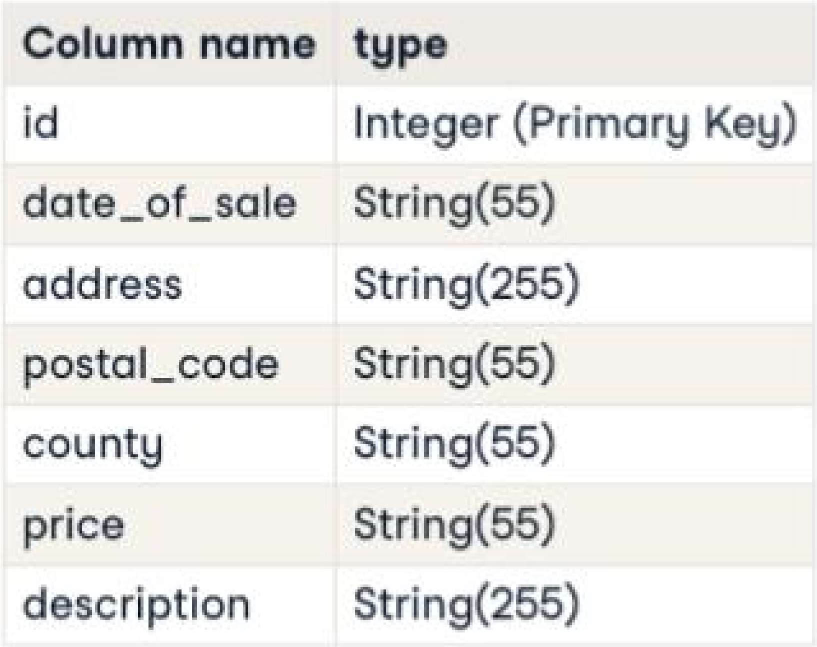
ppr_raw_all



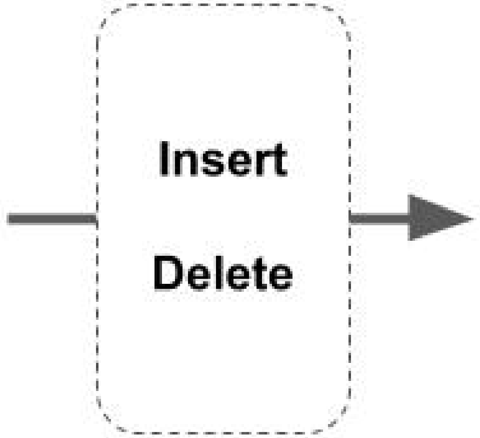
Column name	type
id	Integer (Primary Key)
date_of_sale	String(55)
address	String(255)
postal_code	String(55)
county	String(55)
price	String(55)
description	String(255)

What it looks like

ppr_raw_all



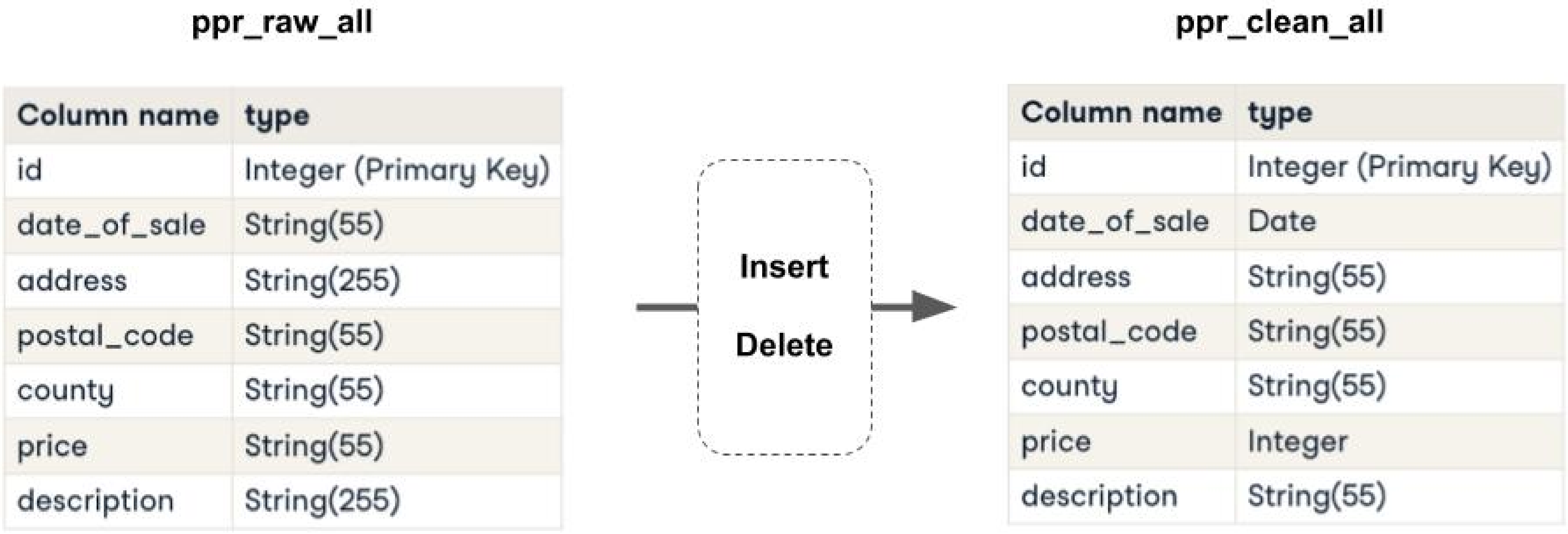
Column name	type
id	Integer (Primary Key)
date_of_sale	String(55)
address	String(255)
postal_code	String(55)
county	String(55)
price	String(55)
description	String(255)



Insert

Delete

What it looks like



ppr_raw_all

Column name	type
id	Integer (Primary Key)
date_of_sale	String(55)
address	String(255)
postal_code	String(55)
county	String(55)
price	String(55)
description	String(255)

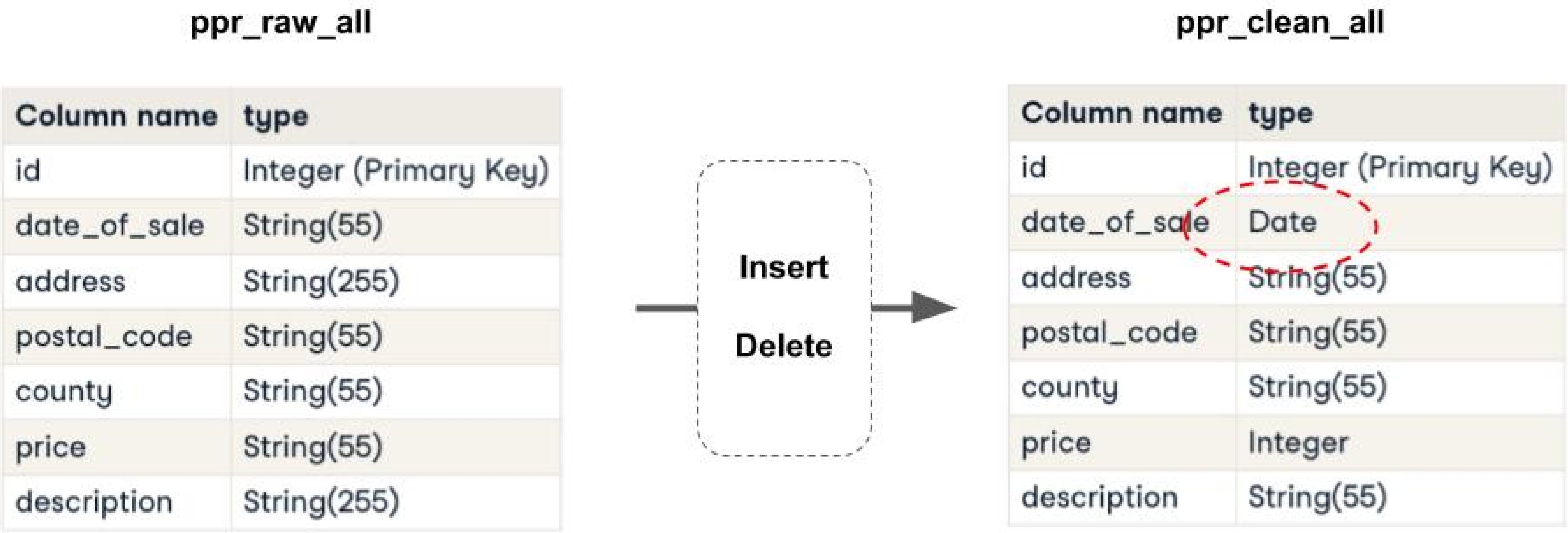
Insert

Delete

ppr_clean_all

Column name	type
id	Integer (Primary Key)
date_of_sale	Date
address	String(55)
postal_code	String(55)
county	String(55)
price	Integer
description	String(55)

What it looks like



ppr_raw_all

Column name	type
id	Integer (Primary Key)
date_of_sale	String(55)
address	String(255)
postal_code	String(55)
county	String(55)
price	String(55)
description	String(255)

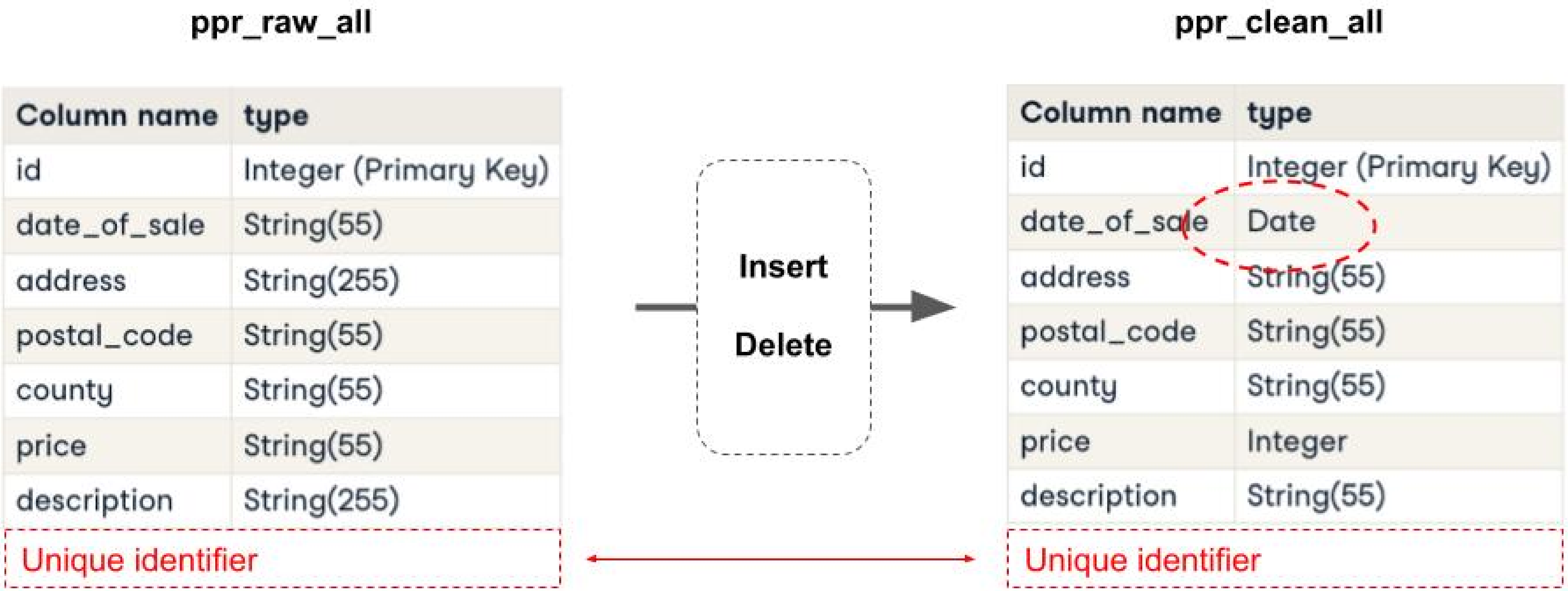
Insert

Delete

ppr_clean_all

Column name	type
id	Integer (Primary Key)
date_of_sale	Date
address	String(55)
postal_code	String(55)
county	String(55)
price	Integer
description	String(55)

What it looks like



ppr_raw_all

Column name	type
id	Integer (Primary Key)
date_of_sale	String(55)
address	String(255)
postal_code	String(55)
county	String(55)
price	String(55)
description	String(255)

ppr_clean_all

Column name	type
id	Integer (Primary Key)
date_of_sale	Date
address	String(55)
postal_code	String(55)
county	String(55)
price	Integer
description	String(55)

