

# Python est désormais le plus populaire mais maintenant, il doit s'améliorer

De nos jours, Python est devenu l'un des langages les mieux notés dans le monde de la programmation. Selon plusieurs ressources telles que le IEEE Spectrum Top Programming Languages ou le PYPL (Popularity of programming language), Python occupe la deuxième place dans le classement des langages.

La popularité de ce langage se traduit par les différentes fonctionnalités qu'il offre tel que la forte analyse de données, la puissance de calcul ainsi que l'intelligence artificielle. Elle est aussi due à sa polyvalence, il n'est plus utilisé que par les développeurs, mais aussi par des data scientists, dans l'intelligence artificielle, et même la science.

Par ailleurs, comme chaque langage informatique, Python a ses avantages ainsi que ses défauts . C'est ce que nous allons découvrir par la suite !

## 1. Python, les points forts

### 1.1. Simplicité de syntaxe

Python est très similaire à la langue anglaise utilisée dans la vie quotidienne. Cette simplicité de syntaxe vous permettra de gérer et coder des programmes très complexes tout en étant à l'aise. Grâce à cela, les codeurs débutants peuvent apprendre ce langage et rejoindre la communauté de programmation plus rapidement !

Ce langage offre aussi une représentation des données dans un format facile à comprendre à l'aide de différents graphiques. Ils constituent un moyen efficace de représenter et de comprendre visuellement les données. Les entreprises de développement Web utilisent des bibliothèques Python (par exemple, Matplotlib) qui permettent de visualiser les données et de créer des rapports clairs et faciles à comprendre.

### 1.2. Orienté-objet

Python prend en charge la programmation orientée procédure et la programmation orientée objet. Les programmes construits autour de l'objet fusionnent les données et les fonctionnalités dans la programmation orientée objet. Python a une façon plus facile de faire OOP surtout par rapport à d'autres langues comme C++ et Java.

### 1.3. Langage interprété

Python ne nécessite pas de compilation binaire comme les autres langages Java C, ou C++. Le programme s'exécute directement à partir du code source. Python convertit le code source en un code intermédiaire à partir de bytecodes appelés, traduit le bytecodes en langage native et l'exécute. Ce processus le rend beaucoup plus facile et nous n'avons pas à nous soucier de compiler des programmes et de charger des bibliothèques, etc. Nous devons juste nous assurer que les bibliothèques appropriées sont liées et chargées.

#### 1.4. Extensible

Python est un langage extensible vers d'autres langages. Il prend en charge l'utilisation d'autres langages de programmation comme le code C ou C++ pour appeler du code python. Ce processus est nécessaire lorsque nous voulons exécuter un morceau de code critique.

Les propriétés portables et extensibles de Python vous permettent d'effectuer des opérations cross-langage de manière transparente. Python est supporté par la plupart des plates-formes présentes dans l'industrie aujourd'hui allant de Windows à Linux passant par Macintosh.

Les fonctionnalités d'extensibilité de Python vous permettent d'intégrer Java ainsi que les composants .NET . Vous pouvez également invoquer les bibliothèques C et C++.

#### 1.5. Développement web :

Python a un ensemble de cadres pour développer des sites Web. Les frameworks populaires sont Django, Flask, Pylons etc. Puisque ces frameworks sont écrits en Python, c'est la raison principale qui rend le code beaucoup plus rapide et stable.

Vous pouvez également effectuer du Web scraping où vous pouvez récupérer des détails à partir d'autres sites Web. Vous serez également impressionné que de nombreux sites Web tels qu'Instagram, Pinterest, et Netflix sont construits à partir de ces frameworks seulement.

#### 1.6. Big Data

Python gère beaucoup de tracas de données. Il prend en charge l'informatique parallèle où vous pouvez utiliser Python pour Hadoop ainsi. En Python, vous avez une bibliothèque appelée « Pydoop » et vous pouvez écrire un programme MapReduce et traiter les données présentes dans le cluster HDFS. Il existe d'autres bibliothèques comme « Dask » et « Pyspark » pour le traitement des mégadonnées. Par conséquent, Python est largement utilisé pour le Big Data où vous pouvez facilement le traiter !

#### 1.7. L'intelligence artificielle

L'IA est le prochain grand développement dans le monde de la technologie. Vous pouvez réellement créer une machine qui imite le cerveau humain et qui a le pouvoir de penser, d'analyser et de prendre des décisions. En outre, des bibliothèques comme Keras et TensorFlow apportent des fonctionnalités d'apprentissage automatique dans sa globalité. Ces bibliothèques peuvent donner la capacité à une machine d'apprendre sans être explicitement programmé, c'est ce qu'on appelle le unsupervised Learning. De plus, nous avons des bibliothèques comme openCv qui aident la vision par ordinateur ou la reconnaissance d'images.

