ASSESSMENT PROBLEM

You are a Data Scientist at **"Sensorlytics,"** a company that deploys environmental monitoring sensors across different cities. Each city has a set of sensor stations that collect data on air quality and weather.

You have received a single compressed file, `sensor_data_2025.zip`, from the data engineering team. This file contains the complete data dump for the first quarter of 2025, organized by city. However, the data is nested within multiple layers of zip archives.

Your task is to create an automated Python script to process this nested data structure, perform a series of analyses, and generate a final report for the operations team.

Data Structure

The top-level file you will be given is **`sensor_data_2025.zip`**. Its internal structure is as follows:

```
sensor_data_2025.zip
|
├── munich_data.zip
|   ├── readings.csv
|   └── station_info.json
|
├── tokyo_data.zip
|   ├── readings.csv
|   └── station_info.json
|
└── sydney_data.zip
    ├── readings.csv
    └── station_info.json
```

File Content Details:

1. **`station_info.json`**: Contains metadata for the sensor stations in a given city.
   - `station_id`: Unique identifier for the sensor station (integer).
   - `location`: A dictionary with `latitude` and `longitude` (floats).
   - `status`: The operational status of the station (e.g., 'active', 'maintenance').

    - `deployment_year`: The year the station was deployed (integer).
2. **`readings.csv`**: Contains the raw sensor readings from all stations in that city.
    - **Columns**: `station_id`, `timestamp`, `temperature_celsius`, `humidity_percent`, `pm2_5` (fine particulate matter).

Required Tasks:

Your script must perform the following tasks in sequence.

Part 1: Get All the Data into Python (40 Marks)

1. Open the main file, **`sensor_data_2025.zip`**.
2. Go through each of the city zip files inside (e.g., `munich_data.zip`, `tokyo_data.zip`).
3. For each city, read the **`readings.csv`** file and the **`station_info.json`** file into two separate Pandas DataFrames.
4. As you read the data for each city, add a new column named `city` to both DataFrames. The value should be the city's name (e.g., 'munich', 'tokyo').
5. Combine the data from all cities into two final DataFrames: one holding all sensor readings, and one holding all station information.

Part 2: Clean and Analyze the Data (40 Marks)

1. Merge your two DataFrames (readings and station info) into one single DataFrame using the `station_id` column.
2. Clean the data:
    - Keep only the rows where the `status` column is **'active'**.
    - Remove any rows that have missing data in the `temperature_celsius` or `pm2_5` columns.
3. Create a final summary. For each **city**, calculate:
    - The average temperature (`temperature_celsius`).
    - The average air pollution (`pm2_5`).

Part 3: Save Your Final Report (20 Marks)

1. Take the summary you created in Part 2.
2. Save this summary as a single CSV file named **`city_summary_report.csv`**. The file should have three columns: `city`, `average_temperature`, and `average_pm2_5`.