[Punkyherisson/LearnGitGithub: Approfondir Git et Githubs](https://github.com/Punkyherisson/LearnGitGithub) (copilot)

J'utilise github pour mes projets personnels depuis un an et je maitrise français, anglais, allemand et espagnol. J'aimerais mieux maitriser github, github workspaces, les gist et toutes les possibilités de GIT. J'ai accés à Datacamp, RealPython, Linkedinlearning, Percipio et replit. Peux tu me donner un plan pour apprendre progressivement en utilisant ces ressources et ce qu'il y a de libre sur internet ?

Parfait, Emmanuel — tu as déjà une bonne base avec GitHub et un atout énorme : tu maîtrises quatre langues, ce qui te donne accès à un volume colossal de ressources.  
Je te propose un **plan d’apprentissage progressif sur 8 semaines**, qui combine :

* **Tes plateformes premium** (DataCamp, RealPython, LinkedIn Learning, Percipio, Replit)
* **Ressources libres** de qualité sur le web
* **Projets pratiques** pour ancrer les compétences

Bien sûr, Emmanuel !  
Voici la **version réécrite de ton plan d’apprentissage Git / GitHub / Gist / Workspaces** avec **tous les liens directs** vers les ressources, en combinant ce que tu as sur DataCamp, RealPython, LinkedIn Learning, Percipio, Replit et ce qui est libre sur internet.

**📅 Plan d’apprentissage progressif (8 semaines)**

**Semaine 1 — Consolider les bases de Git**

🎯 Objectif : maîtriser les commandes essentielles et le flux de travail local → distant.  
**Ressources**

* [Introduction à Git – DataCamp](https://www.datacamp.com/fr/courses/introduction-to-git)
* [Pro Git Book (FR)](https://git-scm.com/book/fr/v2) – livre gratuit officiel
* [Pro Git Book (EN)](https://git-scm.com/book/en/v2) – version anglaise
* *Git Essential Training* – [LinkedIn Learning](https://www.linkedin.com/learning/git-essential-training-the-basics) (EN)  
  **Pratique**
* Créer un repo local → commit → push vers GitHub
* Tester branch, merge, stash, rebase

**Semaine 2 — GitHub en profondeur**

🎯 Objectif : comprendre toutes les fonctionnalités de GitHub (issues, projects, actions, wiki).  
**Ressources**

* [Fondements de GitHub – DataCamp](https://www.datacamp.com/fr/tracks/github-foundations)
* [GitHub Docs – Guide de démarrage](https://docs.github.com/fr/get-started)
* *GitHub for Developers* – [LinkedIn Learning](https://www.linkedin.com/learning/github-for-developers)   
  **Pratique**
* Créer un repo avec README, wiki et tableau de projet
* Configurer une GitHub Action simple (tests Python)

**Semaine 3 — GitHub Workspaces / Codespaces**

🎯 Objectif : utiliser GitHub Codespaces comme IDE cloud.  
**Ressources**

* [Présentation de GitHub Codespaces](https://docs.github.com/fr/codespaces/overview)
* [Configurer un environnement Python dans Codespaces](https://docs.github.com/fr/codespaces/developing-in-codespaces/creating-a-codespace)
* Tutoriel vidéo : [GitHub Codespaces for Python Development](https://www.youtube.com/watch?v=ozuDPmcC1io) (EN)  
  **Pratique**
* Ouvrir un repo dans Codespaces
* Configurer requirements.txt
* Tester le développement collaboratif

**Semaine 4 — Gists et snippets**

🎯 Objectif : partager et réutiliser du code efficacement.  
**Ressources**

* [Documentation GitHub Gist](https://docs.github.com/fr/get-started/writing-on-github/editing-and-sharing-content-with-gists)
* Article RealPython : [How to Share Your Python Code With Gists](https://realpython.com/python-gists/) (EN)  
  **Pratique**
* Créer un gist public et un gist secret
* Intégrer un gist dans un README ou un site

**Semaine 5 — Git avancé**

🎯 Objectif : maîtriser les commandes et stratégies avancées.  
**Ressources**

* *Advanced Git* – [LinkedIn Learning](https://www.linkedin.com/learning/advanced-git)
* Chapitres avancés du [Pro Git Book](https://git-scm.com/book/en/v2)
* RealPython : [Python and Git Hooks](https://realpython.com/python-git-hooks/)   
  **Pratique**
* Rebase interactif (git rebase -i)
* Cherry-pick et bisect
* Hooks Git pour automatiser des tâches