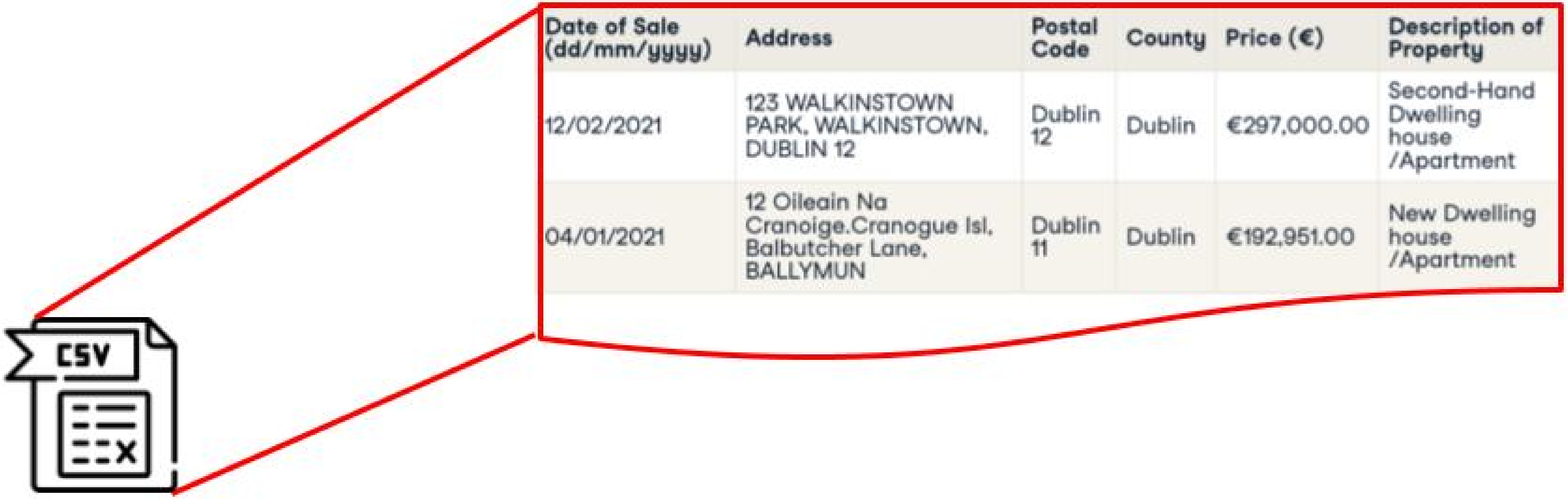
Date datatype

Table Name: `movies`

Column name	type
id	integer
title	varchar(55)
description	varchar(55)
release_date	date

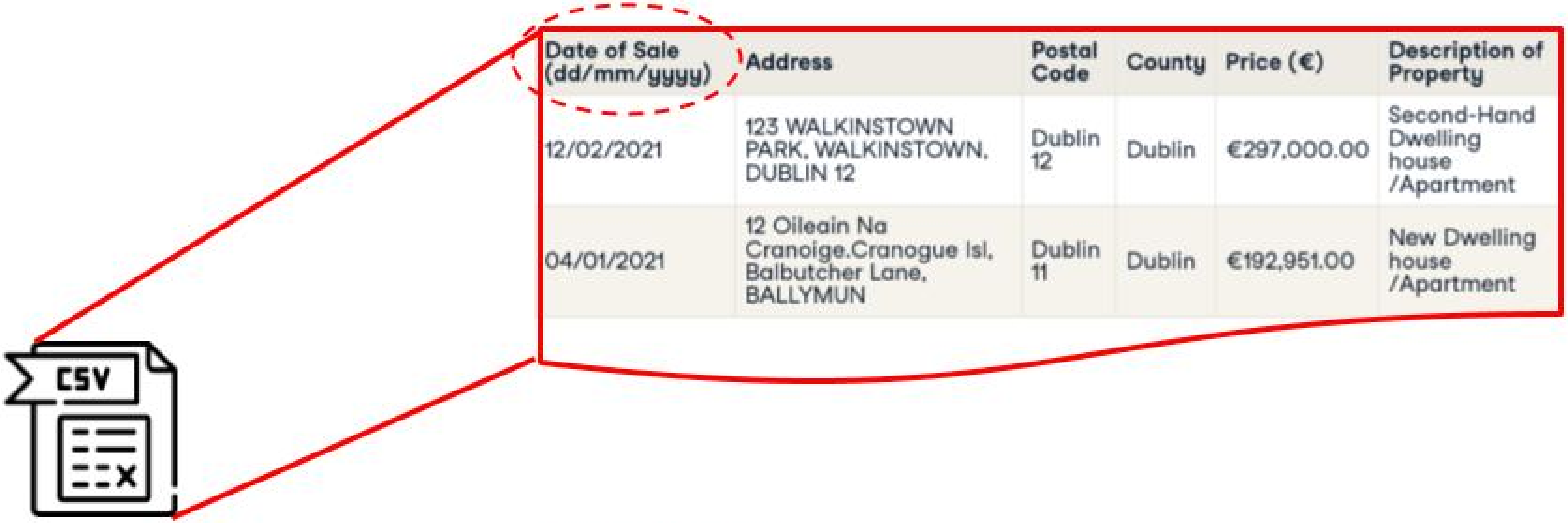
```
from sqlalchemy import Column,
                       Integer,
                       String
                       Date
class Movies(Base):
    __tablename__ = "movies"
    id = Column(Integer)
    title = Column(String(55))
    description = Column(String(255))

    release_date = Column(Date)
```

Uniqueness



Date of Sale (dd/mm/yyyy)	Address	Postal Code	County	Price (€)	Description of Property
12/02/2021	123 WALKINSTOWN PARK, WALKINSTOWN, DUBLIN 12	Dublin 12	Dublin	€297,000.00	Second-Hand Dwelling house /Apartment
04/01/2021	12 Oileain Na Cranoige.Cranogue Isl, Balbutcher Lane, BALLYMUN	Dublin 11	Dublin	€192,951.00	New Dwelling house /Apartment

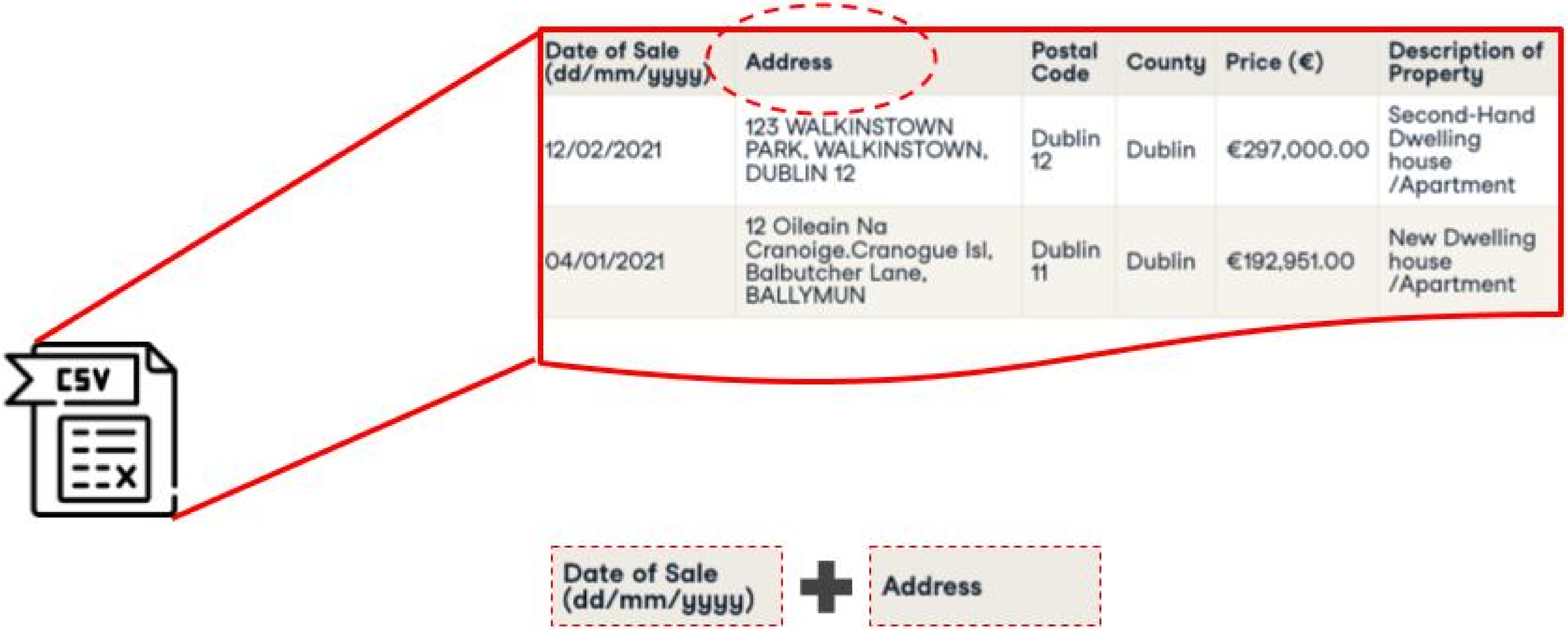
Uniqueness



Date of Sale (dd/mm/yyyy)	Address	Postal Code	County	Price (€)	Description of Property
12/02/2021	123 WALKINSTOWN PARK, WALKINSTOWN, DUBLIN 12	Dublin 12	Dublin	€297,000.00	Second-Hand Dwelling house /Apartment
04/01/2021	12 Oileain Na Cranoige.Cranogue Isl, Balbutcher Lane, BALLYMUN	Dublin 11	Dublin	€192,951.00	New Dwelling house /Apartment

Date of Sale (dd/mm/yyyy)

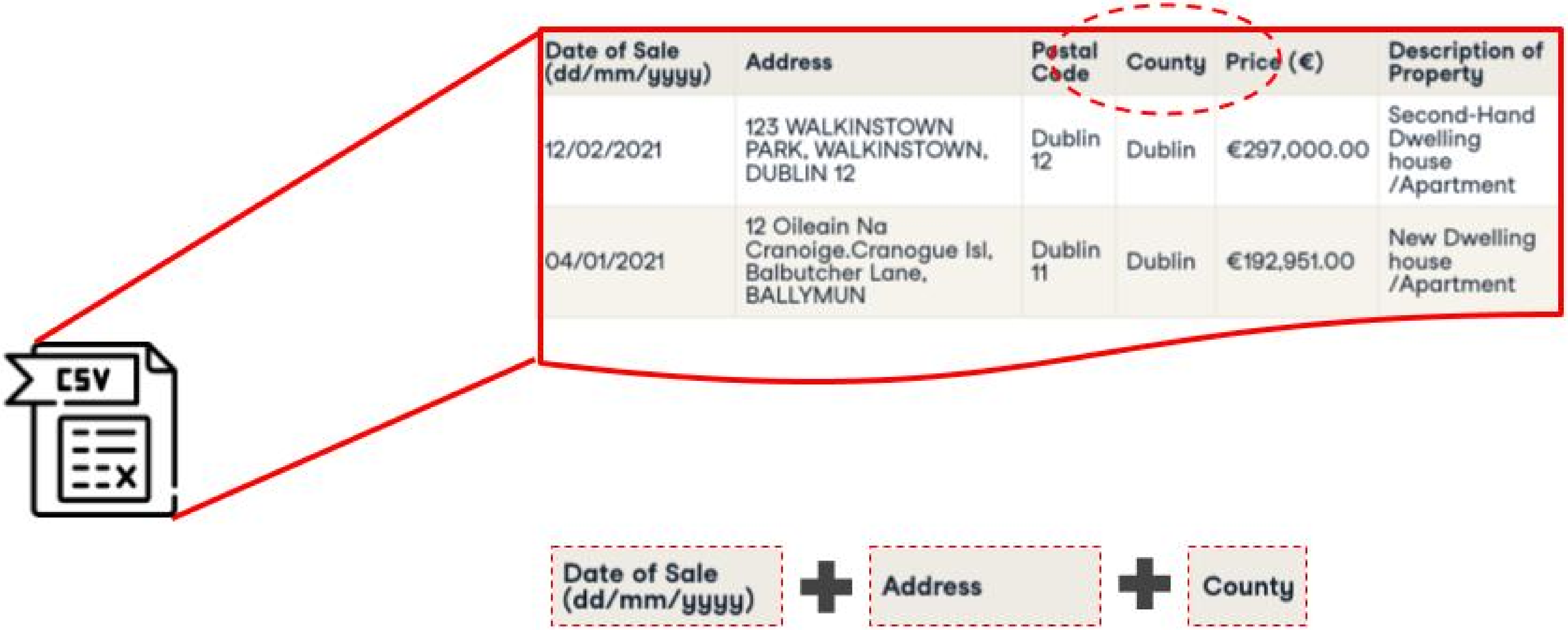
Uniqueness



Date of Sale (dd/mm/yyyy)	Address	Postal Code	County	Price (€)	Description of Property
12/02/2021	123 WALKINSTOWN PARK, WALKINSTOWN, DUBLIN 12	Dublin 12	Dublin	€297,000.00	Second-Hand Dwelling house /Apartment
04/01/2021	12 Oileain Na Cranoige.Cranogue Isl, Balbutcher Lane, BALLYMUN	Dublin 11	Dublin	€192,951.00	New Dwelling house /Apartment

