



VEILLE TECHNOLOGIQUE

Python

Florian Hostachy

2023/2/16

Table des matières

Introduction	3
Objectif de la veille.....	3
Objectifs.....	3
Développement	4
Python c'est quoi :	4
Applications connues développer sous python :	4
Brève présentation de C#.....	5
Les différences entre python et C#.....	5
Quelques exemples de petits projets en python :	7
Conclusion	10
Annexe.....	11

Introduction

Objectif de la veille

Nous allons réaliser une veille technologique sur le langage de programmation python on pourra aussi voir les différences présences avec le langage C# et notamment les différents avantage et inconvénient entre les deux. Il est important de préciser que python est un langage plus ancien que C# il est a donc pu être améliorer au fils des années. Je vais commencer par expliquer les différences notables entre les deux logiciels avec diverses sources pour par la suite me faire mon propre avis sur le sujet et voir le quelle des deux langages à des qualité et des inconvénients pour cela je réaliserais un projet en python ayant déjà réaliser plusieurs projets en C# je pourrais vous donner un retour constructif sur le sujet.

Objectifs

- Réalisé un rapport de veille technologique et découvrir (python)
- Faire une application fonctionnelle avec python
- Donner des sources pour nos recherches
- Voir les différences notables entre C# et python

Développement

Python c'est quoi :

Python est un langage de programmation interprété, multiparadigme et multiplateformes. Il favorise la programmation impérative structurée, fonctionnelle et orientée objet. Il est doté d'un typage dynamique fort, d'une gestion automatique de la mémoire par ramasse-miettes et d'un système de gestion d'exceptions. Il est le open source le plus employé par les informaticiens. Ce langage s'est propulsé en tête de la gestion d'infrastructure, d'analyse de données ou dans le domaine du développement de logiciels. En effet, parmi ses qualités, Python permet notamment aux développeurs de se concentrer sur ce qu'ils font plutôt que sur la manière dont ils le font. Il a libéré les développeurs des contraintes de formes qui occupaient leur temps avec les langages plus anciens. Ainsi, développer du code avec Python est plus rapide qu'avec d'autres langages.

Applications connues développer sous python :

- Instagram : Instagram utilise Python pour gérer son infrastructure backend et pour développer ses fonctionnalités de traitement d'image.
- Dropbox : Dropbox a été initialement développé en Python et utilise toujours Python pour de nombreuses tâches, y compris le développement web et l'automatisation des tâches.
- YouTube : YouTube utilise Python pour gérer les processus de traitement vidéo et pour l'analyse de données.

On peut voir avec les exemples cités que python est souvent utilisé pour réaliser des tâches en backend ou faire de l'automatisation de tâche. Il est en entreprises très utilisé et apporte un gain de temps considérable.

C# est un langage de programmation orientée objet, commercialisé par Microsoft depuis 20023 et destiné à développer sur la plateforme Microsoft .NET.

Il est dérivé du C++ et très proche du Java dont il reprend la syntaxe générale ainsi que les concepts, y ajoutant des notions telles que la surcharge des opérateurs, les indexeurs (en) et les délégués. Il est utilisé notamment pour développer des applications web sur la plateforme ASP.NET.

Les différences entre python et C#

C#	Python
C# est développé par Microsoft et est disponible gratuitement, même à des fins commerciales. L'écosystème .NET est entièrement open source et appartient à .NET Foundation, une organisation indépendante.	Python est également une plate-forme open source et disponible gratuitement, même à des fins commerciales.
C # prend en charge la programmation orientée objet, la programmation structurée, la programmation fonctionnelle et bien d'autres	Alors que python prend en charge les langages de programmation multi-paradigmes tels que les OOP et la programmation structurée et bien d'autres.
C # nécessite le SDK et l'exécution .NET. L'écosystème .NET offre une interopérabilité avec des langages tels que F #, VB.NET, JavaScript, Python, etc.	Bien qu'il puisse également être intégré à .NET, C, JavaScript et java.

Bien qu'il s'agisse principalement d'un langage compilé, C # prend également en charge les environnements de développement d'interpréteurs interactifs à l'aide de dotnet-interactive. Vous écrivez également du C# dans les notebooks Jupyter.	Python prend en charge l'interpréteur interactif pour écrire des programmes.
C# est un langage de type statique.	Alors que le langage python est un langage de type dynamique.

