📊 Automating QuickSight Dataset Refresh with Python, S3, and Task Scheduler

This guide takes you from raw daily CSV to fully automated, up-to-date QuickSight dashboards—with no manual intervention after setup.

🎯 Business Use Case

You run a daily sales reporting dashboard for leadership. Each morning:

1. **5:45 AM: A Python script cleans and uploads the latest sales CSV to S3.**
2. **5:47 AM: The script updates manifest.json in S3 to point to the new file.**
3. **6:00 AM: QuickSight automatically refreshes the dataset via the manifest URL.**
4. **6:05 AM: Dashboards show the night's sales—ready before business hours.**

📁 Folder Structure (Local)

C:\Users\Win10\Desktop\schedule_refresh

│

├── 2025-04-30.csv # Raw daily data files, autopopulated or dropped in

├── manifest.json # Template/updated by script

├── refresh_upload.py # Main Python script

└── README.md # This guide

🔑 Prerequisites

1. **AWS Credentials:**
    - **Create an IAM user with s3:PutObject, s3:GetObject, s3:ListBucket, and QuickSight permissions.**
    - **Store them as environment variables:**
    - **setx Amazon.ACCESS_KEY "YOUR_ACCESS_KEY"**
    - **setx Amazon.SECRET_KEY "YOUR_SECRET_KEY"**
2. **Python 3.8+ with packages:**
3. **pip install pandas boto3**
4. **AWS S3 Bucket in eu-north-1 (e.g., quicksight-data-nanda).**
5. **Amazon QuickSight subscription (Standard or Enterprise).**

---

1. Setup S3 Bucket

- **Create bucket quicksight-data-nanda in eu-north-1.**
- **Grant your IAM user PutObject, GetObject, ListBucket.**
- **(Optional) Create a subfolder if desired, but root is simplest.**

---

2. manifest.json Explained

QuickSight uses a manifest.json file to know which CSV to load.

It must live in S3 at: s3://quicksight-data-nanda/manifest.json

Structure

```
{
  "fileLocations": [
    { "URIs": [ "s3://quicksight-data-nanda/CLEANED_2025-04-30.csv" ] }
  ],
  "globalUploadSettings": {
    "format": "CSV",
    "delimiter": ",",
    "textqualifier": "\"",
    "containsHeader": "true"
  }
}
```

- **fileLocations: array; each entry has URIs.**
- **URIs: exactly one s3:// link to your cleaned file. Do not include the HTTPS URL here—QuickSight treats each URI as a separate source and will double-ingest if you include both.**
- **globalUploadSettings:**
  - **format: file format (CSV).**
  - **delimiter: field separator (,).**
  - **textqualifier: quoting character (").**
  - **containsHeader: must be "true" if first row is header.**

QuickSight Connection URL

When you create your dataset, QuickSight needs the HTTPS URL:

https://quicksight-data-nanda.s3.eu-north-1.amazonaws.com/manifest.json

QuickSight will fetch this JSON to know which CLEANED_*.csv file to load.

---

3. Python Script: refresh_upload.py

Automates:

1. **Find latest YYYY-MM-DD.csv in local folder.**
2. **Clean OrderDate into YYYY-MM-DD text.**
3. **Upload cleaned CSV as CLEANED_YYYY-MM-DD.csv to S3.**
4. **Update manifest.json to point at the new S3 URI.**
5. **Overwrite the manifest in S3.**

 # See user's full script above; it handles env vars, cleaning, upload, manifest.

1. **Place your raw CSVs in the folder.**
2. **Run:**
3. **python refresh_upload.py**
4. **Confirm logs: original rows, cleaned rows, S3 upload, manifest update.**

---

4. Create QuickSight Dataset via Manifest

1. **In QuickSight, go to Datasets → New dataset → S3.**
2. **Choose Manifest file.**
3. **Paste:**
4. **https://quicksight-data-nanda.s3.eu-north-1.amazonaws.com/manifest.json**
5. **Name it (e.g., DailySalesData), import into SPICE for better performance.**
6. **In the Fields panel, find OrderDate, click ⋮ → Change data type → String (prevents T00:00:00Z).**
7. **Save dataset.**

---

5. Automate with Windows Task Scheduler (5:45 AM)

1. **Open Task Scheduler.**
2. **Create Basic Task:**
    - **Name: DailySalesRefresh.**
3. **Trigger: Daily at 5:45 AM.**
4. **Action: Start a program:**
    - **Program: python**
    - **Arguments: "C:\Users\Win10\Desktop\schedule_refresh\refresh_upload.py"**

5. **Finish.**

Now at 5:45 AM, your script runs automatically, updating S3 and manifest.

---

6. Schedule QuickSight Refresh (6:00 AM)

1. **In QuickSight, navigate to your dataset DailySalesData.**
2. **Click ... → Schedule refresh.**
3. **Add a daily refresh at 6:00 AM (your local timezone).**
4. **Save.**

Timeline: | 5:45 AM | Python script uploads fresh data and manifest to S3 | | 6:00 AM | QuickSight auto-refreshes dataset via manifest.json |

---

7. Build and Share Dashboards

- **Use DailySalesData fields to create visuals (charts, tables, KPIs).**
- **Dashboards now auto-update daily without manual steps.**

✅ **Conclusion**

You've configured:

- **Environment variables for AWS credentials**
- **Automated Python upload & manifest updates**
- **S3 storage of cleaned CSV and manifest**
- **QuickSight dataset via manifest URL**
- **Windows & QuickSight scheduling at 5:45 AM and 6:00 AM**

Enjoy your automated BI pipeline! 🎉