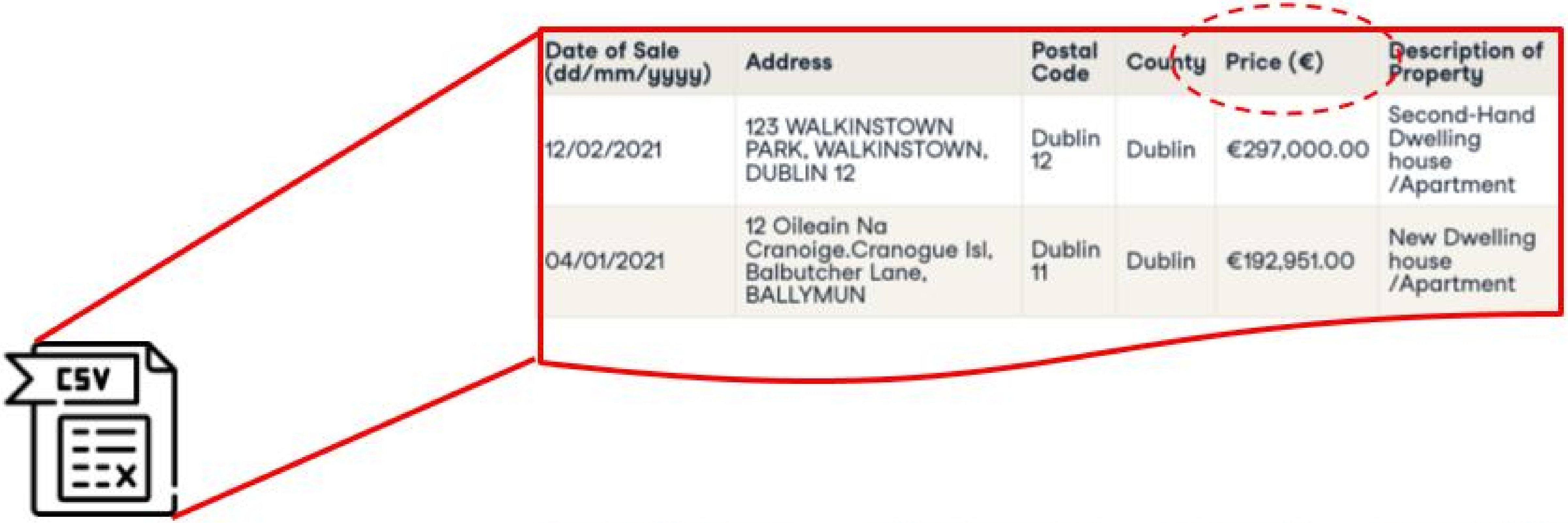
Date of Sale (dd/mm/yyyy) + Address

Uniqueness



Date of Sale (dd/mm/yyyy)	Address	Postal Code	County	Price (€)	Description of Property
12/02/2021	123 WALKINSTOWN PARK, WALKINSTOWN, DUBLIN 12	Dublin 12	Dublin	€297,000.00	Second-Hand Dwelling house /Apartment
04/01/2021	12 Oileain Na Cranoige.Cranogue Isl, Balbutcher Lane, BALLYMUN	Dublin 11	Dublin	€192,951.00	New Dwelling house /Apartment

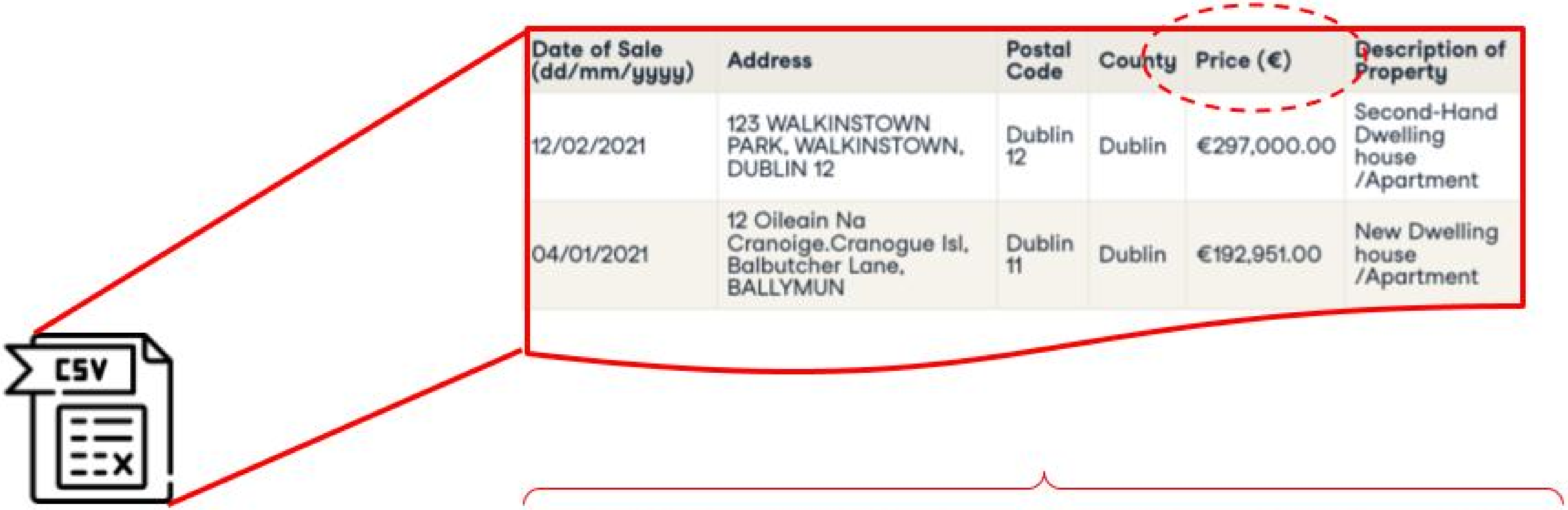
Uniqueness



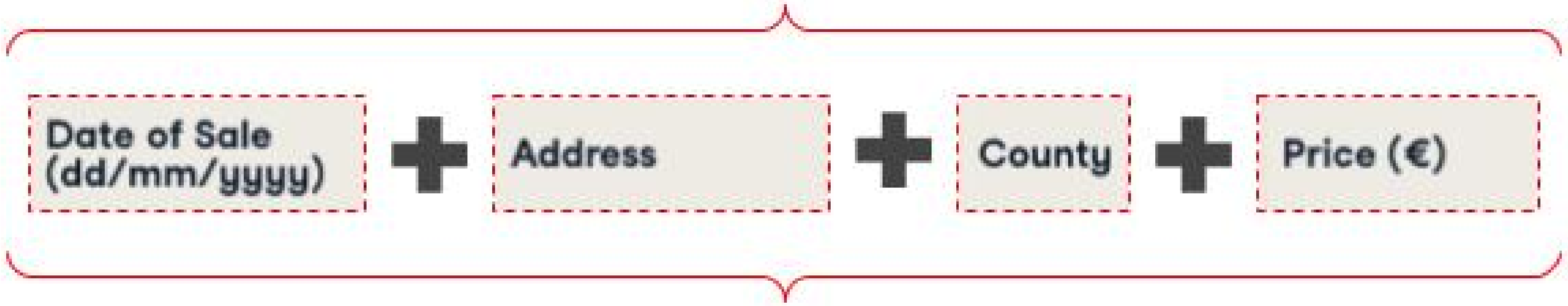
Date of Sale (dd/mm/yyyy)	Address	Postal Code	County	Price (€)	Description of Property
12/02/2021	123 WALKINSTOWN PARK, WALKINSTOWN, DUBLIN 12	Dublin 12	Dublin	€297,000.00	Second-Hand Dwelling house /Apartment
04/01/2021	12 Oileain Na Cranoige.Cranogue Isl, Balbutcher Lane, BALLYMUN	Dublin 11	Dublin	€192,951.00	New Dwelling house /Apartment

Date of Sale (dd/mm/yyyy) ➕ Address ➕ County ➕ Price (€)

Uniqueness



Date of Sale (dd/mm/yyyy)	Address	Postal Code	County	Price (€)	Description of Property
12/02/2021	123 WALKINSTOWN PARK, WALKINSTOWN, DUBLIN 12	Dublin 12	Dublin	€297,000.00	Second-Hand Dwelling house /Apartment
04/01/2021	12 Oileain Na Cranoige.Cranogue Isl, Balbutcher Lane, BALLYMUN	Dublin 11	Dublin	€192,951.00	New Dwelling house /Apartment

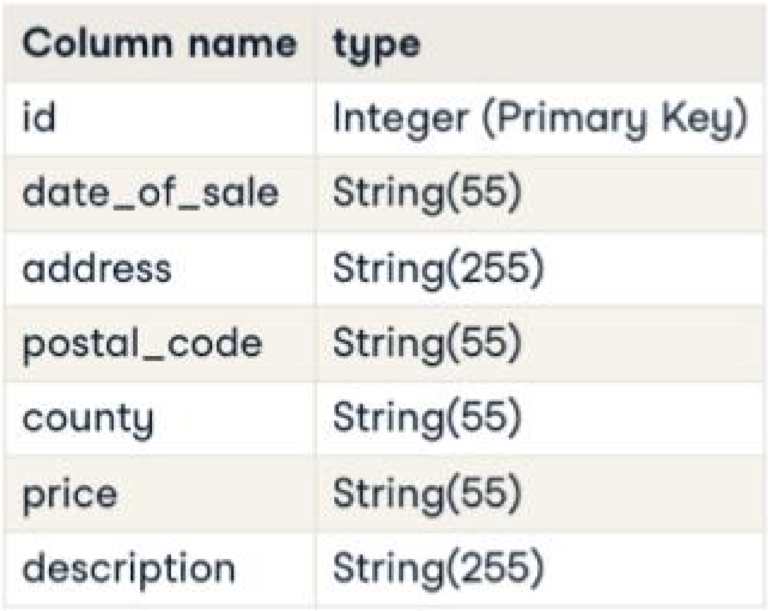


Date of Sale (dd/mm/yyyy) + Address + County + Price (€)

Unique Key

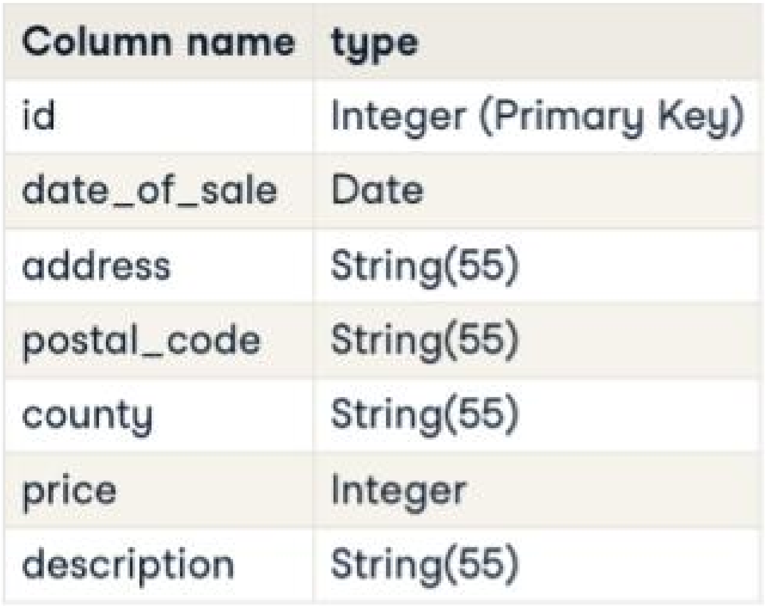
Column property

ppr_raw_all



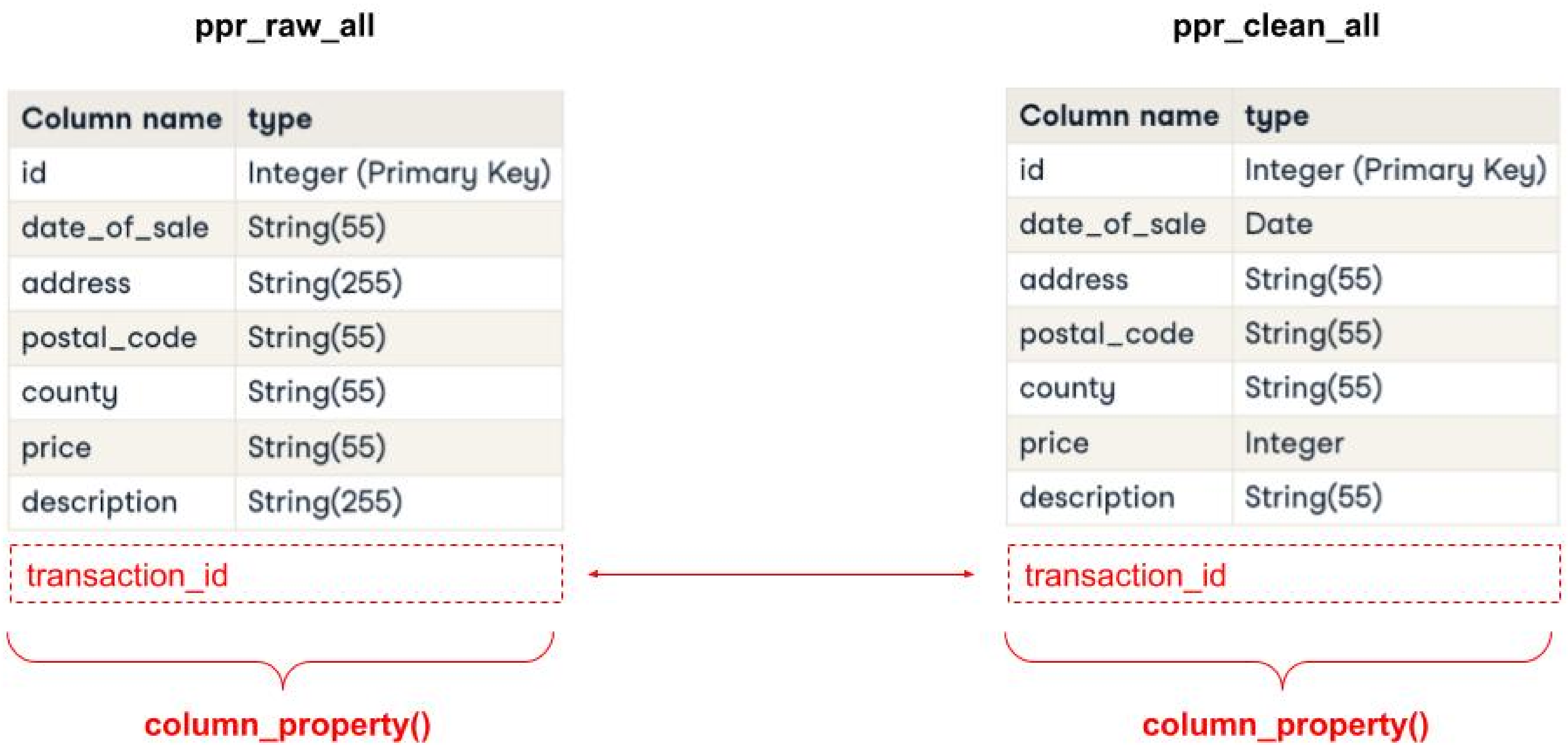
Column name	type
id	Integer (Primary Key)
date_of_sale	String(55)
address	String(255)
postal_code	String(55)
county	String(55)
price	String(55)
description	String(255)

ppr_clean_all



Column name	type
id	Integer (Primary Key)
date_of_sale	Date
address	String(55)
postal_code	String(55)
county	String(55)
price	Integer
description	String(55)

Column property



ppr_raw_all

Column name	type
id	Integer (Primary Key)
date_of_sale	String(55)
address	String(255)
postal_code	String(55)
county	String(55)
price	String(55)
description	String(255)
transaction_id	

column_property()

ppr_clean_all

Column name	type
id	Integer (Primary Key)
date_of_sale	Date
address	String(55)
postal_code	String(55)
county	String(55)
price	Integer
description	String(55)
transaction_id	

column_property()

Column property

- `from sqlalchemy.orm import column_property`
- Loaded at load time

Column property: example

```python
from sqlalchemy.orm import column_property
class User(Base):
    __tablename__ = 'user'
    id = Column(Integer, primary_key=True)
    firstname = Column(String(50))
    lastname = Column(String(50))
    fullname = column_property(firstname + " " + lastname)

    user = User(firstname="John", lastname="Smith")
    print("User:", user.fullname)
```

```
User: John Smith
```

Let's practice!

