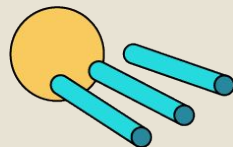


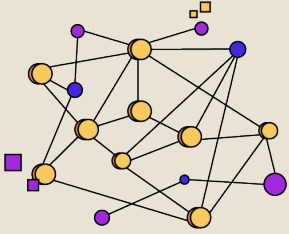
# **Présenter son projet de Data Science avec Streamlit et son portfolio avec GitHub**

**Pages**



DataScientest





# L'outil Streamlit



## Définition de Streamlit

- Streamlit est une librairie Python créée en 2018.
- Elle permet de créer des applications web interactives avec de nombreux widgets.
- C'est donc un outil très utile pour présenter les résultats d'un projet de Data Science (figures de DataVizualization, prédictions par modèles de Machine Learning)



<https://studio.datascientest.com/project/betpy/>

## Comparaison avec PowerBI

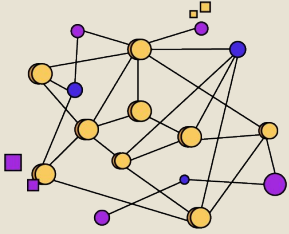
- **PowerBI** : outil de Business Intelligence permettant aux entreprises de prendre des décisions rapidement grâce à de la DataVizualization, des tableaux de bord et des rapports interactifs

*PowerBI possède son propre langage de programmation et sa propre interface graphique*

- **Streamlit** : outil permettant la présentation de projet de Data Science entier (pas uniquement la DataVizualization) de façon visuelle et résumée

*Streamlit est une librairie Python*

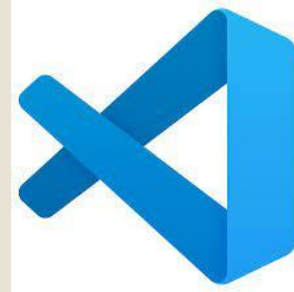




## Particularité de Streamlit



1. Utilisation d'un éditeur de code et pas d'un notebook :  
Spyder ou VisualStudioCode pour obtenir un fichier Python au format .py



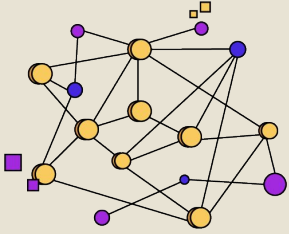
## 2. Utilisation des commandes sur le Terminal :

### Commandes principales :

- `cd` : permet de se placer dans le bon dossier de l'ordinateur (dossier contenant le fichier `.py`)
- `pip install streamlit` : permet d'installer la librairie streamlit
- `streamlit run` : permet de lancer l'application web Streamlit







# Création d'une application web Streamlit



# Projet à présenter avec Streamlit

- Nous nous plaçons dans un contexte immobilier. L'objectif est de prédire le prix d'un logement à partir de ses caractéristiques.
- Pour cela, nous avons accès à un fichier "housing.csv" comportant des données immobilières.
- Le projet a déjà traité dans le Data Atelier "Introduction au Machine Learning", disponible au lien suivant :  
<https://www.youtube.com/watch?v=6D3sKeoUhHo>

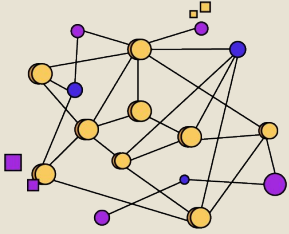


# Projet à présenter avec Streamlit

1. Etape 1 : Installation de Streamlit
2. Etape 2 : Création d'un fichier `streamlit_immobilier.py`
3. Etape 3 : Adaptation du code du Jupyter Notebook au fichier `streamlit_immobilier_prep.py`, en intégrant les commandes Streamlit
4. Etape 4 : Utilisation des commandes du Terminal pour obtenir l'application web Streamlit

Documentation Streamlit : <https://docs.streamlit.io/library/api-reference>





**Partager son application web  
Streamlit**



## Partager son application web

### Streamlit

- 1ère possibilité : continuer à travailler en local en envoyant le fichier .py à une personne qui maîtrise le Terminal
- 2ème possibilité : héberger l'application sur le Cloud Streamlit (<https://streamlit.io/cloud>)



# Hébergement sur le Cloud Streamlit

- Permet de partager son application web Streamlit (notamment avec des équipes qui ne maîtrisent pas le Terminal)
- Hébergement fait à partir d'un repo GitHub :
  1. création d'un repo GitHub pour le projet
  2. dépôt du fichier .py contenant le code Streamlit sur le repo
  3. création d'un compte sur le cloud Streamlit
  4. association du repo GitHub au compte Streamlit
  5. déploiement de l'application sur le Cloud Streamlit



Computer\_vision ▾

2 branches

0 tags

Go to file


Add file ▾

Code ▾

This branch is 4 commits ahead of main.

Pull request

Compare




8a69086 5 minutes ago

5 commits

README.md	Initial commit	3 hours ago
modele_0.h5	App_files_upload	3 hours ago
requirements.txt	Update requirements.txt	3 hours ago
streamlit_app_cv.py	App_files_upload	3 hours ago

README.md



## streamlit\_apps

Streamlit applications will be stored here

## Deploy an app

Apps are deployed directly from their GitHub repo. Enter the location of your app below.

