

structure du code python de tvideos.py

but du projet

Ce projet a pour but de générer des données aléatoires pour simuler des informations sur des vidéos YouTube, des chaînes YouTube et des commentaires.

Bibliothèques utilisées

- `random` : utilisée pour générer des données aléatoires telles que des likes, des dislikes, des vues, des abonnés, des ID de vidéos et des timestamp.
- `sqlite3` : utilisée pour gérer des bases de données SQLite.
- `Faker` : utilisée pour générer des données aléatoires en français, comme des titres de vidéos, des descriptions, des noms d'utilisateurs, des bio et des emails.
- `BaseProvider` : utilisée pour créer des fournisseurs de données aléatoires personnalisés pour `Faker`.

Classes définies

Classe `Key`

Cette classe est utilisée pour générer des mots-clés vidéo et des ID de vidéo YouTube aléatoires en utilisant les méthodes `video_keyword()` et `id()` respectivement.

Classe `Video`

Cette classe utilise les méthodes de la classe `Key` pour générer des informations aléatoires sur les vidéos telles que les titres, les descriptions, les likes, les dislikes, les vues, les ID et les timestamp.

Classe `Chaine`

Cette classe utilise les méthodes de `Faker` pour générer des informations sur les chaînes telles que les pseudos, les abonnés, la bio, le timestamp et les emails.

Classe `Commentaire`

Cette classe utilise les méthodes de `Faker` pour générer des informations sur les commentaires telles que les contenus et les timestamp.

Utilisation de la classe `Video`

Méthode `__init__()`

Cette méthode est appelée lorsque la classe est instanciée et initialise les attributs suivants :

- `title` : un titre aléatoire généré à l'aide de la méthode `random_title()`
- `description` : une description aléatoire générée à l'aide de la méthode `random_description()`
- `likes` : un nombre aléatoire de likes généré à l'aide de la méthode `random_like()`
- `dislikes` : un nombre aléatoire de dislikes généré à l'aide de la méthode `random_dislike()`

- `views` : un nombre aléatoire de vues généré à l'aide de la méthode `random_view()`
- `id` : un ID aléatoire de vidéo généré à l'aide de la méthode `video_id()`
- `timestamp` : un timestamp aléatoire généré à l'aide de la méthode `random_timestamp()`

Méthode `random_title()`

Cette méthode utilise la méthode `video_keyword()` de la classe `Key` pour générer un titre aléatoire pour la vidéo.

Méthode `random_description()`

Cette méthode utilise les méthodes de `Faker` pour générer une description aléatoire pour la vidéo.

Méthode `random_like()`, `random_dislike()`, `random_view()`

Ces méthodes utilisent `random.randint()` pour générer des valeurs aléatoires pour les likes, les dislikes et les vues de la vidéo.

Méthode `video_id()`

Cette méthode utilise la méthode `id()` de la classe `Key` pour générer un ID aléatoire de vidéo YouTube.

Méthode `random_timestamp()`

Cette méthode utilise `random.randrange()` pour générer un timestamp aléatoire entre le 1er janvier 2005 et le 1er janvier 2078.

Utilisation de la classe `Chaine`

Méthode `__init__()`

Cette méthode est appelée lorsque la classe est instanciée et initialise les attributs suivants :

- `pseudo` : un pseudo aléatoire généré à l'aide de la méthode `random_pseudo()`
- `abonnes` : un nombre aléatoire d'abonnés généré à l'aide de `random.randint()`
- `bio` : une bio aléatoire générée à l'aide de la méthode `random_bio()`
- `timestamp` : un timestamp aléatoire généré à l'aide de `random.randrange()`
- `email` : un email aléatoire généré à l'aide de la méthode `random_email()`

Méthode `random_pseudo()`

Cette méthode utilise la méthode `user_name()` de `Faker` pour générer un pseudo aléatoire pour la chaîne.

Méthode `random_bio()`

Cette méthode utilise la méthode `text()` de `Faker` pour générer une bio aléatoire pour la chaîne.

Méthode `random_email()`

Cette méthode utilise la méthode `email()` de `Faker` pour générer un email aléatoire pour la chaîne.

Utilisation de la classe `Commentaire`

Méthode `__init__()`

Cette méthode est appelée lorsque la classe est instanciée et initialise les attributs suivants :

- `contenu` : un contenu aléatoire généré à l'aide de la méthode `random_contenu()`
- `timestamp` : un timestamp aléatoire généré à l'aide de `random.randrange()`

Méthode `random_contenu()`

Cette méthode utilise les méthodes de `Faker` pour générer un contenu aléatoire pour le commentaire.

Auteur

Ce code a été écrit par Dylan.

Références

Le code source de ce projet est disponible sur GitHub à l'adresse suivante : [tvideos](#)

Temps de développement

Le temps de développement de ce projet est disponible sur Wakatime :  wakatime 17 hrs 29 mins