ETL IN PYTHON

Insert and delete operations

ETL IN PYTHON

Stefano Francavilla
CEO - Geowox

Query API

- `SELECT * FROM movies`
- `session.query(Movies)`
- `session.query(Movies).all()`

Query API: an example

Table name: movies

id	title
1	The Big Short
2	The Social Network
3	The Avengers

```python
from sqlalchemy import Column, Integer
from sqlalchemy.orm import
                    declarative_base

Base = declarative_base()

class Movies(Base):
    __tablename__ = "movies"
    id = Column(Integer,
                primary_key=True)
    title = Column(String(50))
```

Query API: an example

```sql
SELECT * FROM movies
```

```python
session = Session(engine)
result = session.query(Movies).all()
for row in result:
    print("Title: ", row.title)
```

```
Title: The Big Short
Title: The Social Network
Title: The Avengers
```

Query API: an example

```sql
SELECT * FROM movies
WHERE id=1
```

```python
session = Session(engine)

result = session.query(Movies)
                .filter(Movies.id == 1)

for row in result:
    print("Title: ",row.title)
```

```
Title: The Big Short
```

Delete

- `session.query().filter()`
- `session.query().filter().delete()`
- `session.query(Movies).filter(Movies.title == "The Big Short").delete()`
- `session.query(Movies).filter(Movies.title == "").delete()`

Insert

```python
from sqlalchemy.dialects.postgresql import insert

values = [{"title": "Luca"}, {"title": "The Lord of the Rings"}]

insert(Movies).values(values)
```

Commit into table

```
stm = delete(Movies).filter(Movies.id == 1)

session.execute(stm)

session.commit()
```

Let's practice!

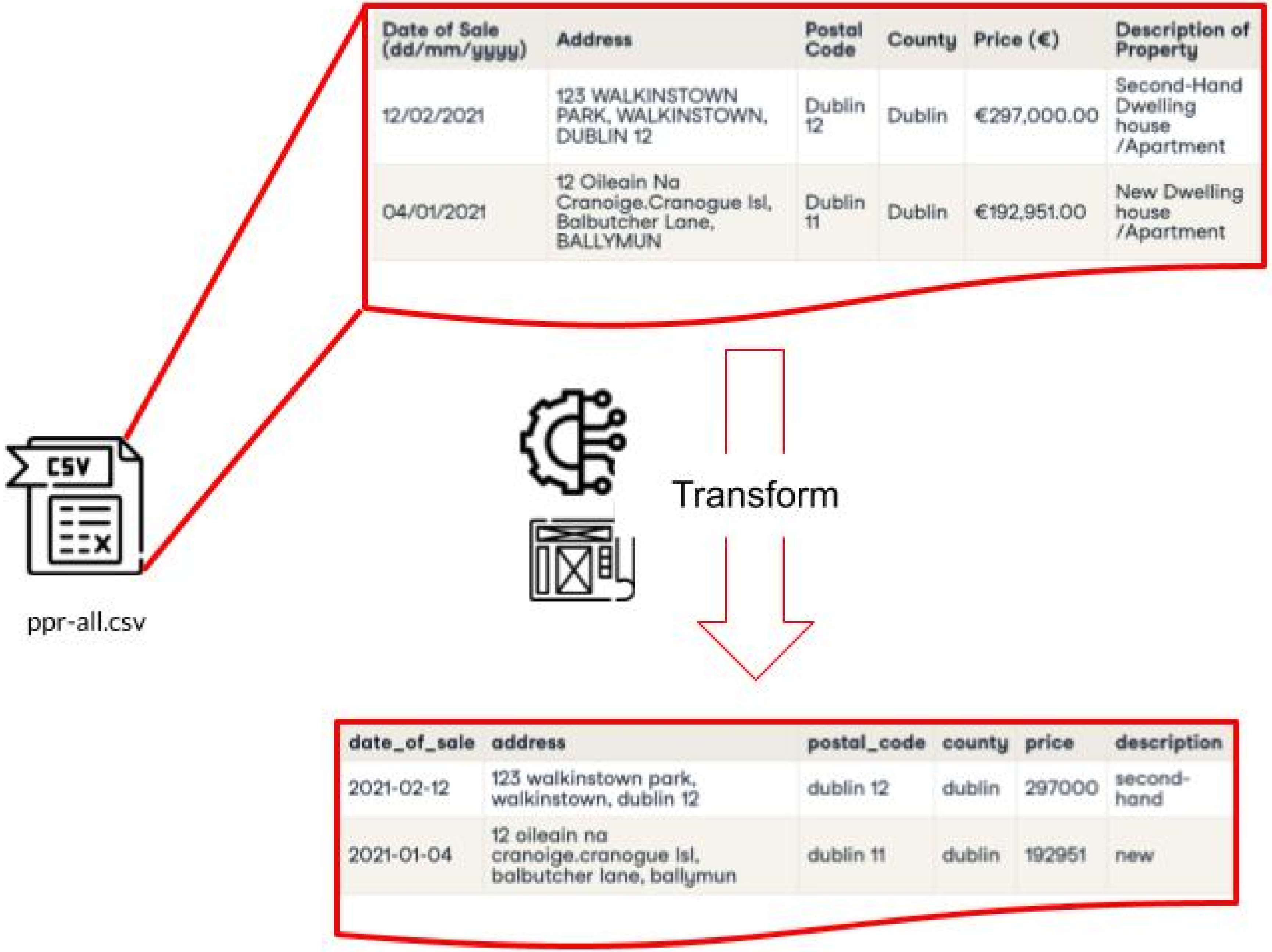
ETL IN PYTHON

Put load operations together

ETL IN PYTHON

Stefano Francavilla
CEO - Geowox

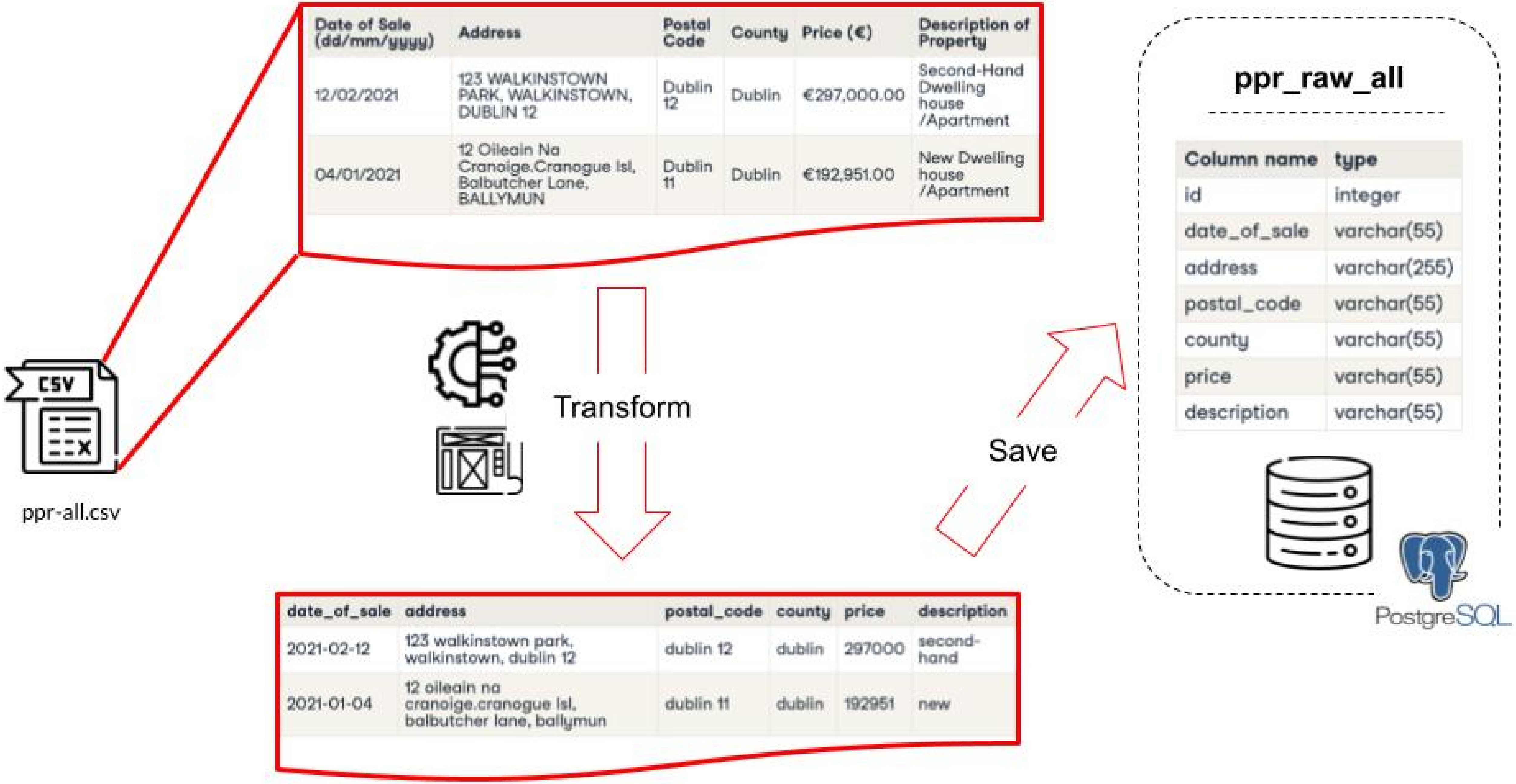
Where we have left



Date of Sale (dd/mm/yyyy)	Address	Postal Code	County	Price (€)	Description of Property
12/02/2021	123 WALKINSTOWN PARK, WALKINSTOWN, DUBLIN 12	Dublin 12	Dublin	€297,000.00	Second-Hand Dwelling house /Apartment
04/01/2021	12 Oileain Na Cranoige.Cranogue Isl, Balbutcher Lane, BALLYMUN	Dublin 11	Dublin	€192,951.00	New Dwelling house /Apartment

date_of_sale	address	postal_code	county	price	description
2021-02-12	123 walkinstown park, walkinstown, dublin 12	dublin 12	dublin	297000	second-hand
2021-01-04	12 oileain na cranoige.cranogue Isl, balbutcher lane, ballymun	dublin 11	dublin	192951	new

