

# Analyste de données certifié Python – Niveau de base

Formation (présentiel ou distance) pour maîtriser les bases de Python et les principales bibliothèques d'analyse de données (NumPy, pandas, Matplotlib). Elle constitue une excellente introduction à l'analytique de données et prépare à une première certification d'analyste de données en Python.

Distributeur officiel Certiport

Centre d'examen Pearson VUE

Formateurs certifiés Microsoft

<b>5 jours</b> 35 heures	<b>Examen</b> PCED-30-01
<b>Modalité</b> Présentiel Distanciel	<b>Niveau</b> Débutant

## INSCRIPTION / RÉSERVATION



Je m'inscris  
maintenant



### OBJECTIFS PÉDAGOGIQUES

- Acquérir les bases de Python pour l'analyse de données.
- Importer, préparer et manipuler des données avec NumPy et pandas.
- Créer des visualisations claires avec Matplotlib et Seaborn.
- Appliquer les notions clés de statistiques descriptives et d'analyse exploratoire.
- Réaliser un mini-projet d'analyse en Python pour préparer la certification.

### PUBLIC CIBLE

- Débutants, étudiants et personnes en reconversion vers la data.
- Professionnels voulant analyser leurs données avec Python.
- Futurs data analysts ou data scientists juniors.
- Enseignants souhaitant intégrer Python à leurs cours.

### PRÉREQUIS

- Débutants acceptés, aucune expérience en Python requise.
- Notions de base en bureautique souhaitables.
- Compréhension minimale des concepts numériques (pourcentages, moyennes, etc.).

## PROGRAMME DE LA FORMATION – DÉTAILLÉ

### **Introduction à Python pour l'analyse de données**

#### **Environnement de travail & prise en main**

- Présentation de Python et de ses usages en data analysis.
- Installation de Python, Anaconda, Jupyter Notebook / JupyterLab.
- Découverte de l'environnement de travail : notebooks, cellules, exécution pas à pas.
- Organisation des projets et bonnes pratiques de base (structure des dossiers, sauvegardes).

#### **Syntaxe de base de Python**

- Variables, types de données (int, float, bool, str).
- Opérateurs arithmétiques, de comparaison et logiques.
- Entrées / sorties simples, affichage formaté.

#### **Structures de données fondamentales**

- Listes, tuples, ensembles (sets) et dictionnaires.
- Accès aux éléments, ajout, suppression, modification.
- Parcours et itérations sur les structures de données.

#### **Contrôle de flux & fonctions**

- Conditions : if, elif, else.
- Boucles for et while, compréhension de listes (list comprehensions).
- Définition et appel de fonctions, paramètres et valeurs de retour.
- Notion de modules et d'import (import, from ... import).

### **NumPy : base du calcul scientifique**

#### **Introduction à NumPy**

- Pourquoi NumPy ? Avantages par rapport aux listes Python.
- Création de tableaux (ndarray) : à partir de listes, de fonctions NumPy.
- Dimensions, formes et types de données (shape, dtype).

#### **Opérations sur les tableaux**

- Indexation, slicing, sélection conditionnelle.
- Opérations vectorisées : addition, multiplication, etc.
- Fonctions mathématiques de base (min, max, mean, sum, std).



## PROGRAMME DE LA FORMATION – DÉTAILLÉ

### **Génération et manipulation de données**

- Génération de données aléatoires (normal, uniforme...).
- Reshape, concaténation et empilage (stack, concatenate).
- Conversion entre tableaux NumPy et listes Python.

### **Pandas : manipulation de données tabulaires**

#### **Objets fondamentaux : Series et DataFrame**

- Création d'une Series et d'un DataFrame.
- Index, colonnes, types de données et affichage des premières lignes.
- Chargement de données depuis des fichiers CSV, Excel, JSON.

#### **Sélection, filtrage et tri**

- Sélection de colonnes et de lignes (loc, iloc).
- Filtres conditionnels simples et multiples.
- Tri des données (sort\_values, sort\_index).

#### **Transformation et agrégation**

- Création et modification de colonnes calculées.
- Fonctions apply, map, transform sur les colonnes.
- Groupement et agrégation (groupby, agg).

#### **Jointures et fusion de données**

- Concaténation de DataFrames.
- Jointures de type SQL (merge : inner, left, right, outer).
- Cas d'usage : fusion de tables clients, ventes, produits.

### **Nettoyage et préparation des données**

#### **Gestion des valeurs manquantes et des doublons**

- Détection des valeurs manquantes (isna, isnull).
- Suppression ou imputation (dropna, fillna).
- Détection et suppression des doublons (duplicated, drop\_duplicates).

#### **Types de données et formatage**

- Conversion de types (astype) : numériques, chaînes, booléens.
- Traitement des dates et heures (to\_datetime, resample simple).
- Nettoyage de chaînes de caractères (strip, lower, replace, split).



## PROGRAMME DE LA FORMATION – DÉTAILLÉ

### **Normalisation simple et préparation**

- Mise à l'échelle de variables (normalisation basique).
- Codage simple de variables catégorielles (get\_dummies).
- Constitution d'un jeu de données propre pour l'analyse.

### **Visualisation de données avec Python**

#### **Introduction à Matplotlib**

- Structure d'une figure : figure, axes, légendes, titres.
- Graphiques de base : courbes, barres, histogrammes, secteurs (camemberts).
- Personnalisation (couleurs, styles, labels et annotations simples).

#### **Visualisation avec Seaborn (niveau initiation)**

- Présentation de Seaborn et intégration avec pandas.
- Graphiques orientés analyse : distplot / histplot, boxplot, countplot.
- Nuages de points et cartes de chaleur (heatmap de corrélations).

#### **Bonnes pratiques de data storytelling**

- Choisir le bon graphique pour le bon message.
- Simplifier les visualisations pour un public non technique.
- Export des graphiques pour un rapport ou une présentation.

### **Statistiques descriptives pour l'analyste de données**

#### **Rappels de statistiques descriptives**

- Mesures de tendance centrale : moyenne, médiane, mode.
- Mesures de dispersion : variance, écart-type, minimum, maximum, quartiles.
- Distribution des données et interprétation de base.

#### **Corrélations et relations entre variables**

- Notion de corrélation : coefficient de corrélation et interprétation.
- Visualisation des relations : nuages de points et heatmap.
- Limites de la corrélation (corrélation  $\neq$  causalité).



## PROGRAMME DE LA FORMATION – DÉTAILLÉ

### **Premiers pas vers la modélisation**

- Introduction à la régression linéaire simple (concepts, pas de démonstration mathématique avancée).
- Exemple illustratif avec un jeu de données simple.
- Interprétation basique des résultats pour un public métier.

### **Projet de synthèse & préparation à la certification**

#### **Mini-projet d'analyse de données**

- Choix d'un jeu de données réel (marketing, ventes, finance, RH, etc.).
- Définition d'une problématique métier simple (ex. segmentation clients, analyse de ventes, performance d'une campagne).
- Application de la démarche complète : import, nettoyage, exploration, visualisation et synthèse.

#### **Préparation à l'évaluation & certification**

- Révision structurée des notions clés du cours.
- Exemples de questions théoriques et exercices pratiques.
- Conseils pour structurer un notebook de projet clair et professionnel.

#### **Passage de la certification "Analyste de données certifié Python – Niveau de base"**

- Mise en situation d'évaluation (QCM + cas pratique court).
- Critères d'évaluation : compréhension des bases Python, manipulation de données, visualisation, interprétation.
- Retours personnalisés et plan de progression proposé au participant.

