**Generate reports**

Тепер ми потренуємось створити сценарій під назвою **ticky\_check.py** , який генерує два різні звіти з цього внутрішнього файлу журналу системи продажу квитків, тобто syslog.log . Цей сценарій створить такі звіти:

* **Рейтинг помилок, створених системою** : список усіх зареєстрованих повідомлень про помилки та кількість виявлень кожної помилки, відсортований від найпоширенішої до найменш поширеної. Цей звіт не враховує залучених користувачів.
* **Статистика використання користувачами служби** : список усіх користувачів, які користувалися системою, включаючи кількість інформаційних повідомлень і кількість повідомлень про помилки, які вони створили. Цей звіт відсортовано за іменем користувача.

Щоб створити ці звіти, напишіть скрипт на Python під назвою ticky\_check.py . Використовуйте для цього редактор nano .

**nano ticky\_check.py**

Скопійовано!

content\_copy

Додайте лінію шебанг.

**#!/usr/bin/env python3**

Скопійовано!

content\_copy

Ось ваше завдання: напишіть сценарій для створення двох різних звітів на основі рейтингу помилок, створених системою, і статистики використання користувачами служби. Ви напишете сценарій самостійно, але ми допоможемо вам у всьому.

Спочатку імпортуйте всі модулі Python, які ви використовуватимете в цьому сценарії Python. Після імпорту потрібних модулів ініціалізуйте два словники: один для кількості різних повідомлень про помилки, а інший для підрахунку кількості записів для кожного користувача (розподіл між INFO та ERROR).

Тепер проаналізуйте кожен запис журналу у файлі syslog.log, перебираючи файл.

Для кожного запису журналу вам доведеться спочатку перевірити, чи відповідає він формату повідомлення **INFO** або **ERROR** . Для цього слід використовувати регулярні вирази. Коли ви знайдете успішний збіг, додайте одиницю до відповідного значення в словнику per\_user . Якщо ви отримуєте повідомлення **ERROR** , додайте його до відповідного запису в словнику помилок, використовуючи правильну структуру даних.

Після того, як ви обробили записи журналу з файлу syslog.log, вам потрібно відсортувати як per\_user і error dictionary перед створенням файлів звіту CSV.

Майте на увазі, що:

* error dictionary має бути відсортований за кількістю помилок від найбільш поширених до найменш поширених.
* user dictionary має бути відсортований за іменем користувача.

Вставте назви стовпців як  ("Error", "Count") у позиції нульового індексу відсортованого словника error . І вставте назви стовпців як ("Username", "INFO", "ERROR") у позиції нульового індексу відсортованого словника per\_user .

Після сортування цих словників збережіть їх у двох різних файлах: **error\_message.csv** і **user\_statistics.csv** .

Збережіть файл **ticky\_check.py** , натиснувши Ctrl-o, клавішу Enter і Ctrl-x.

**Visualize reports**

По-перше, надайте дозвіл на виконання сценарію Python ticky\_check.py.

**chmod +x ticky\_check.py**

Скопійовано!

content\_copy

Запустіть ticky\_check.py за допомогою такої команди:

**./ticky\_check.py**

Скопійовано!

content\_copy

Виконання ticky\_check.py створить два файли звіту \_\_error\_message.csv \_\_і **user\_statistics.csv** .

Тепер ви можете візуалізувати \_\_error\_message.csv \_\_і **user\_statistics.csv** , перетворивши їх на сторінки HTML. Для цього передайте файли один за одним у файл сценарію **csv\_to \_html.py** , як ми робили в попередньому розділі.

Щоб перетворити **error\_message.csv** у файл HTML, виконайте таку команду:

**./csv\_to\_html.py error\_message.csv /var/www/html/<html-filename>.html**

Скопійовано!

content\_copy

Замініть <html-filename> на назву за вашим вибором.

Щоб перетворити **user\_statistics.csv** у файл HTML, виконайте таку команду:

**./csv\_to\_html.py user\_statistics.csv /var/www/html/<html-filename>.html**

Скопійовано!

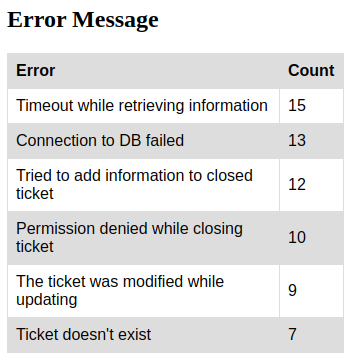
content\_copy

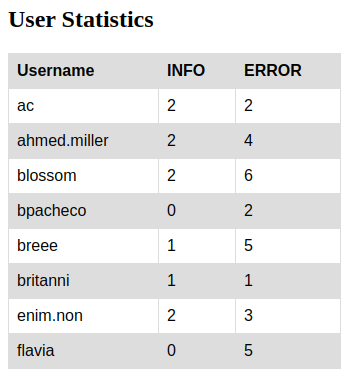
Замініть <html-filename> новою назвою

Тепер, щоб переглянути ці HTML-сторінки, відкрийте будь-який веб-браузер і введіть наступну URL-адресу в рядок пошуку.

**[linux-instance-external-IP]**/**[html-filename].html**

Вихід:





Натисніть *Перевірити мій прогрес* , щоб перевірити мету.