‹ Return to "Full Stack Web Developer Nanodegree" in the classroom

Logs Analysis

REVIEW | CODE REVIEW 1 | HISTORY

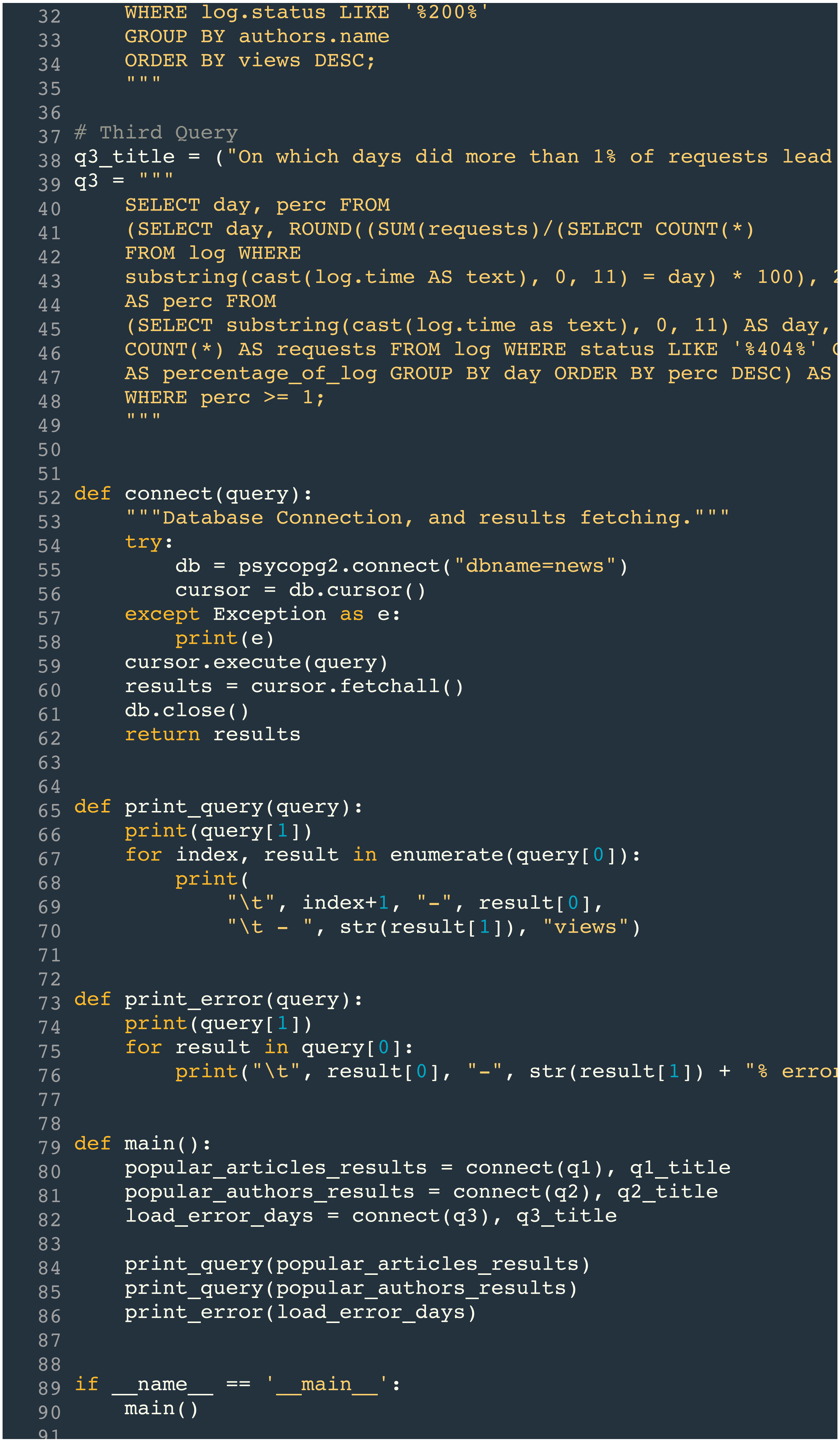
▾ reporting.py 1

```
#!/usr/bin/env python3
"""
    The project summarizes and reports three questions from
       a large database of a News website/application.
"""

# Imported libralies
import psycopg2

# Here are the questions the reporting tool should answer.
# First Query
q1_title = ("What are the most popular three articles of all
q1 = """
    SELECT articles.title, COUNT(*) AS viewNumbmer
    FROM articles INNER JOIN log on log.path
    like concat('%', articles.slug, '%')
    where log.status like '%200%'
    GROUP BY articles.title, log.path
    ORDER BY viewNumbmer DESC
    LIMIT 3;
    """


# Second Query
q2_title = ("Who are the most popular article authors of all
q2 = """
    SELECT authors.name, COUNT(*) AS views
    FROM articles
    INNER JOIN authors ON articles.author = authors.id
    INNER JOIN log ON log.path
    LIKE concat('%', articles.slug, '%')
```



```
    WHERE log.status LIKE '%200%'
    GROUP BY authors.name
    ORDER BY views DESC;
    """

# Third Query
q3_title = ("On which days did more than 1% of requests lead
q3 = """
    SELECT day, perc FROM
    (SELECT day, ROUND((SUM(requests)/(SELECT COUNT(*)
    FROM log WHERE
    substring(cast(log.time AS text), 0, 11) = day) * 100), 2
    AS perc FROM
    (SELECT substring(cast(log.time as text), 0, 11) AS day,
    COUNT(*) AS requests FROM log WHERE status LIKE '%404%' G
    AS percentage_of_log GROUP BY day ORDER BY perc DESC) AS
    WHERE perc >= 1;
    """


def connect(query):
    """Database Connection, and results fetching."""
    try:
        db = psycopg2.connect("dbname=news")
        cursor = db.cursor()
    except Exception as e:
        print(e)
    cursor.execute(query)
    results = cursor.fetchall()
    db.close()
    return results


def print_query(query):
    print(query[1])
    for index, result in enumerate(query[0]):
        print(
            "\t", index+1, "-", result[0],
            "\t - ", str(result[1]), "views")


def print_error(query):
    print(query[1])
    for result in query[0]:
        print("\t", result[0], "-", str(result[1]) + "% error


def main():
    popular_articles_results = connect(q1), q1_title
    popular_authors_results = connect(q2), q2_title
    load_error_days = connect(q3), q3_title

    print_query(popular_articles_results)
    print_query(popular_authors_results)
    print_error(load_error_days)


if __name__ == '__main__':
    main()
```

AWESOME

Awesome job fixing the typos pointed in the previous review!

I'm proud of you!

This project is great!

RETURN TO PATH

Rate this review

☆☆☆☆☆