En bref, C # et Python sont tous deux des langages de haut niveau, orientés objet et faciles à apprendre. Ils assurent un développement rapide et de bonnes performances. Cependant, C# est plus clair et organisé, et il est beaucoup plus rapide à l'exécution. Alors que Python est plus facile à apprendre et à écrire que C # et possède de vastes bibliothèques standard.

C# et Python sont tous deux d'excellents langages de programmation. Ainsi, choisir l'un plutôt que l'autre est plus une question de préférence que le risque de choisir la mauvaise langue pour le projet. Alors, faites votre choix et créons ensemble un logiciel unique !

Quelques exemples de petits projets en python :

Calculatrice basique :

```
# Définition de la fonction qui effectue les calculs
def calculer(operation, a, b):
    if operation == "+":
        return a + b
    elif operation == "-":
        return a - b
    elif operation == "*":
        return a * b
    elif operation == "/":
        if b != 0:
            return a / b
        else:
            return "Erreur: division par zéro"

# Demande de l'opération à effectuer et des deux nombres à calculer
operation = input("Entrez l'opération à effectuer (+, -, *, /) : ")
a = float(input("Entrez le premier nombre : "))
b = float(input("Entrez le deuxième nombre : "))

# Affichage du résultat
resultat = calculer(operation, a, b)
print("Résultat : ", resultat)
```

Résultat :

```
Entrez l'opération à effectuer (+, -, *, /) : *
Entrez le premier nombre : 54
Entrez le deuxième nombre : 450
Résultat : 24300.0
```

On peut donc choisir l'opération à réaliser avec le bon opérateur pour ensuite mettre les deux chiffres et on obtient le résultat.

Chifoumi :

```
# Coup aléatoirement choisie par l'ordinateur
from random import choice

COUP = ("Pierre", "Feuille", "Ciseaux")

# Demande si le joueur veut jouer si il répond non il aura le droit au message bye bye
while input("Jouez (y/n): ").lower() != "n":

    print("\n-----")
    print("Le jeu du: Pierre - Feuille - Ciseaux")
    print("-----\n")

    a = int(input("Choisissez un chiffre:\n0: Pierre\n1: Feuille\n2: Ciseaux\n-> "))
    b = choice(range(3))

    print("\n{} VS {}".format(COUP[a], COUP[b]))
    if a == b:
        print("ÉGALITÉ\n")
    # Opération boolean pour savoir qui gagne |
    elif (a>b and b+1==a) or (a<b and a+b==2):
        print("VOUS GAGNEZ\n")
    else:
        print("VOUS PERDEZ\n")
else:
    print("Bye Bye")
```

