

Exercice 1 - pandas

Consigne : Lire un fichier CSV de capteurs, calculer les moyennes, max et min. Enregistrer un résumé dans 'resume.txt'.

Fichier associé : capteurs.csv

Exercice 2 - json

Consigne : Lire un fichier JSON d'état machine. Vérifier le status. Si != 'running', écrire une alerte dans 'log.txt'.

Fichier associé : machine_status.json

Exercice 3 - matplotlib / plotly

Consigne : Afficher un graphique température/temps à partir de 'capteurs.csv'. Enregistrer le graphique en .png.

Fichier associé : a creer ou reutiliser un existant

Exercice 4 - xml.etree.ElementTree

Consigne : Lire un fichier XML de config machine. Vérifier les paramètres, générer un rapport JSON.

Fichier associé : config_machine.xml

Exercice 5 - PySide6 / PyQt5

Consigne : Créer une interface pour ouvrir un fichier CSV et afficher un tableau des valeurs avec stats.

Fichier associé : a creer ou reutiliser un existant

Exercice 6 - PySide6 / PyQt5

Consigne : Créer une UI avec champ texte + slider pour saisir une configuration, enregistrée en JSON.

Fichier associé : a creer ou reutiliser un existant

Exercice 7 - argparse / pandas

Consigne : Créer un script CLI qui analyse tous les fichiers CSV d'un dossier et génère un rapport global.

Fichier associé : a creer ou reutiliser un existant

Exercice 8 - requests / json

Consigne : Appeler une API météo simulée et sauvegarder les infos dans un fichier CSV.

Fichier associé : a creer ou reutiliser un existant

Exercice 9 - sqlite3 / pandas

Consigne : Créer une base SQLite 'production.db' et importer les données de 'capteurs.csv'.

Fichier associé : a creer ou reutiliser un existant

Exercice 10 - sqlite3 / pandas

Consigne : Extraire toutes les températures > 25°C depuis la base SQLite et sauvegarder dans 'alertes.csv'.

Fichier associé : a creer ou reutiliser un existant