1. **Подробная структура команды INSERT:**

INSERT INTO ***table\_name*** [ AS ***alias*** ] [ ( ***column\_name*** [, ...] ) ]

[ OVERRIDING { SYSTEM | USER } VALUE ]

{ DEFAULT VALUES | VALUES ( { ***expression*** | DEFAULT } [, ...] ) [, ...] | ***query*** }

[ ON CONFLICT [ ***conflict\_target*** ] ***conflict\_action*** ]

[ RETURNING { \* | ***output\_expression*** [ [ AS ] ***output\_name*** ] } [, ...] ]

* 1. ***table\_name – имя таблицы для вставки***
  2. ***alias – псевдоним таблицы для дальнейшего обращения***
  3. ***column\_name – указанные колонки***
  4. ***OVERRIDING :***
     1. ***- OVERRIDING SYSTEM VALUE - Если это предложение указано, то любые значения, указанные для столбцов идентификаторов, будут переопределять значения, сгенерированные последовательностью по умолчанию.***
     2. ***- OVERRIDING USER VALUE - Если это предложение указано, то любые значения, указанные для столбцов идентификаторов, игнорируются и применяются значения, сгенерированные последовательностью по умолчанию***
     3. Пример копирования таблицы с **OVERRIDING** :

**INSERT INTO tbl2 OVERRIDING USER VALUE SELECT \* FROM tbl1**

* 1. **DEFAULT VALUES** - Все столбцы будут заполнены значениями по умолчанию
  2. **VALUES** – указывает значения для столбцов (в том же порядке что и об]явление столбцов)
  3. **ON CONFLICT** – определяет действие при конфликте уникальных значений
     1. ***conflict\_target – указывает на какое поле смотреть для выявления конфликта. Это поле обязательно должно быть под unique (уникальным) constraint пример :***

**ON** **CONFLICT** (id) – смотреть на конфликты для колонки id

* + 1. ***conflict\_action – действие которое нужно выполнить при обнаружении конфликта DO NOTHING или DO UPDATE, при DO UPDATE дальше идет инструкция как при команде UPDATE , при DO NOTHING ничего не делаем пример:***
       1. **DO** **UPDATE** **SET** "name"=EXCLUDED."name",
       2. **DO** **UPDATE** **SET** "name"=’new\_name’,

**1.7.2.1.1** EXCLUDED – это ссылка на новые вставляемые значения (чтобы не дублировать их). EXCLUDED."name" – это новое вставляемое значение для колонки “name”

* 1. **RETURNING** – без этой команды возвращается количество вставленных строк,

**С помощью команды RETURNING можно корректировать результирующий вывод**

**После вставки**

* + 1. **RETURNING \* - вернет вставленные строки с содержанием всех столбцов**
    2. **RETURNING “name”, “lastName” – вернет вставленные строки только с указанными колонками**

Пример конфликтующего INSERT :

**INSERT** **INTO** "user" **AS** u (id, "nameFirst", "nameLast")

**VALUES** (23, 'conflict\_first\_name', 'conflict\_last\_name')

**ON CONFLICT** (id)

**DO** **UPDATE SET** "nameFirst"=**EXCLUDED**."nameFirst", "nameLast"=**EXCLUDED**."nameLast"

**RETURNING** “id”, "nameFirst", "nameLast"

1. **Вставка INSERT**

INSERT INTO "role\_entity" ("id", "role") VALUES (1, 'manager')

1. "role\_entity" – имя таблицы
2. ("id", "role") – колонки (для postgress кавычки “” обозначают колонку)
3. (1, 'manager') – значения в эти колонки (кавычки ‘’ обозначают значение, если здесь поставить двойные кавычки “” то будет ошибка )