



# Mode d'emploi : Dashboard Streamlit & Google Sheets

Contexte : Windows, Proxy Entreprise, GitHub, Google Cloud.

---

## Étape 1 : Préparation du terrain (L'Isolation)

L'objectif est de créer une "bulle" (environnement virtuel) pour éviter les conflits entre projets.

1. **Créer le dossier** : Crée ton dossier projet (ex: monfinary2).
  2. **Créer la bulle** : Ouvre un terminal dans ce dossier et tape :  
python -m venv .venv
  3. **Activer la bulle** : .\venv\Scripts\activate**Note** : Le préfixe (.venv) doit apparaître dans le terminal. C'est le signe que tu travailles dans ton environnement isolé.
  4. **Lien Spyder** : Dans Spyder > Tools > Preferences > Python Interpreter, sélectionne "**Use the following Python interpreter**" et pointe sur le python.exe situé dans .venv\Scripts\.
- 

## Étape 2 : Liste de courses (requirements.txt)

Crée un fichier nommé requirements.txt à la racine de ton projet.

- Contenu type : streamlit, st-gsheets-connection, pandas.
- 

## Étape 3 : Percer le Proxy (Accès Internet)

Nécessaire pour télécharger les bibliothèques sur le réseau d'entreprise.

1. **Variables de session (Terminal) :**

PowerShell

```
$env:HTTP_PROXY="http://user:password@proxy:port"  
$env:HTTPS_PROXY="https://user:password@proxy:port"
```

2. Installer les outils :  
python -m pip install -r requirements.txt
-

## Étape 4 : Sauvegarder sur GitHub (Le coffre-fort)

1. Initialiser Git : git init
2. Créer le `.gitignore` : Crée un fichier nommé `.gitignore` (sans extension) et ajoute ces lignes :

```
Plaintext  
.venv/  
.streamlit/  
*.json  
*.toml
```

*Indispensable pour ne jamais uploader tes clés privées sur GitHub.*

3. Lier et Envoyer :

```
PowerShell  
git remote add origin URL_DU_REPO_GITHUB  
git add .  
git commit -m "Initial commit"  
git push -u origin main
```

---

## Étape 5 : Accès aux données Google (La Sécurité)

Le pont sécurisé entre ton Sheet privé et ton code.

1. **Console Google Cloud** : Crée un projet -> Active l'API "**Google Sheets API**".
2. **Service Account** : Crée un compte de service -> Génère une clé au format **JSON**.
3. **L'invitation** : Copie l'adresse mail du service account (...gserviceaccount.com) et ajoute-la comme **Lecteur** sur ton Google Sheet via le bouton **Partager**.
4. **Secrets Streamlit** : \* Crée un dossier `.streamlit/` à la racine.
  - Crée un fichier `secrets.toml` à l'intérieur.
  - Recopie les données du JSON dans la section [connections.gsheets] du TOML.

---

## Étape 6 : Le code final (app.py)

Voici la structure minimale pour afficher tes données.

Python

```
import streamlit as st
from streamlit_gsheets import GSheetsConnection

# 1. Configuration de la page
st.set_page_config(page_title="Mon Dashboard", layout="centered")

# 2. Connexion au Sheet
conn = st.connection("gsheets", type=GSheetsConnection)
url = "https://docs.google.com/spreadsheets/d/TON_ID_ICI/edit#gid=0"

# 3. Lecture et Affichage
df = conn.read(spreadsheet=url, ttl=0)

st.title("📊 Mes Données Financières")
st.dataframe(df, use_container_width=True)
```



## Routine de travail (Réutilisation)

Pour reprendre le projet ou en lancer un nouveau basé sur ces notes :

1. **Ouvrir le terminal** dans le dossier.
  2. **Activer** : .\venv\Scripts\activate
  3. **Lancer l'app** : streamlit run src/app.py
  4. **Mettre à jour GitHub** : git add . -> git commit -m "update" -> git push
-