Where we have left



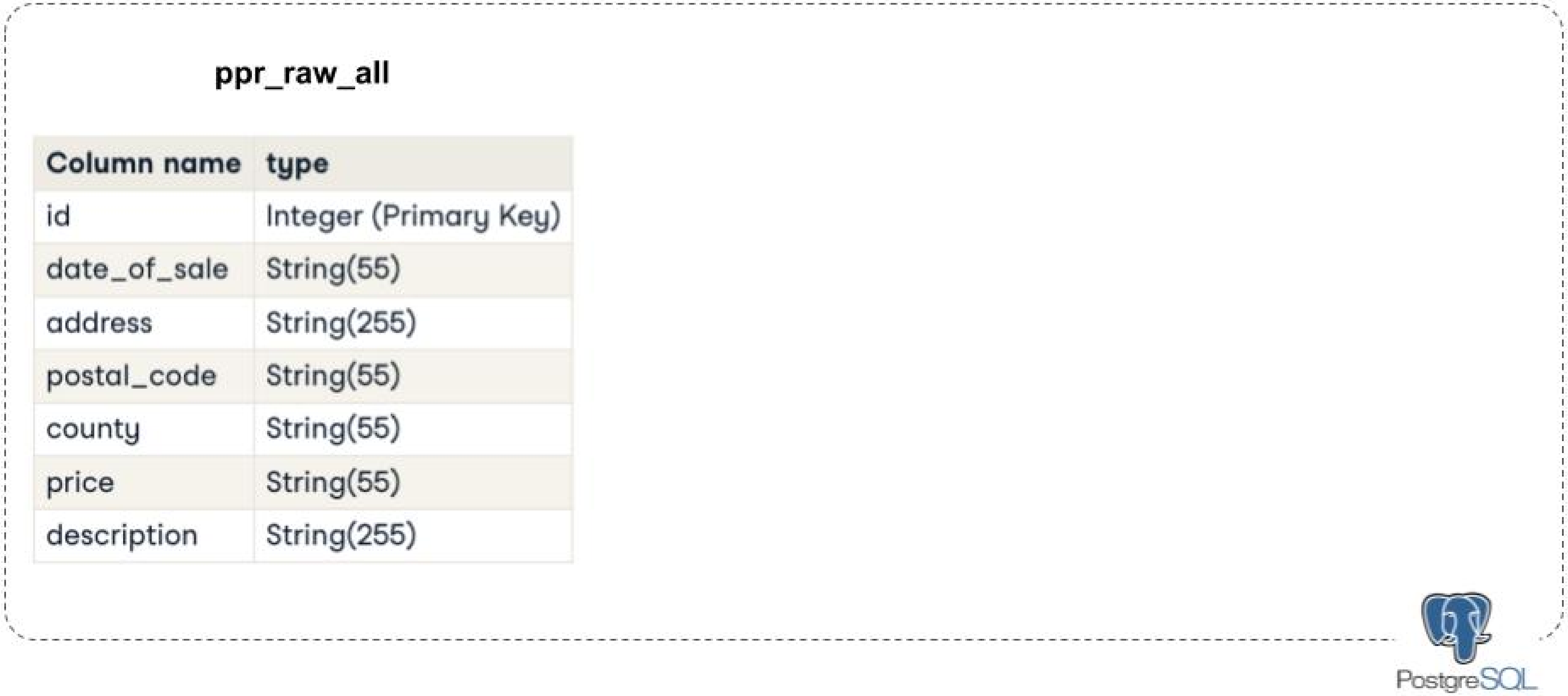
Date of Sale (dd/mm/yyyy)	Address	Postal Code	County	Price (€)	Description of Property
12/02/2021	123 WALKINSTOWN PARK, WALKINSTOWN, DUBLIN 12	Dublin 12	Dublin	€297,000.00	Second-Hand Dwelling house /Apartment
04/01/2021	12 Oileain Na Cranoige.Cranogue Isl, Balbutcher Lane, BALLYMUN	Dublin 11	Dublin	€192,951.00	New Dwelling house /Apartment

date_of_sale	address	postal_code	county	price	description
2021-02-12	123 walkinstown park, walkinstown, dublin 12	dublin 12	dublin	297000	second-hand
2021-01-04	12 oileain na cranoige.cranogue Isl, balbutcher lane, ballymun	dublin 11	dublin	192951	new

ppr_raw_all

Column name	type
id	integer
date_of_sale	varchar(55)
address	varchar(255)
postal_code	varchar(55)
county	varchar(55)
price	varchar(55)
description	varchar(55)

ETL(oad)

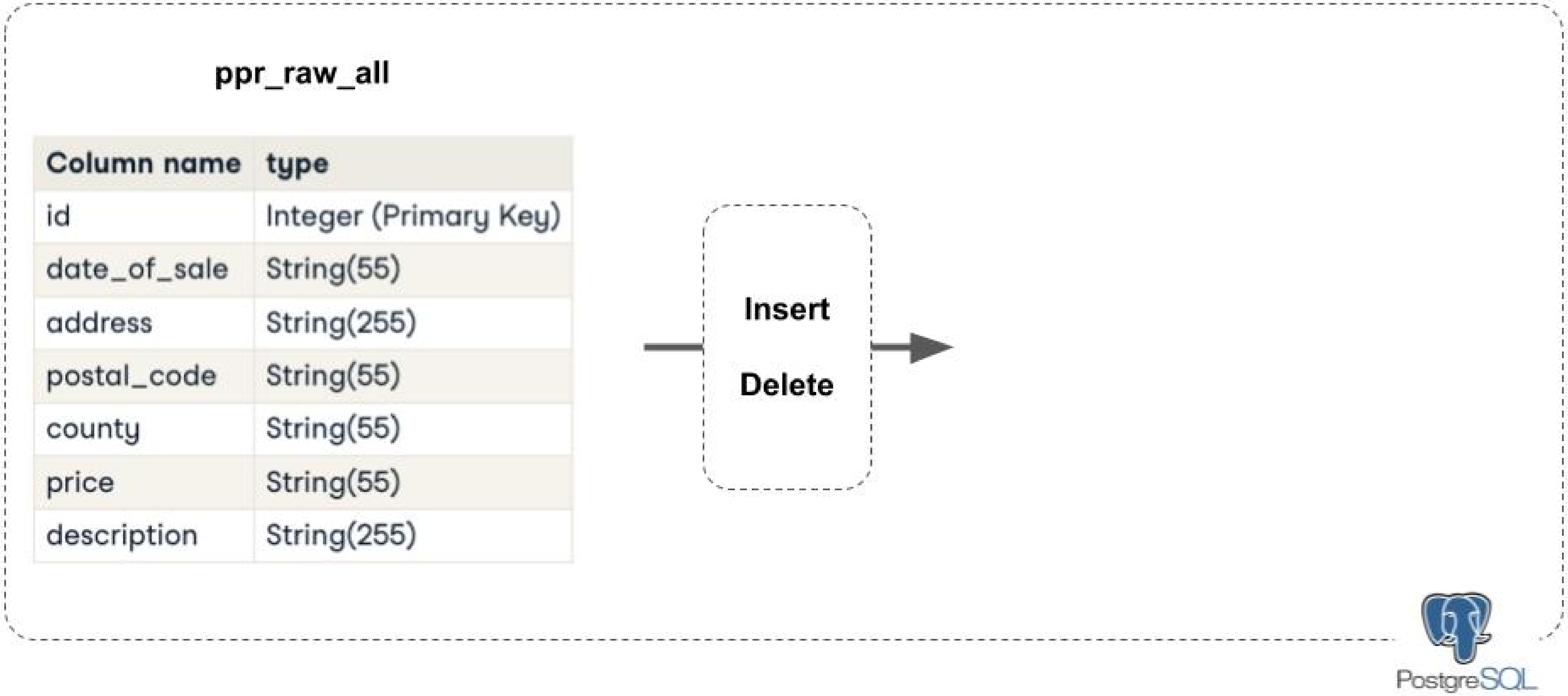


ppr_raw_all

Column name	type
id	Integer (Primary Key)
date_of_sale	String(55)
address	String(255)
postal_code	String(55)
county	String(55)
price	String(55)
description	String(255)

PostgreSQL

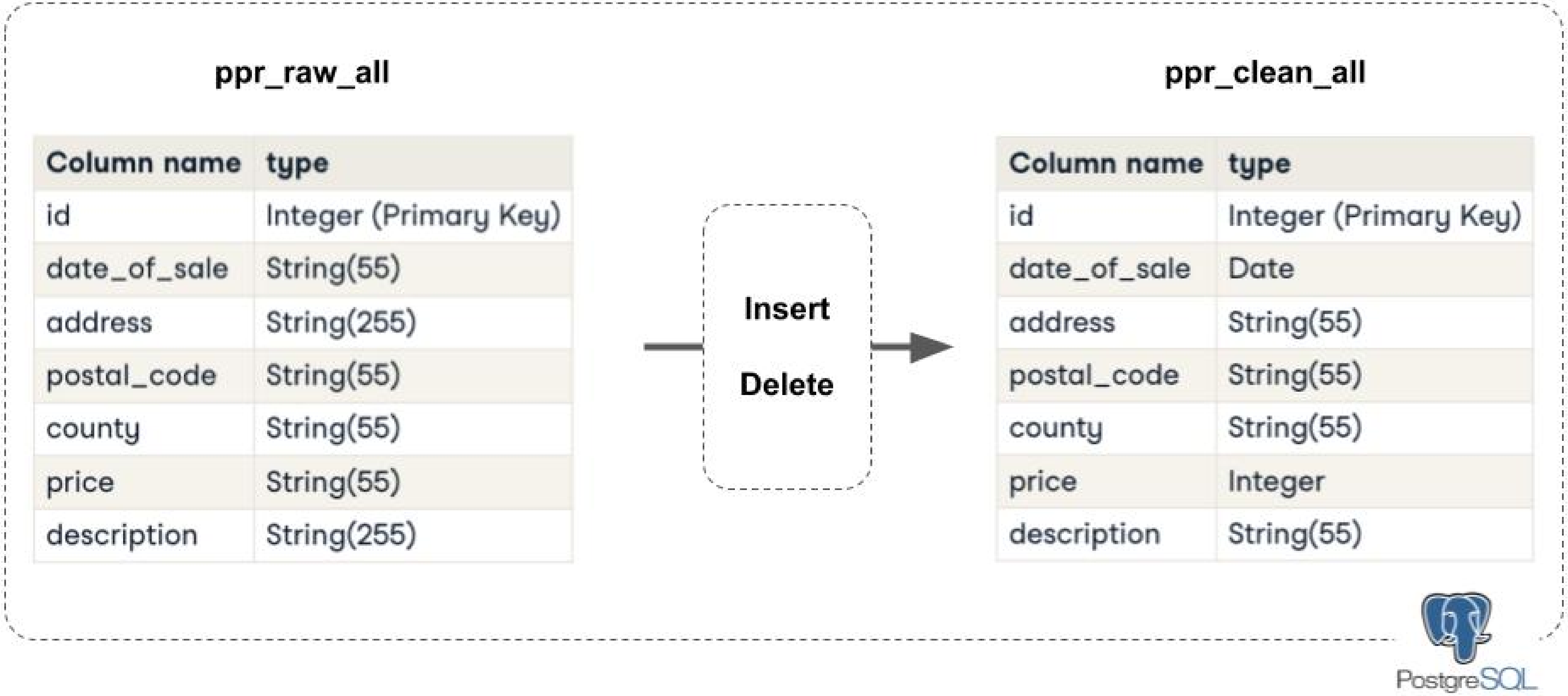
ETL(oad)



ppr_raw_all

Column name	type
id	Integer (Primary Key)
date_of_sale	String(55)
address	String(255)
postal_code	String(55)
county	String(55)
price	String(55)
description	String(255)

ETL(oad)



ppr_raw_all

Column name	type
id	Integer (Primary Key)
date_of_sale	String(55)
address	String(255)
postal_code	String(55)
county	String(55)
price	String(55)
description	String(255)

Insert

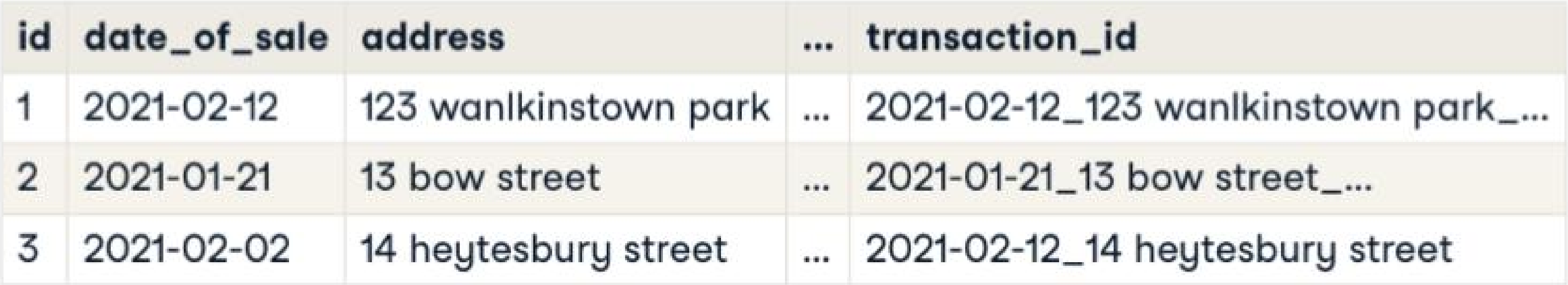
Delete

ppr_clean_all

Column name	type
id	Integer (Primary Key)
date_of_sale	Date
address	String(55)
postal_code	String(55)
county	String(55)
price	Integer
description	String(55)

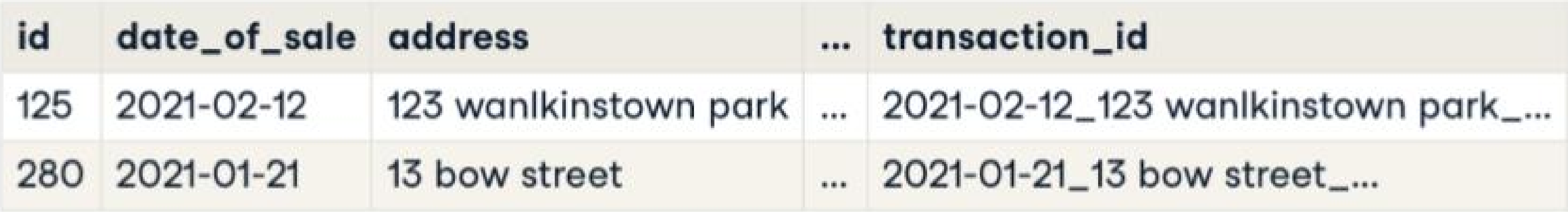
Insert

ppr_raw_all



id	date_of_sale	address	...	transaction_id
1	2021-02-12	123 wanlkinstown park	...	2021-02-12_123 wanlkinstown park_...
2	2021-01-21	13 bow street	...	2021-01-21_13 bow street_...
3	2021-02-02	14 heytesbury street	...	2021-02-12_14 heytesbury street

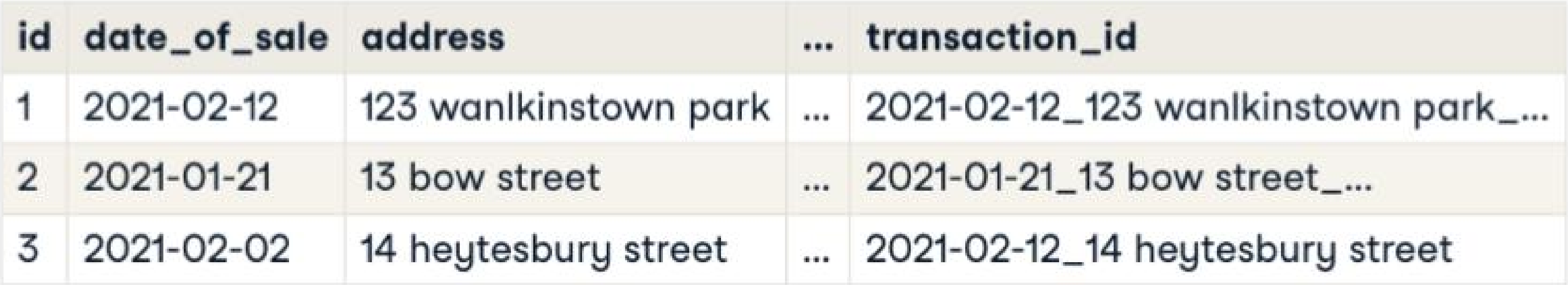
ppr_clean_all



id	date_of_sale	address	...	transaction_id
125	2021-02-12	123 wanlkinstown park	...	2021-02-12_123 wanlkinstown park_...
280	2021-01-21	13 bow street	...	2021-01-21_13 bow street_...