Résultat :

```
Choisissez un chiffre:
0: Pierre
1: Feuille
2: Ciseaux
-> 0

Pierre VS Feuille
VOUS PERDEZ

Jouez (y/n): y

-----
Le jeu du: Pierre - Feuille - Ciseaux
-----

Choisissez un chiffre:
0: Pierre
1: Feuille
2: Ciseaux
-> 2

Ciseaux VS Pierre
VOUS PERDEZ

Jouez (y/n): y

-----
Le jeu du: Pierre - Feuille - Ciseaux
-----

Choisissez un chiffre:
0: Pierre
1: Feuille
2: Ciseaux
-> 1

Feuille VS Pierre
VOUS GAGNEZ

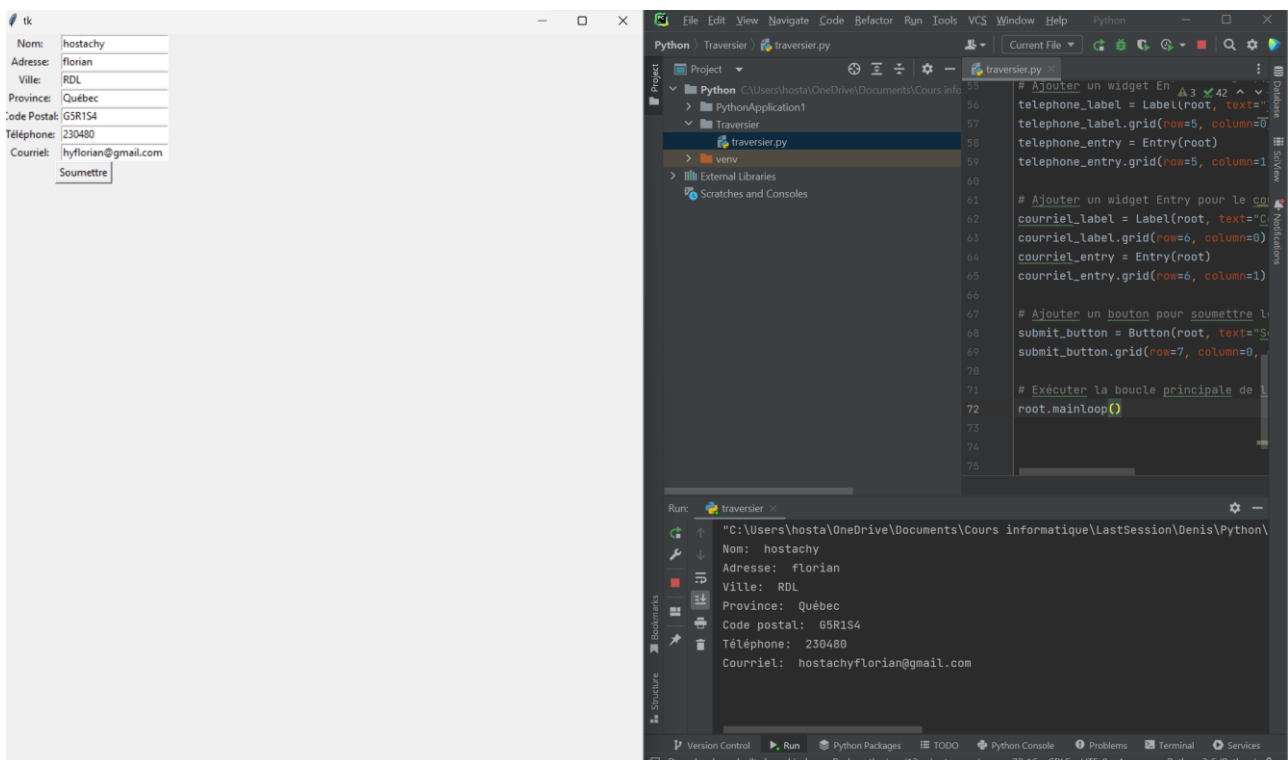
Jouez (y/n): n
Bye Bye
```


Je montre dans cet exemple que l'ordinateur peut aussi jouer avec nous en lui permettant de donner un chiffre aléatoirement et donc de faire avancer notre jeu.

Petite conclusion :

Il est très facile d'utiliser python pour des opération simple le langage est très vite assimilé la communauté est très présente et permet de trouver les ressources nécessaires pour réaliser ses projets et avoir une documentations très fournie.

Projet Traversier :



On peut voir ici pour que j'ai fait un formulaire que j'arrive à récupérer les informations une fois quelle sont soumise

Progression et avancement du projet :

Création des interfaces et mise en place des classes.

J'ai pu finir la totalité de mes interfaces en les corrigeant car le ne fonctionnais pas correctement maintenant elles sont à la bonne dimension j'ai commencé la partie backend je peux recevoir mes enregistrements dans une liste et ensuite les récupérer dans un fichier xml.

Conclusion

Annexe

- [https://fr.wikipedia.org/wiki/Python \(langage\)](https://fr.wikipedia.org/wiki/Python_(langage))
- <https://www.journaldunet.fr/web-tech/dictionnaire-du-webmastering/1445304-python-definition-et-utilisation-de-ce-langage-informatique/>
- [https://fr.wikipedia.org/wiki/C sharp](https://fr.wikipedia.org/wiki/C_sharp)
- <https://www.geeksforgeeks.org/difference-between-python-and-c-sharp/>
- <https://litslink.com/blog/csharp-vs-python-choosing-right-language-for-your-project#:~:text=In%20short%2C%20C%23%20and%20Python,and%20has%20vast%20standard%20libraries.>
- <https://www.youtube.com/live/psaDHhZ0cPs>