#### 1.8. Framework de test

Python est idéal pour la validation d'idées ou de produits pour les entreprises. Il a de nombreux frameworks de test intégrés qui couvrent le débogage et les flux de travail les plus rapides. Il y a beaucoup d'outils et de modules pour faciliter les choses comme le sélénium et le Splinter. Il prend en charge les tests multi-plateformes et multi-navigateurs avec des frameworks tels que PyTest et Robot Framework. Tester est une tâche fastidieuse et Python est le booster pour elle, donc chaque testeur devrait certainement choisir Python !

### 2. Python, les points faibles

## 2.1. Lenteur d'exécution

Python est souvent critiqué pour sa vitesse d'exécution, c'est un langage de script interprété, ce qui le rend relativement plus lent que beaucoup de ses homologues compilés, comme C/C++ ou Java, en raison des différentes méthodes qu'il utilise pour traduire du code. Pourtant, certains benchmarks Python fonctionnent plus rapidement que ceux de C et C++. Ceci dit, Python n'est pas le seul avec des problèmes de vitesse potentiels. Ruby, Perl, et même JavaScript ont des contraintes et limitations de vitesse. Certains problèmes liés à la vitesse ont été traités et optimisés, de sorte que Python reste l'un des meilleurs choix des équipes de développement de logiciels. Mais ce problème de vitesse peut pousser plusieurs développeurs à ne pas s'intéresser à ce langage !

## 2.2. Support multiprocesseur très faible

Le multitraitement est une partie importante dans la programmation d'une application. Python prend en charge le multitraitement, bien qu'il ne soit pas aussi flexible ou pratique que d'autres langages. Cela peut créer certaines limitations lorsque vous écrivez le code.

## 2.3. Le langage le moins apprécié pour le développement d'applications mobiles

Ce n'est pas un mauvais langage pour le développement mobile. C'est juste que peu d'entreprises l'utilisent pour ce but, préférant le développement natif pour iOS et Android ou le développement React Native. Vous aurez probablement du mal à recruter des développeurs ayant de l'expérience dans le développement mobile Python, aussi, pour la même raison. Il n'est tout simplement pas aussi populaire que les autres technologies dans ce domaine.

## 2.4. Contraintes de design

Python est dactylographié dynamiquement, ce qui signifie qu'il exécute certaines tâches pendant l'exécution de l'application qui seraient autrement effectuées dans un langage typé statiquement. Cela met certaines restrictions sur la conception. Si votre conception est chargée avec des éléments, il pourrait bloquer le programme et empêcher le fonctionnement en douceur. Une autre chose que vous devez savoir lorsque vous envisagez Python pour votre projet est que la concurrence et le parallélisme ne sont pas destinés à une utilisation élégante en Python. Pour cette raison, le design pourrait ne pas sembler aussi sophistiqué que vous le souhaitez.

## 2.5. Utilisation excessive de la mémoire

Ce langage est connu pour la flexibilité de ses types de données. Cela se traduit par une consommation de mémoire assez élevée et rend son utilisation pour des tâches à forte intensité de mémoire pas très recommandée. D'un autre côté, Pour offrir une certaine simplicité aux programmeurs et aux développeurs, Python doit faire des compromis. Ce langage utilise une énorme quantité de mémoire, qui agit comme un inconvénient, surtout lorsque vous développez une application avec une préférence pour l'optimisation de la mémoire.

## Conclusion

Python vous permet de développer des applications claires et simples qui sont faciles à obtenir d'un petit projet à une application complète et complexe. Que vous soyez un programmeur débutant apprenant à coder ou un propriétaire d'entreprise, Python peut être une bonne option pour de nombreux types de projets. Il est reconnu comme l'un des meilleurs langages de programmation pour les startups et il est facile de voir pourquoi lorsque vous comparez côte à côte les avantages de Python avec le besoin des startups.

Les startups sont constamment à la recherche de certitude et de risques réduits, elles disposent de ressources limitées et ont besoin d'espace pour croître. D'autre part, Python est flexible et facile à adapter, ne nécessite pas une grande équipe, et peut être utilisé pour construire des prototypes et des MVP.

Maintenant, vous êtes capable de décider si Python est adapté pour vos choix de programmation et d'applications ou non. Ceci dit, le choix reste à vous !