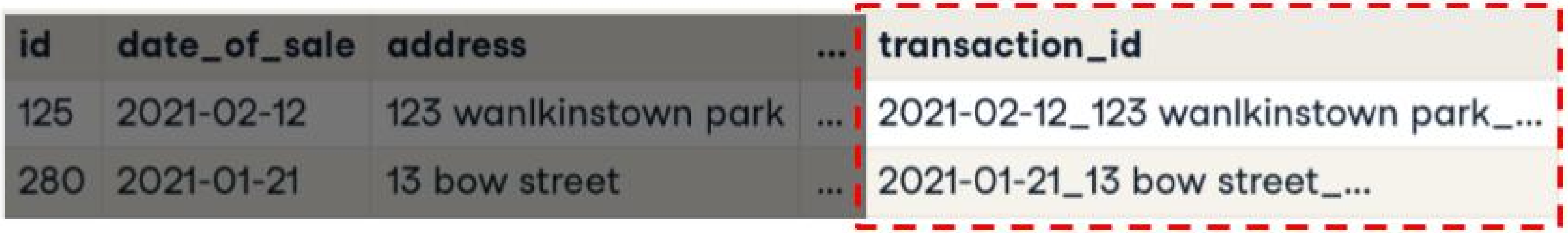
Insert

ppr_raw_all



id	date_of_sale	address	...	transaction_id
1	2021-02-12	123 wanlkinstown park	...	2021-02-12_123 wanlkinstown park_...
2	2021-01-21	13 bow street	...	2021-01-21_13 bow street_...
3	2021-02-02	14 heytesbury street	...	2021-02-12_14 heytesbury street

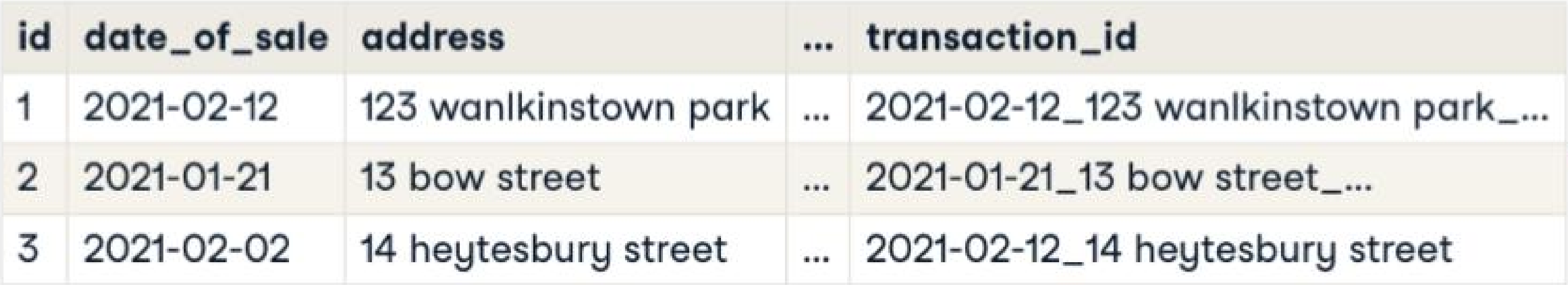
ppr_clean_all



id	date_of_sale	address	...	transaction_id
125	2021-02-12	123 wanlkinstown park	...	2021-02-12_123 wanlkinstown park_...
280	2021-01-21	13 bow street	...	2021-01-21_13 bow street_...

Insert

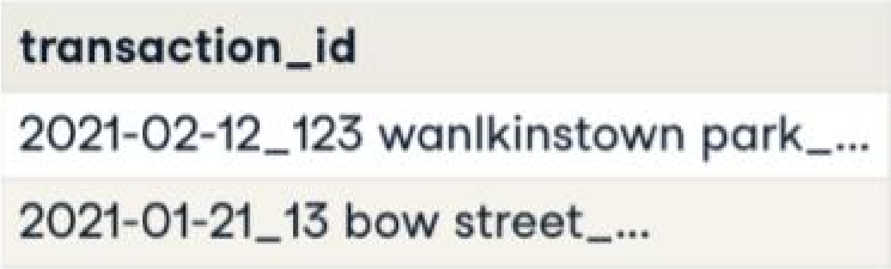
ppr_raw_all



id	date_of_sale	address	...	transaction_id
1	2021-02-12	123 wanlkinstown park	...	2021-02-12_123 wanlkinstown park_...
2	2021-01-21	13 bow street	...	2021-01-21_13 bow street_...
3	2021-02-02	14 heytesbury street	...	2021-02-12_14 heytesbury street

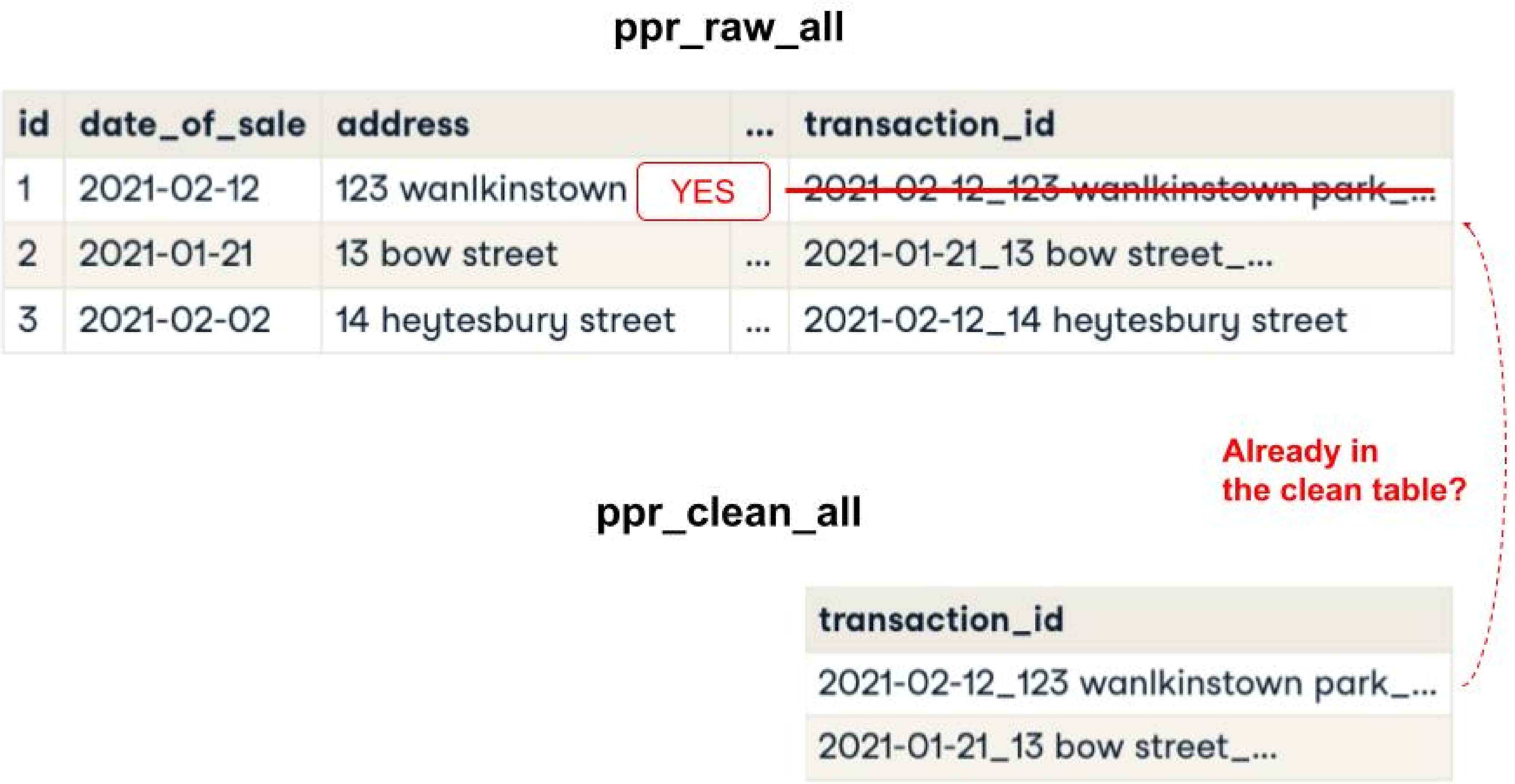
Already in
the clean table?

ppr_clean_all



transaction_id
2021-02-12_123 wanlkinstown park_...
2021-01-21_13 bow street_...

Insert



ppr_raw_all

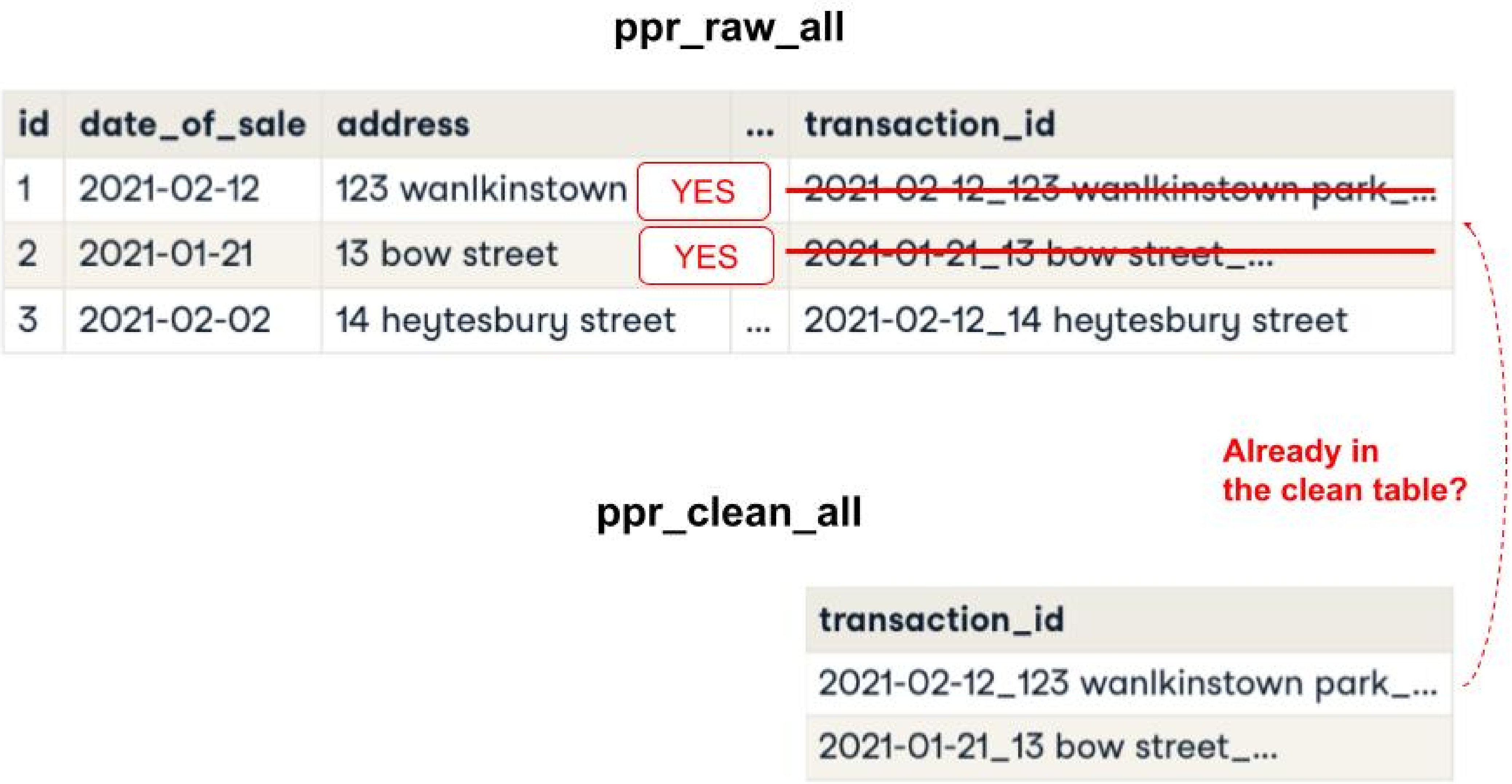
id	date_of_sale	address	...	transaction_id
1	2021-02-12	123 wanlkinstown	YES	~~2021-02-12_123 wanlkinstown park_...~~
2	2021-01-21	13 bow street	...	2021-01-21_13 bow street_...
3	2021-02-02	14 heytesbury street	...	2021-02-12_14 heytesbury street

Already in
the clean table?

ppr_clean_all

transaction_id
2021-02-12_123 wanlkinstown park_...
2021-01-21_13 bow street_...

