

## Задание

Написать проверку создания таблиц на языке Python с использованием любых вспомогательных библиотек.

В качестве СУБД можно выбрать любую доступную СУБД (Postgres, SQLite, MySQL и тп).

Таблицы достаточно создавать по схеме CREATE TABLE <tablename> (<columnname1> <columnname1>, .. , <columnnameN> <columnnameN>).

Дополнительные атрибуты (NULL, NOT NULL, PRIMARY KEY, FOREIGN KEY) необходимо опустить.

Цель:

Проверить всевозможные классы запросов, которые будут использовать прикладные разработчики.

Необходимо представить, что тестируемый механизм создания таблиц будет использоваться на ядерной станции и сбой недопустим.

Ожидаемый результат:

- 1/ Описание порядка проверки создания таблиц, результат представить в текстовом файле.
- 2/ Автоматизация порядка проверки, результат предоставить в виде файла на языке Python.

## Условия

1. Тестирование проводится на БД PostgreSQL
2. Имена полей и таблиц генерируются в соответствии с naming convention  
<https://www.geeksforgeeks.org/postgresql-naming-conventions/>
3. Типы полей взяты из [https://www.tutorialspoint.com/postgresql/postgresql\\_data\\_types.htm](https://www.tutorialspoint.com/postgresql/postgresql_data_types.htm)
4. Синтаксис команды <https://www.postgresql.org/docs/current/sql-createtable.html>
5. Макс количество полей в таблице 1600 <https://www.postgresql.org/docs/current/limits.html>
6. Таблицы создаются в текущей схеме с текущей кодировкой по умолчанию
7. Все параметры запроса, не указанные в задании не проверяются
8. Таблица считается успешно созданной, при ответном сообщении сервера Query returned successfully . При возврате ERROR таблица считается не созданной. Более подробный анализ кодов ошибок to be done
9. Зарезервированные слова, которые нельзя использовать как имена полей и таблиц  
<https://www.drupal.org/docs/develop/coding-standards/list-of-sql-reserved-words>

## Замечания

Стандартная проверка правильного создания таблицы — отсутствие ошибки pyodbc и успешное выполнение команды select.

При проверке разных типов полей выполняется только вышеуказанная проверка, созданный тип поля не проверяется (to be done)

Для отмеченных to be done проверок тесткейсы не реализованы

## **Тест - план**

(+ тест позитивный, - тест негативный)

### 1. Проверка синтаксиса create:

- +Простой правильный запрос create
- Пропустить table
- Пропустить скобку, использовать квадратную скобку, фигурную скобку
- Пустые скобки вместо колонок

### 2 Проверка имени таблицы по nameconvention

- + Максимальная длина 64, >64 (длина обрезается до 64), мин длина 1
- Длина 0
- Число в начале имени
- + Число в середине имени
- Спец символы
- + Символы другой кодировки /to be done
- + Все запрещенные символы при использовании ""

### 3. Проверка полей

Проверка имен полей по nameconvention (см. П1)

- + Количество полей: от 0 до макс. Знач. 1600
- Количество полей >1600
- +Типы полей: все возможные типы по списку
- Несуществующий тип

Проверка полей с переменной длиной - текстовые, даты / to be done

Проверка массивов типов полей /to be done

проверка мах размера поля 1 ГБ /to be done

### 4. Проверка создания таблиц с указанием схемы /to be done