截止日期：2024 年 11 月 9 日
Max Marks: 100

产品说明:
•所有代码都应该有良好的注释，并且易于阅读。
•允许使用外部库，如 pandas， numpy, matplotlib，
sqlalchemy、pymongo、Flask、Streamlit 和 kafka-python。
•提交你的解决方案，提供一个链接到 GitHub 存储库
包含:
-任务的 Python 脚本（Jupyter notebook 或.py 文件）。
- SQL 和 MongoDB 查询的。txt 文件。
—CSV 文件、图形和任务所需的任何其他输出。
-自述文件。Md 文件，说明如何在本地运行项目。
-托管链接到您的 Streamlit web 应用程序。

问题：全栈数据科学应用程序
使用全球天气数据集
你需要创建一个涉及数据的全栈数据科学项目
处理，数据分析，基于 kafka 的数据流，数据库操作，
以及使用 Streamlit 托管 web 应用程序。对于这个评估，你将
使用 Kaggle 的**世界天气数据集**，该数据集可以下载
摘自以下链接：
URL:
https://www.kaggle.com/datasets/nelgiriyewithana/global-weather-repository

循序渐进的任务:
1.**数据集探索和论证（5 分）：**
•从 Kaggle 下载数据集。
•提供一个简短的理由（3-4 行），说明为什么这个数据集适合于全球范围内的天气相关分析和预测。

2. **数据处理（20 分）：**
•将数据集加载到 pandas DataFrame 中。
•清理数据集(例如，处理缺失的值，将列转换为
适当的数据类型)。
•将清理后的数据存储在 CSV 文件中，并带有适当的列
名称和格式。
•显示数据集中关键统计数据的摘要(例如，平均值，
最大和最小温度，湿度水平等)。

3. **数据分析与可视化（20 分）：**
•使用 pandas 和 numpy，执行以下分析：
-基于全球天气数据集生成摘要
（例如，全球最热和最冷的 5 个地点）。

-将数据按相关字段（如地区、年份）和进行分组
计算平均、最高或最低温度；
湿度或降水量。
-绘制至少两种不同的可视化图（例如，的直方图）
温度，显示温度或温度变化的线形图
特定地区随时间的降水量)。

4.流数据的 Kafka 生产者和消费者（20 分）：
•设置一个名为全球天气的 Kafka 主题。
编写一个模拟实时天气的 Kafka 生产者脚本
基于你的数据集更新全球位置。每秒钟，发送一个更新的数据点（例如，温度，湿度，降水）到 Kafka 主题。
编写一个 Kafka Consumer 脚本，监听全球天气主题，并将更新的数据记录到 CSV 文件中。
•显示运行 Producer 60 秒后消费的更新摘要。

5.数据库操作（20 分）：
•创建一个名为 GlobalWeatherDB 的 MySQL 数据库并导入
清理后的数据集。
•编写 SQL 查询到：
-检索气温最高或降水最低的前 5 个地点。
-检索特定日期或条件的所有记录（例如，温度 35°C，降水 100 毫米）。
-按操作执行组（例如，平均温度）
国家、地区的总降水量)。
•创建一个 MongoDB 数据库，并将相同的数据集导入到名为 GlobalWeather data 的集合中。
•使用 MongoDB 查询，检索：
-匹配特定条件的所有记录(例如，数据来自 a
特定的月份或大洲)。
-某项的最高值或最低值的前 3 条记录
特定指标（例如，最热的位置，最高的天数）
降水)。

6. Streamlit Web 应用程序（20 分）：
•使用 Flask 构建一个 REST API，通过 Kafka 提供清洁数据和流更新的全球天气数据。
•使用 Streamlit 构建一个与 Flask API 交互的 web 应用程序。
web 应用程序应该有两个部分：
-**数据仪表板：** 显示清洁的全球天气数据集与过滤器（例如，按日期，地区，或其他相关指标）。
-**实时数据更新：** 显示实时天气更新正在通过 Kafka 流，与数据点的实时绘图。
•在免费的 Streamlit 云平台上部署 Streamlit 应用程序
在 GitHub README.md 中包含到托管应用程序的链接
文件。

注意：将链接提交到包含代码、查询、CSV 输出和 README 的 GitHub 存储库。md 文件。README 应该包括如何在本地运行项目和托管 Streamlit 应用程序链接的说明