Insert



ppr_raw_all

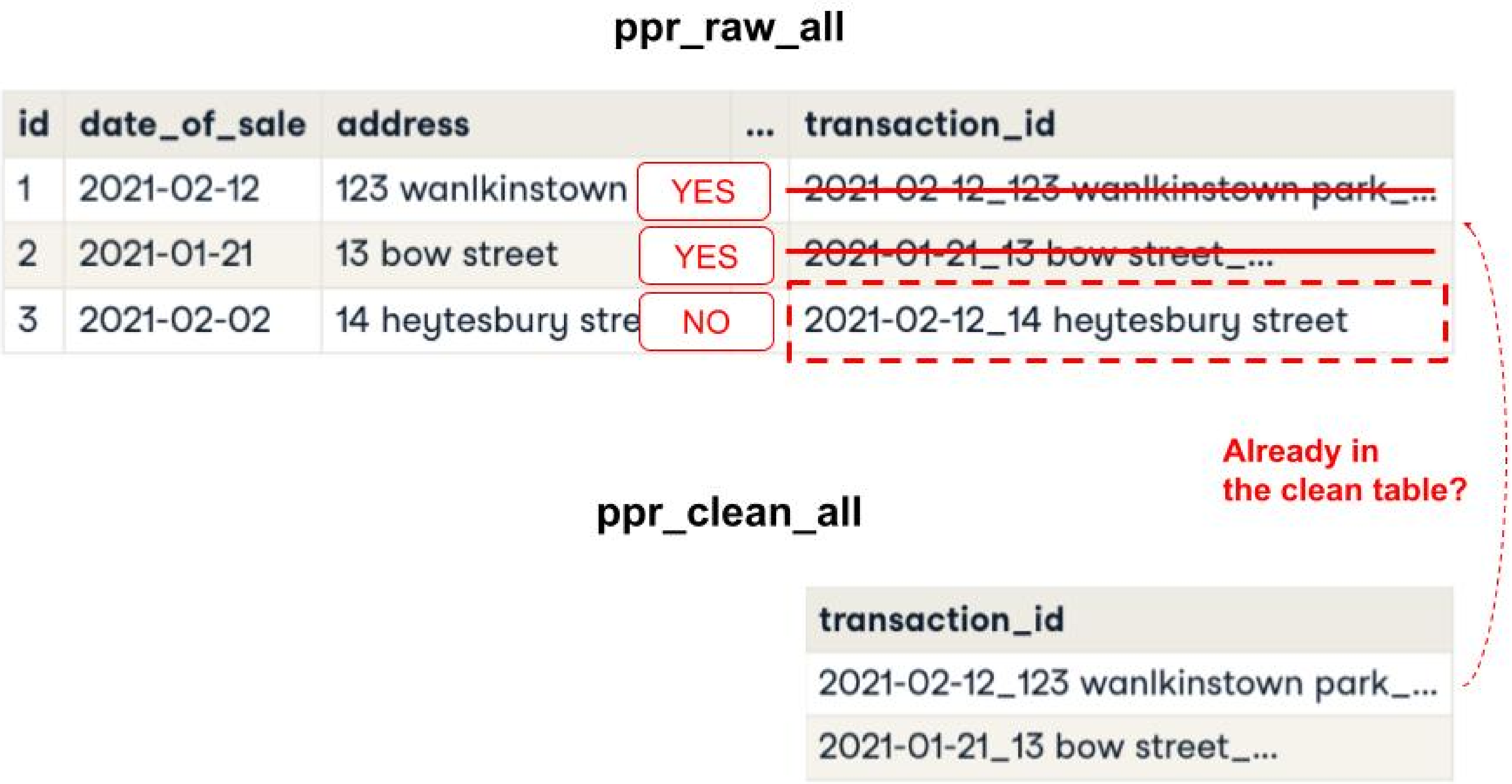
id	date_of_sale	address	...	transaction_id
1	2021-02-12	123 wanlkinstown	YES	~~2021-02-12_123 wanlkinstown park_...~~
2	2021-01-21	13 bow street	YES	~~2021-01-21_13 bow street_...~~
3	2021-02-02	14 heytesbury street	...	2021-02-12_14 heytesbury street

Already in the clean table?

ppr_clean_all

transaction_id
2021-02-12_123 wanlkinstown park_...
2021-01-21_13 bow street_...

Insert



ppr_raw_all

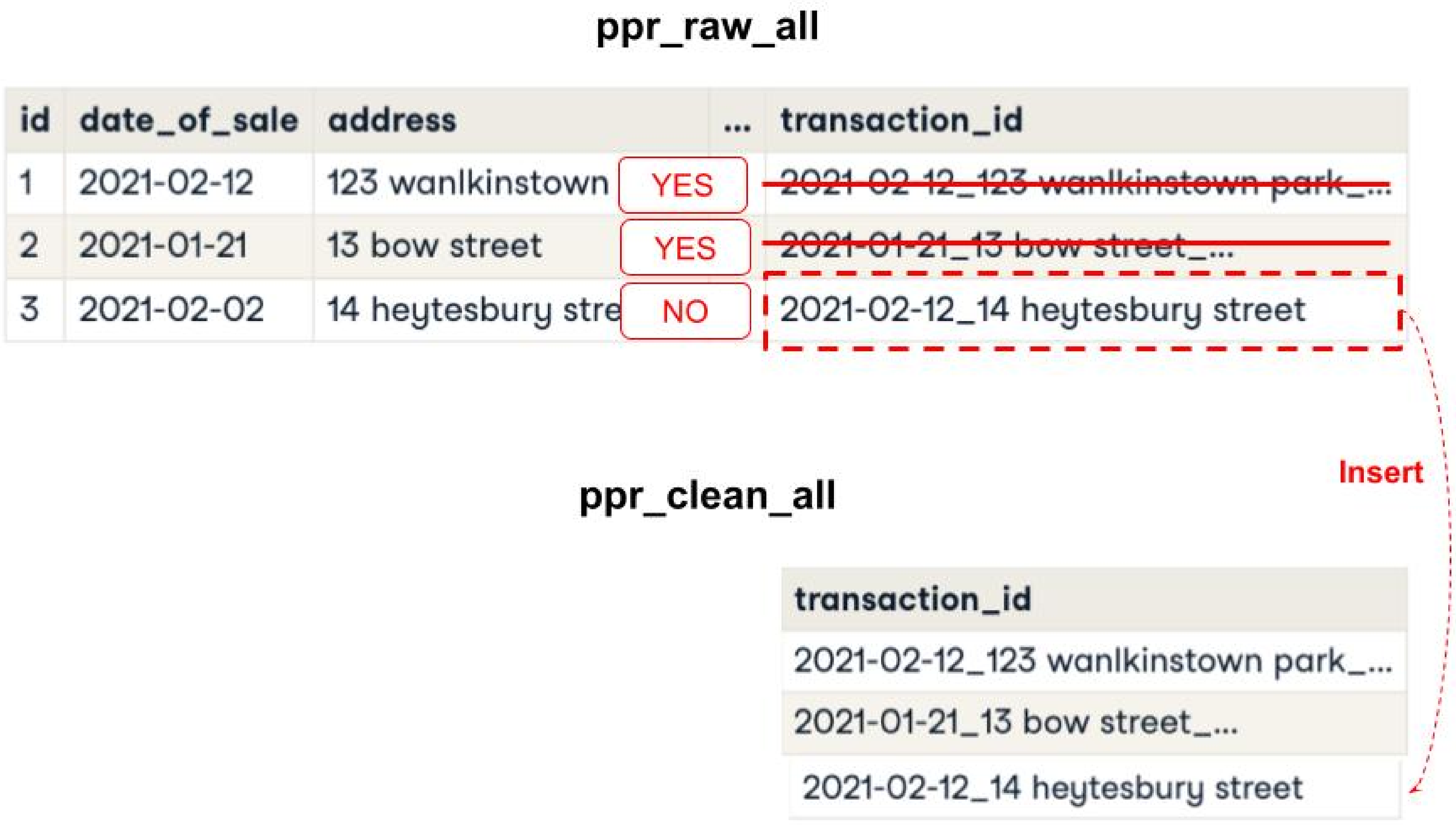
id	date_of_sale	address	...	transaction_id
1	2021-02-12	123 wanlkinstown	YES	~~2021-02-12_123 wanlkinstown park_...~~
2	2021-01-21	13 bow street	YES	~~2021-01-21_13 bow street_...~~
3	2021-02-02	14 heytesbury stre	NO	2021-02-12_14 heytesbury street

Already in the clean table?

ppr_clean_all

transaction_id
2021-02-12_123 wanlkinstown park_...
2021-01-21_13 bow street_...

Insert



id	date_of_sale	address	...	transaction_id
1	2021-02-12	123 wanlkinstown	YES	~~2021-02-12_123 wanlkinstown park_...~~
2	2021-01-21	13 bow street	YES	~~2021-01-21_13 bow street_...~~
3	2021-02-02	14 heytesbury stre	NO	2021-02-12_14 heytesbury street

transaction_id
2021-02-12_123 wanlkinstown park_...
2021-01-21_13 bow street_...
2021-02-12_14 heytesbury street

Insert

- To select rows to insert:
  - In SQL: `NOT IN`
  - In Python: `~` and `.in_()`
- `insert().from_select([<columns>], <ids>)`

Insert: an example

SQL

```sql
INSERT INTO ppr_clean_all
(SELECT date_of_sale, address, ...
FROM ppr_raw_all
WHERE transaction_id
NOT IN
    (SELECT transaction_id
     FROM ppr_clean_all)
)
```

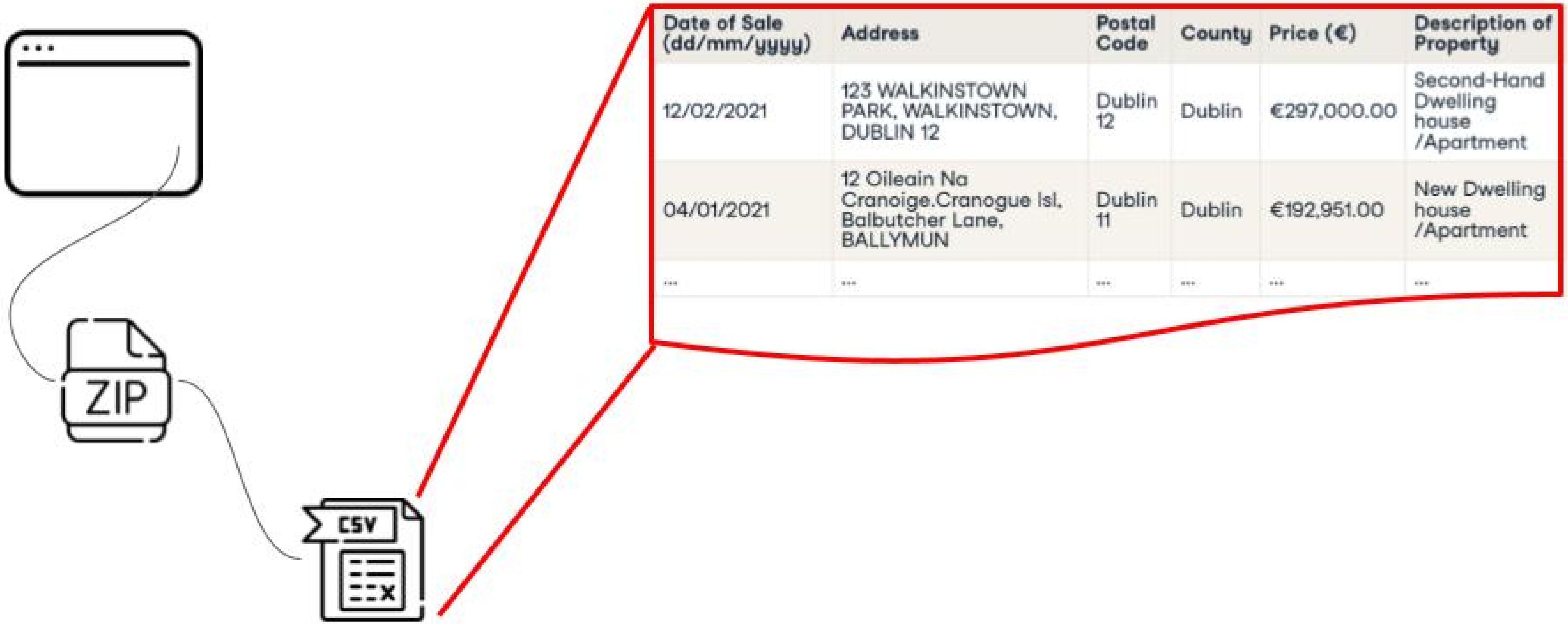
Insert: an example

Python

```
clean_transaction_ids = session.query(PprCleanAll.date_of_sale,
                                      PprCleanAll.address,
                                      ...)
transactions_to_insert = session.query(PprRawAll)
                                .filter(
                                ~PprRawAll.transaction_id.in_(clean_transaction_ids)
                                )
stm = insert(PprCleanAll).from_select(['date_of_sale', 'address', ...],
                                      transactions_to_insert)
```

