

| | |
|-----------|--------------------------------|
| 2023/2024 | TCXX - TP n°XX |
| BTS SIO | Auteur : Loric Worms |
| 1SIOB | Date de rédaction : 12/01/2024 |

Compte-rendu de TP

Étape 11 : Evolution 1 - Suppression des caractères spéciaux dans les noms d'utilisateur

Reprendre les fonctionnalités du programme genererScriptSqlCreate.py et mettre au point une nouvelle version de celui-ci, nommée genererScriptSqlCreateV2.py permettant de respecter les règles imposées par l'administrateur du système :

- prénoms composés : on juxtapose les initiales de chaque prénom
- le nom d'utilisateur ne contient que des caractères alphabétiques en majuscules ('A' à 'Z') ; tous les autres caractères seront éliminés.

Exemple :
Jean-Baptiste LE FLOC'H doit donner jblefloch
Xavier Bertrand DE LA VILLETANAY doit donner xbdelavilletanay

jeu d'essai :

| Prénom | / Nom | / compte mariadb | / nom bd |
|--------------|------------|------------------|--------------|
| Jean-Bernard | DUPONT | jbdupont | bdjpdupont |
| Patrice | DU CHÂTEAU | pduchateau | bdpdechateau |

```
# Fonction pour créer le script SQL
def generer_script_sql(nom_fichier_csv, nom_fichier_mdp, nom_fichier_sql):
    with open(nom_fichier_csv, 'r') as csv_file, open(nom_fichier_mdp, 'r') as mdp_file, open(nom_fichier_sql, 'w') as sql_file:
        lignes_csv = csv_file.read().splitlines()[4:]
        lignes_mdp = mdp_file.read().splitlines()[1:]

        # Boucle à travers chaque ligne des fichiers CSV
        for ligne, mdp_ligne in zip(lignes_csv, lignes_mdp):
            colonnes = ligne.split(';')
            nom_complet = colonnes[5].split()
            prenom = nom_complet[1]
            nom = nom_complet[0]

            compte_mariadb = generer_compte_mariadb(prenom, nom)
            nom_bdd = generer_nom_bdd(prenom, nom)
            mot_de_passe = mdp_ligne

            # Instructions SQL pour créer l'utilisateur et la base de données
            sql_file.write(f"CREATE USER '{compte_mariadb}'@'localhost' IDENTIFIED BY '{mot_de_passe}';\n")
            sql_file.write(f"CREATE DATABASE {nom_bdd};\n")
            sql_file.write(f"GRANT {generer_privileges()} ON {nom_bdd}.* TO '{compte_mariadb}'@'localhost';\n")
            sql_file.write("\n")
```

fonction principale

```
# Fonction pour nettoyer le nom d'utilisateur
def nettoyer_nom(nom):
    return re.sub(r'[^A-Z]', '', nom).lower()
```

fonction pour ne garder que les majuscules dans le nom de l'utilisateur.

| | |
|--------------------------------|--------------------------------|
| 2023/2024 | TCXX - TP n°XX |
| BTS SIO 1SIOB | Auteur : Loric Worms |
| | Date de rédaction : 12/01/2024 |

```
CREATE USER 'avergne'@'localhost' IDENTIFIED BY 'rN4zd96HeN';
CREATE DATABASE dbavergne;
GRANT CREATE, ALTER, DROP, INSERT, UPDATE, DELETE, SELECT ON dbavergne.* TO 'avergne'@'localhost';
```

Extrait du fichier creerUsersBddAccesV2.sql

A remettre :

- Un compte-rendu contenant l'énoncé de l'étape, un jeu d'essai et les captures d'écran du contenu des fichiers CSV, python et sql
- le programme python commenté genererScriptSqlCreateV2.py
- le script obtenu creerUsersBddAcces.sql