

## Projet SUNU COUTURE – Extension Python

Ce document constitue la suite du projet SUNU COUTURE, initialement réalisé en HTML5 et CSS3. Nous allons maintenant y ajouter une couche de logique et de traitement de données en Python en s'appuyant sur les compétences acquises dans le cours "Apprenez les bases du langage python".

---

### Objectif de cette étape

Rendre le formulaire de prise de mesures plus intelligent :

- Enregistrer et manipuler les données avec Python (variables, listes, dictionnaires)
  - Utiliser des fonctions pour organiser le traitement
  - Appliquer des conditions pour valider ou traiter les données
  - Gérer des boucles pour les entrées utilisateurs multiples
  - Utiliser des modules standards pour traiter des cas concrets (datetime, math, json...)
  - Faire de petits traitements de texte (mise en forme de noms, nettoyage)
  - Bonus : simuler une base de données avec un fichier .json
  - Créer une interface en ligne de commande avec menu numéroté
- 

### Fonctions à implémenter en Python

#### 1. Création d'une fiche client

- Créer un dictionnaire `client` avec les champs suivants :
  - nom
  - prénom
  - email
  - téléphone
  - type de vêtement
  - sexe
  - mesures : taille, hanche, longueur
  - options : livraison, retouche, commentaire

#### 2. Fonction de saisie

Créer une fonction `saisir_commande()` qui demande à l'utilisateur de remplir chaque champ du dictionnaire via `input()`, avec validation des données (ex. taille > 0).

#### 3. Nettoyage et formatage des chaînes

- Utiliser `.strip()`, `.capitalize()` ou `.lower()` pour nettoyer et normaliser les saisies.

#### 4. Ajout de la date de commande

- Ajouter automatiquement la date du jour à chaque commande (module `datetime`).

#### 5. Stockage en liste

- Enregistrer chaque commande sous forme de dictionnaire dans une liste `commandes`.

#### 6. Affichage récapitulatif

- Créer une fonction `afficher_commandes()` qui affiche toutes les commandes enregistrées avec un joli format.

#### 7. Calculs simples avec les mesures

- Fonction `calcul_moyenne_mesure(commandes)` qui calcule la moyenne des tailles, hanches et longueurs.
- Fonction `taille_recommandee(mesure)` qui recommande un type de tissu ou coupe selon les valeurs.

#### 8. Sauvegarde dans un fichier JSON

- Sauvegarder toutes les commandes dans un fichier `commandes.json` à la fin.
- Ajouter une fonction `charger_commandes()` pour relire le fichier plus tard.

#### 9. Création d'un menu interactif en ligne de commande

- Proposer à l'utilisateur une interface de menu :
  1. Ajouter une nouvelle commande
  2. Afficher toutes les commandes
  3. Sauvegarder les commandes
  4. Charger les commandes depuis le fichier
  5. Statistiques sur les mesures
  6. Rechercher une commande par nom
  7. Afficher les 3 derniers clients
  8. Quitter le programme
- Créer une boucle principale `while True:` avec un `input()` pour que l'utilisateur choisisse une option.
- Chaque option appelle une des fonctions décrites ci-dessus.

---

### Prolongements possibles (100% Python)

- Ajouter une simulation de génération de numéro de commande automatique
- Lire un tableau HTML avec `BeautifulSoup` (bonus scraping)
- Créer une interface terminale avec un menu principal
- Analyser les statistiques sur les commandes :
  - Combien de commandes Homme / Femme ?

- Quel vêtement est le plus demandé ?
    - Afficher les 3 derniers clients
  - Ajouter une option pour **rechercher** une commande par nom ou téléphone
-