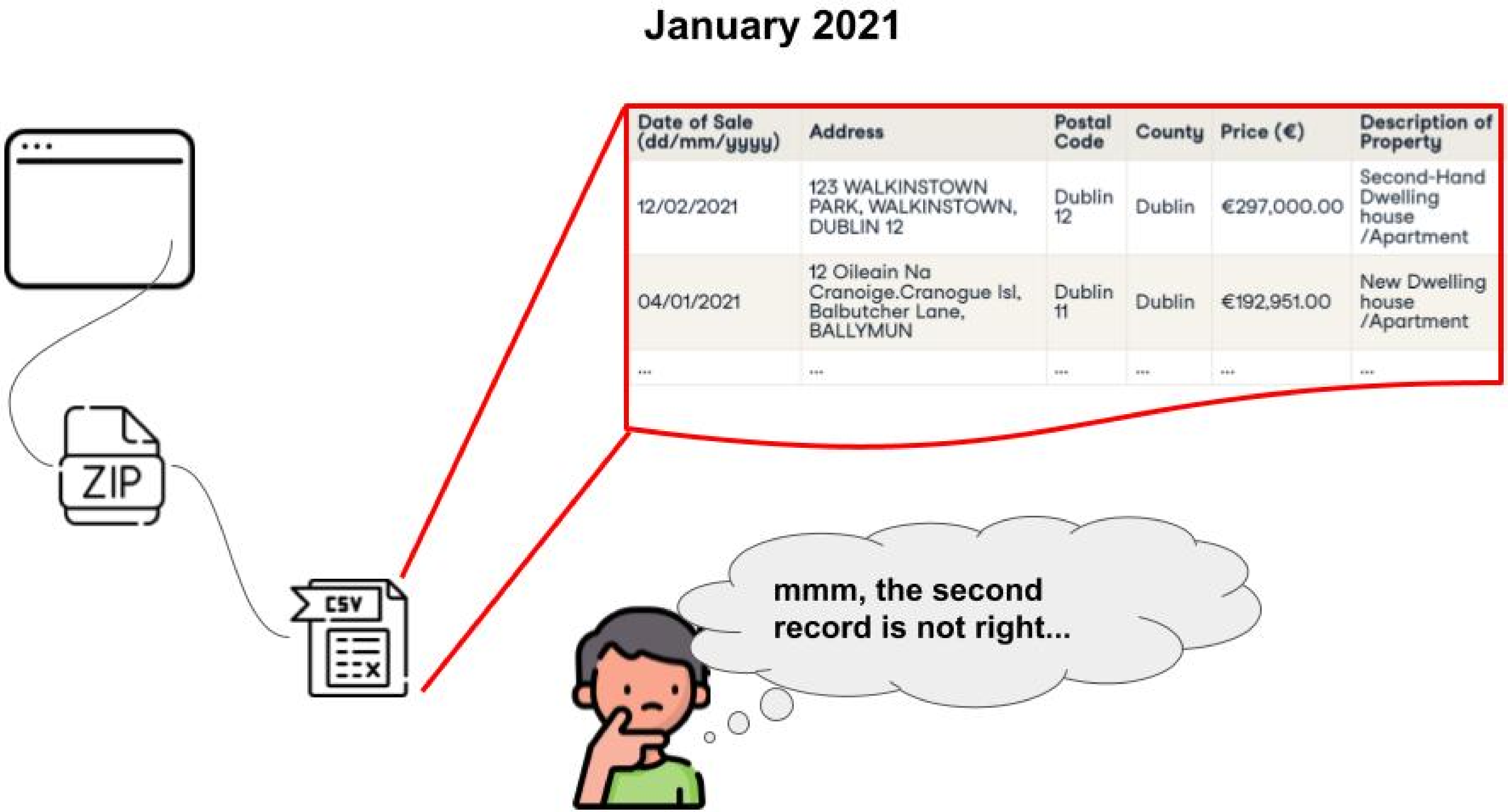
Delete

January 2021



Date of Sale (dd/mm/yyyy)	Address	Postal Code	County	Price (€)	Description of Property
12/02/2021	123 WALKINSTOWN PARK, WALKINSTOWN, DUBLIN 12	Dublin 12	Dublin	€297,000.00	Second-Hand Dwelling house /Apartment
04/01/2021	12 Oileain Na Cranoige.Cranogue Isl, Balbutcher Lane, BALLYMUN	Dublin 11	Dublin	€192,951.00	New Dwelling house /Apartment
...	...	...	...	...	...

Delete

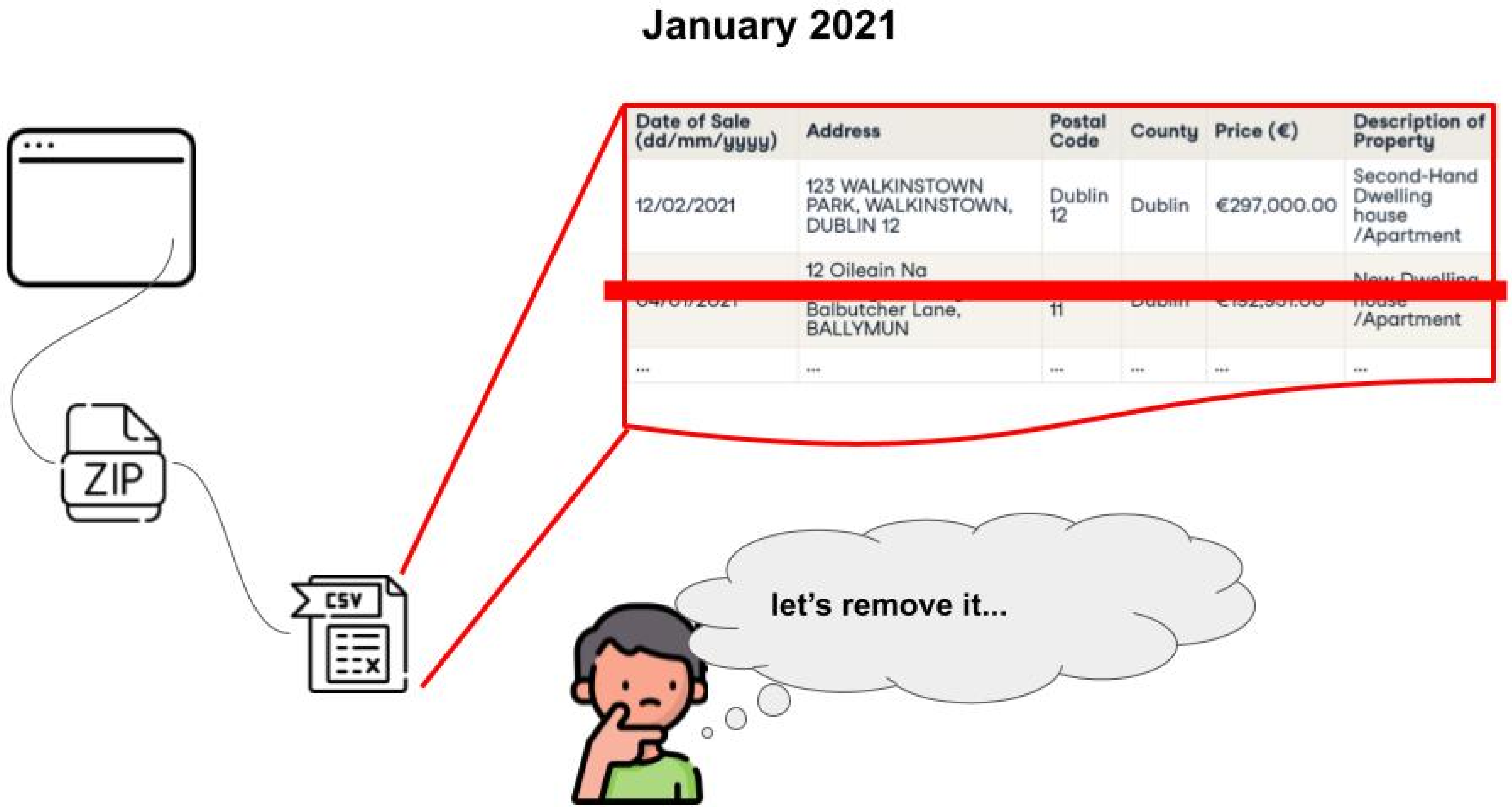


January 2021

Date of Sale (dd/mm/yyyy)	Address	Postal Code	County	Price (€)	Description of Property
12/02/2021	123 WALKINSTOWN PARK, WALKINSTOWN, DUBLIN 12	Dublin 12	Dublin	€297,000.00	Second-Hand Dwelling house /Apartment
04/01/2021	12 Oileain Na Cranoige.Cranogue Isl, Balbutcher Lane, BALLYMUN	Dublin 11	Dublin	€192,951.00	New Dwelling house /Apartment
...	...	...	...	...	...

mmm, the second record is not right...

Delete

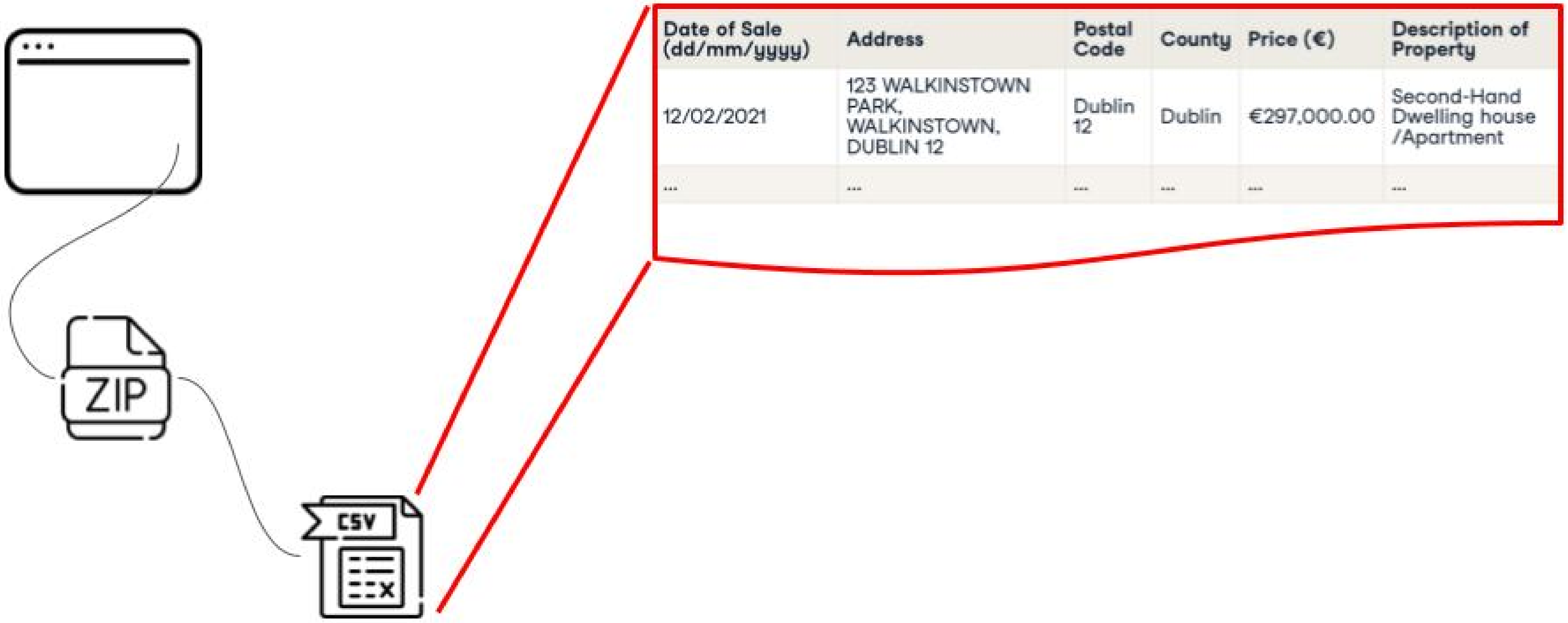


January 2021

Date of Sale (dd/mm/yyyy)	Address	Postal Code	County	Price (€)	Description of Property
12/02/2021	123 WALKINSTOWN PARK, WALKINSTOWN, DUBLIN 12	Dublin 12	Dublin	€297,000.00	Second-Hand Dwelling house /Apartment
~~04/01/2021~~	~~12 Oileain Na [illegible] Balbutcher Lane, BALLYMUN~~	~~[illegible] 11~~	~~Dublin~~	~~[illegible]~~	~~New Dwelling house /Apartment~~
...	...	...	...	...	...

Delete

February 2021



Date of Sale (dd/mm/yyyy)	Address	Postal Code	County	Price (€)	Description of Property
12/02/2021	123 WALKINSTOWN PARK, WALKINSTOWN, DUBLIN 12	Dublin 12	Dublin	€297,000.00	Second-Hand Dwelling house /Apartment
...	...	...	...	...	...

Delete

- `delete(PprCleanAll).filter()`
- SQL `NOT IN`

Delete: an example

SQL

```sql
DELETE FROM ppr_clean_all
WHERE transaction_id
NOT IN ("transaction_1", "transaction_2", "transaction_3")
```

Python

```python
raw_transaction_ids = session.query(PprRawAll.transaction_id)
session.query(PprCleanAll)
       .filter(~PprCleanAll.transaction_id.in_(raw_transaction_ids))
       .delete()
```

Let's practice!

ETL IN PYTHON