Or [click here to fork and deploy a sample app](#).

Repository

[Paste GitHub URL](#)

Vous choisissez le nom de votre repository

Branch

Vous choisissez le nom de votre branche sur laquelle les fichiers se situent

Main file path

Vous écrivez le nom du fichier .py de votre application

Deploy!

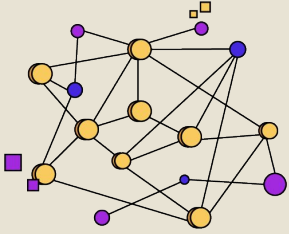


## Confidentialité pour l'hébergement

- Nombre illimité de Streamlit hébergés à partir de repo GitHub publics mais possibilité de partir d'un repo GitHub privé
- Gestion des données confidentielles avec les "Secrets Streamlit". Les parties sensibles des données/du code sont stockées dans un fichier sans être commit sur le repo GitHub. Elles sont ensuite accessibles sur Streamlit en tant que variables d'environnement en utilisant l'onglet "Paramètres avancés" lors du déploiement sur le cloud Streamlit et en utilisant la fonction `st.secrets()` dans le fichier `.py`







# L'outil GitHub Pages



## Définition de GitHub Pages

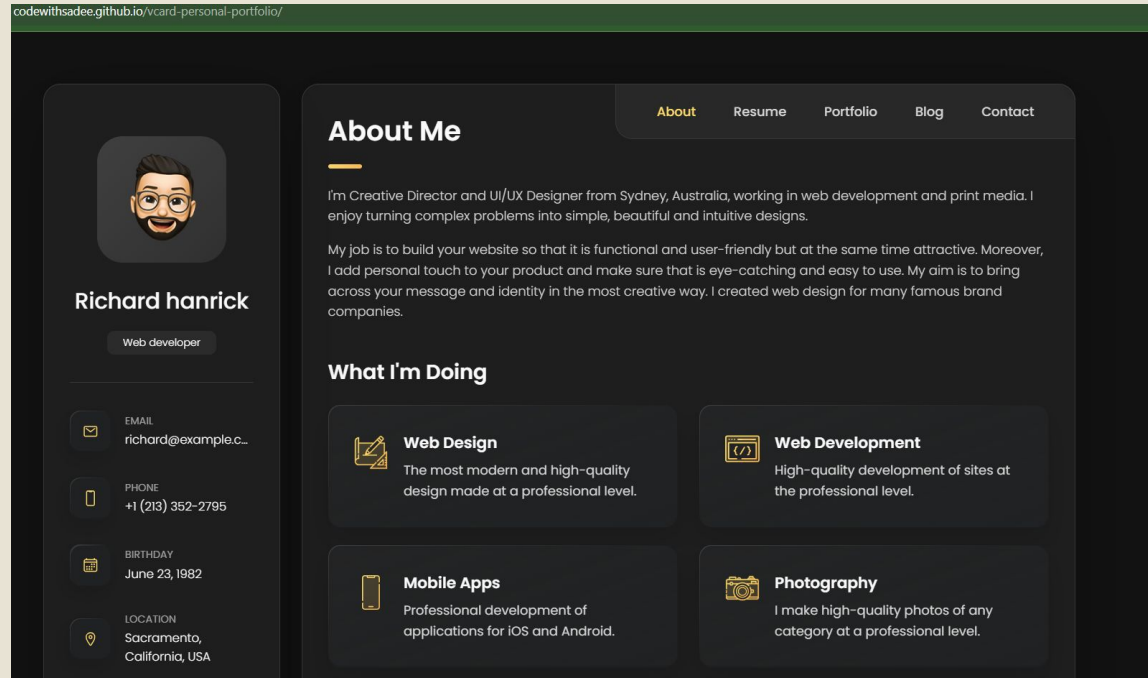
- GitHub Pages est un outil provenant de GitHub
- Il permet d'héberger un site internet à partir d'un repository GitHub
- Il est basé sur des fichiers HTML, CSS et JavaScript
- Il est très utilisé pour présenter son portfolio en regroupant tous ses projets Data sur un même site

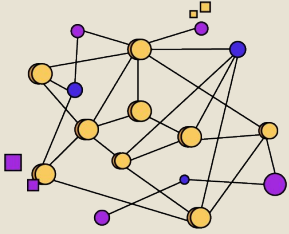


<https://docs.github.com/fr/pages/getting-started-with-github-pages/about-github-pages>

<https://codewithsadee.github.io/vcard-personal-portfolio/>

<https://github.com/codewithsadee/vcard-personal-portfolio>





# Création d'un site GitHub Pages



# Création d'un site GitHub Pages

1. création d'un repo GitHub vide dont le nom est `usernamegithub.github.io`
2. modification des Paramètres > Pages > Déployer à partir d'une branche
3. création automatique du site `usernamegithub.github.io`
4. remplissage du site à partir d'un fichier appelé `index.html` pushé sur le repo GitHub



## Remarques sur GitHub Pages

- Pas d'update instantané du site web (environ 5 mins)
- Limite de 1GB
- Site privé accessible uniquement par certaines personnes en disposant de la version GitHub Enterprise Cloud
- URL non-modifiable -> création d'un username GitHub adapté à la page web souhaitée si nécessaire

