

API DATA GOUV -> GRIST

Script d'appel à synchronisation des données de DATA GOUV dans GRIST



1) Insérer le fichier CSV ci-dessous " importer un fichier"

identifiant_de_l_etablissement,nom_.csv

2) Créer une vue page ou une page avec "Custom widget"

3) Copier le code ci-dessous dans la partie HTML en faisant bien attention de faire appel à la bonne table " const table = grist.getTable("Annuaire_etab");"

```
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="utf-8">
  <title>Import Lycées La Réunion</title>
  <script src="https://docs.getgrist.com/grist-plugin-api.js"></script>
<style>
  #importEtabBtn {
    margin: 10px;
```

```
padding: 10px 20px;
background-color: #4CAF50;
color: white;
border: none;
border-radius: 4px;
cursor: pointer;
font-family: sans-serif;
font-size: 14px;
}
#importEtabBtn:hover {
  background-color: #45a049;
}
</style>
</head>
<body>
  <div style="font-family: sans-serif; padding: 1em;">
    <h2>Import des établissements depuis data.education.gouv.fr</h2>
    <p>
      Cliquez sur le bouton ci-dessous pour mettre à jour la table "annuaire_etab"
      avec les données des lycées de La Réunion.
    </p>
    <button id="importEtabBtn">Mettre à jour l'annuaire</button>
    <p id="status" style="margin-top: 10px; color: #666;"></p>
  </div>

  <script>
    grist.ready({requiredAccess: 'full'});

    async function fetchEtab() {
      const apiUrl = "https://data.education.gouv.fr/api/explore/v2.1/catalog/datasets/fr-en-annuaire-
education/records?select=identifiant_de_l_etablissement%2Cnom_etablissement%2Ctype_etablissement%2Csta
tut_public_privé%2Cadresse_1%2Cadresse_2%2Cadresse_3%2Ccode_postal%2Cnom_commune%2Cvoie_genera
le%2Cvoie_technologique%2Cvoie_professionnelle%2Ctelephone%2Cfax%2Cweb%2Cmail%2Crestauration%2Ch
bergement%2Cculis%2Capprentissage%2Csegpa%2Csection_arts%2Csection_cinema%2Csection_theatre%2Cse
ction_sport%2Csection_internationale%2Csection_europeenne%2Clycee_agricole%2Clycee_militaire%2Clycee_d
es_metiers%2Cpost_bac%2Cappartenance_education_prioritaire%2Cgreta%2Cnombre_d_eleves%2Cfiche_onisep
%2Ctype_contrat_privé%2Cnom_circonscription%2Clatitude%2Clongitude%2Cprecision_localisation%2Cdate_ma
j_ligne%2Ccode_nature%2Clibelle_nature%2Ccode_type_contrat_privé%2Cpial%2Cetablissement_mere%2Ctype
_rattachement_etablissement_mere&limit=100&refine=type_etablissement%3A%22Lyc%3A%22&refine=li
belle_academie%3A%22La%20R%3A%22";
```

```

try {
  document.getElementById("status").textContent = "Récupération des données en cours...";
  const response = await fetch(apiUrl);
  if (!response.ok) {
    throw new Error("HTTP " + response.status);
  }
  const data = await response.json();
  const results = data.results || [];
  document.getElementById("status").textContent =
    results.length + " établissements trouvés.";
  return results;
} catch (error) {
  document.getElementById("status").textContent =
    "Erreur lors de la récupération des données.";
  console.error("Erreur API :", error);
  return [];
}
}

```

```

async function insertIntoGrist(etabs) {
  const table = grist.getTable("Annuaire_etab");

  const records = etabs.map(etab => ({
fields: {
  identifiant_de_l_etablissement: etab.identifiant_de_l_etablissement,
  nom_etablissement: etab.nom_etablissement,
  type_etablissement: etab.type_etablissement,
  statut_public_prive: etab.statut_public_prive,
  adresse_1: etab.adresse_1 || "",
  adresse_2: etab.adresse_2 || "",
  adresse_3: etab.adresse_3 || "",
  code_postal: etab.code_postal,
  nom_commune: etab.nom_commune,
  voie_generale: etab.voie_generale,
  voie_technologique: etab.voie_technologique,
  voie_professionnelle: etab.voie_professionnelle,
  telephone: etab.telephone,
  fax: etab.fax || "",
  web: etab.web || "",

```

```
mail: etab.mail || "",
restauration: etab.restauration || "",
hebergement: etab.hebergement || "",
ulis: etab.ulis || "",
apprentissage: etab.apprentissage || "",
segpa: etab.segpa || "",
section_arts: etab.section_arts || "",
section_cinema: etab.section_cinema || "",
section_theatre: etab.section_theatre || "",
section_sport: etab.section_sport || "",
section_internationale: etab.section_internationale || "",
section_europeenne: etab.section_europeenne || "",
lycee_agricole: etab.lycee_agricole || "",
lycee_militaire: etab.lycee_militaire || "",
lycee_des_metiers: etab.lycee_des_metiers || "",
post_bac: etab.post_bac || "",
appartenance_education_prioritaire: etab.appartenance_education_prioritaire || "",
greta: etab.greta || "",
nombre_d_eleves: etab.nombre_d_eleves || "",
fiche_onisep: etab.fiche_onisep || "",
type_contrat_prive: etab.type_contrat_prive || "",
nom_circonscription: etab.nom_circonscription || "",
latitude: etab.latitude,
longitude: etab.longitude,
precision_localisation: etab.precision_localisation || "",
date_maj_ligne: etab.date_maj_ligne || "",
code_nature: etab.code_nature || "",
libelle_nature: etab.libelle_nature || "",
code_type_contrat_prive: etab.code_type_contrat_prive || "",
pial: etab.pial || "",
etablissement_mere: etab.etablissement_mere || "",
type_rattachement_etablissement_mere: etab.type_rattachement_etablissement_mere || ""
}
}));
```

```
try {
  document.getElementById("status").textContent =
    "Insertion des données dans Grist...";
  await table.create(records);
  document.getElementById("status").textContent =
```

```
    records.length + " établissements ajoutés avec succès !";
} catch (error) {
    document.getElementById("status").textContent =
        "Erreur lors de l'insertion des données.";
    console.error("Erreur Grist :", error);
}
}

async function importEtab() {
    const etabs = await fetchEtab();
    if (etabs.length > 0) {
        await insertIntoGrist(etabs);
    }
}

document
    .getElementById("importEtabBtn")
    .addEventListener("click", importEtab);
</script>
</body>
</html>
```

Révision #4

Créé 26 novembre 2025 06:34:00 par Dalleau Patrick

Mis à jour 3 décembre 2025 08:50:17 par Dalleau Patrick