**Semaine 6 — Collaboration et workflow d’équipe**

🎯 Objectif : travailler comme dans un vrai projet open source.  
**Ressources**

* *Collaborating with GitHub* – [LinkedIn Learning](https://www.linkedin.com/learning/collaborating-with-github)
* [GitHub Flow](https://docs.github.com/fr/get-started/quickstart/github-flow)   
  **Pratique**
* Forker un projet open source
* Créer une branche, faire un pull request, gérer un code review

**Semaine 7 — Automatisation et intégration continue**

🎯 Objectif : automatiser tests, déploiements et intégrations.  
**Ressources**

* [GitHub Actions – Documentation](https://docs.github.com/fr/actions)
* RealPython : [Continuous Integration With GitHub Actions](https://realpython.com/python-github-actions-ci-cd/)   
  **Pratique**
* Créer un workflow GitHub Actions pour exécuter des tests Python
* Déployer automatiquement sur Replit ou autre hébergeur

**Semaine 8 — Projet final**

🎯 Objectif : mettre en pratique tout ce que tu as appris.  
**Projet proposé**

* Application Python (API Flask ou FastAPI)
* Repo GitHub avec :
  + README complet
  + Wiki
  + GitHub Actions
  + Codespaces configuré
  + Gists liés pour des snippets réutilisables
* Collaboration avec au moins une autre personne

💡 **Bonus** :

* [Replit Docs](https://docs.replit.com/) pour intégrer GitHub et coder en ligne
* [Git Cheat Sheet (FR)](https://training.github.com/downloads/fr/github-git-cheat-sheet/) pour les commandes rapides

Si tu veux, je peux maintenant te **transformer ce plan en tableau interactif** avec cases à cocher et liens cliquables pour suivre ta progression semaine par semaine.  
Veux-tu que je te prépare ce tableau ?

Parfait, Emmanuel !  
Voici ton **tableau de suivi interactif** pour progresser sur **Git, GitHub, Gists, Workspaces** et toutes les fonctionnalités avancées, avec **liens directs** vers les ressources.  
Je t’ajoute aussi une section **Ressources supplémentaires** pour aller encore plus loin.

**📊 Tableau de progression**

| **Semaine** | **Objectif** | **Ressources principales** | **Pratique recommandée** | **Statut ✅** |
| --- | --- | --- | --- | --- |
| **1** | Bases solides de Git | [Introduction à Git – DataCamp](https://www.datacamp.com/fr/courses/introduction-to-git) • [Pro Git Book (FR)](https://git-scm.com/book/fr/v2) • [Pro Git Book (EN)](https://git-scm.com/book/en/v2) • [Git Essential Training – LinkedIn Learning](https://www.linkedin.com/learning/git-essential-training-the-basics) | Créer un repo local → commit → push vers GitHub ; tester branch, merge, stash, rebase | ⬜ |
| **2** | GitHub en profondeur | [Fondements de GitHub – DataCamp](https://www.datacamp.com/fr/tracks/github-foundations) • [GitHub Docs – Guide de démarrage](https://docs.github.com/fr/get-started) • [GitHub for Developers – LinkedIn Learning](https://www.linkedin.com/learning/github-for-developers) | Créer un repo avec README, wiki, tableau de projet ; configurer une GitHub Action simple | ⬜ |
| **3** | GitHub Workspaces / Codespaces | [Présentation Codespaces](https://docs.github.com/fr/codespaces/overview) • [Configurer un projet Python dans Codespaces](https://docs.github.com/en/codespaces/setting-up-your-project-for-codespaces/adding-a-dev-container-configuration/setting-up-your-python-project-for-codespaces) • [Tutoriel vidéo Codespaces Python](https://www.youtube.com/watch?v=ozuDPmcC1io) | Ouvrir un repo dans Codespaces, configurer requirements.txt, tester le dev collaboratif | ⬜ |
| **4** | Gists et snippets | [Docs GitHub Gist](https://docs.github.com/fr/get-started/writing-on-github/editing-and-sharing-content-with-gists) • [RealPython – Python Gists](https://realpython.com/python-gists/) | Créer un gist public et un gist secret ; intégrer un gist dans un README ou site | ⬜ |
| **5** | Git avancé | [Advanced Git – LinkedIn Learning](https://www.linkedin.com/learning/advanced-git) • [Pro Git Book – Chapitres avancés](https://git-scm.com/book/en/v2) • [RealPython – Git Hooks](https://realpython.com/python-git-hooks/) | Rebase interactif, cherry-pick, bisect, hooks Git | ⬜ |
| **6** | Collaboration & workflow d’équipe | [Collaborating with GitHub – LinkedIn Learning](https://www.linkedin.com/learning/collaborating-with-github) • [GitHub Flow](https://docs.github.com/fr/get-started/quickstart/github-flow) | Forker un projet open source, créer une branche, pull request, code review | ⬜ |
| **7** | Automatisation & CI/CD | [GitHub Actions Docs](https://docs.github.com/fr/actions) • [RealPython – CI/CD avec GitHub Actions](https://realpython.com/python-github-actions-ci-cd/) | Créer un workflow pour tests Python ; déploiement auto sur Replit ou autre | ⬜ |
| **8** | Projet final complet | — | Application Python (Flask/FastAPI) avec README, wiki, Actions, Codespaces, Gists, collab | ⬜ |

**📚 Ressources supplémentaires**

**Guides et docs**

* [Git Cheat Sheet (FR)](https://training.github.com/downloads/fr/github-git-cheat-sheet/) – résumé visuel des commandes Git
* [GitHub Learning Lab](https://lab.github.com/) – parcours interactifs gratuits
* [Oh My Git!](https://ohmygit.org/) – jeu interactif pour apprendre Git
* [Atlassian Git Tutorials](https://www.atlassian.com/git/tutorials) – explications claires sur Git et workflows

**Outils et bibliothèques utiles**

* [gistyc (PyPI)](https://pypi.org/project/gistyc/) – bibliothèque Python pour créer/mettre à jour des Gists automatiquement
* [GitKraken](https://www.gitkraken.com/) – client Git visuel
* [Sourcetree](https://www.sourcetreeapp.com/) – gestion Git en interface graphique

**Pratique collaborative**

* [Initiation à GitHub Projects – Kanban](https://techcommunity.microsoft.com/blog/educatordeveloperblog/initiation-%c3%a0-github-projects---partie-i-tableau-kanban/3784393) – organiser ton travail avec GitHub Projects
* [Awesome GitHub](https://github.com/phillipadsmith/awesome-github) – liste de ressources et outils GitHub

💡 **Astuce** : Tu peux importer ce tableau dans un Google Sheet ou Notion pour cocher les cases au fur et à mesure, et même y ajouter tes propres notes ou liens.

Si tu veux, je peux aussi te préparer **un tableau Kanban GitHub Projects** déjà configuré avec ces étapes, pour que tu puisses suivre ta progression directement dans GitHub et lier chaque étape à un repo ou un gist.  
Veux-tu que je te prépare ce Kanban ?

Parfait, Emmanuel !  
Voici comment tu peux mettre en place **ton tableau Kanban GitHub Projects** pour suivre ton plan d’apprentissage Git/GitHub/Gists/Workspaces directement dans GitHub, avec un modèle prêt à l’emploi et des ressources pour le configurer.

**🗂 Création de ton Kanban GitHub Projects**

**1 — Créer le projet**

1. Connecte-toi à GitHub.
2. Depuis ta page d’accueil ou un dépôt dédié à ton apprentissage, clique sur **Projects** → **New project**.
3. Choisis le modèle **Board (Kanban)**.
4. Donne-lui un nom, par exemple : Plan d'apprentissage Git & GitHub.

**2 — Colonnes recommandées**

* **À faire** (To Do) → toutes les étapes non commencées
* **En cours** (In Progress) → ce que tu travailles actuellement
* **En révision** (Review) → si tu veux valider ou approfondir
* **Terminé** (Done) → ce que tu as acquis

**3 — Cartes à créer**

Pour chaque semaine de ton plan, crée une carte avec :

* **Titre** : Semaine X — Objectif
* **Description** : résumé + liens vers les ressources
* **Checklist** : sous-tâches pratiques à réaliser
* **Labels** : par exemple Git, GitHub, Gist, Codespaces, CI/CD

**4 — Exemple de carte**

**Titre** : Semaine 1 — Bases de Git   
**Description** :

Objectif : maîtriser les commandes essentielles et le flux local → distant.  
Ressources :

* [Introduction à Git – DataCamp](https://www.datacamp.com/fr/courses/introduction-to-git)
* [Pro Git Book (FR)](https://git-scm.com/book/fr/v2)
* [Git Essential Training – LinkedIn Learning](https://www.linkedin.com/learning/git-essential-training-the-basics)

**Checklist** :

* [ ] Créer un repo local
* [ ] Faire un commit
* [ ] Pousser vers GitHub
* [ ] Tester branch, merge, stash, rebase

**5 — Ressources pour configurer ton Kanban**

* [Initiation à GitHub Projects – Partie I (Tableau Kanban)](https://techcommunity.microsoft.com/blog/educatordeveloperblog/initiation-%c3%a0-github-projects---partie-i-tableau-kanban/3784393)
* [Gestion de projet avec Kanban sur GitHub – Cursa](https://cursa.app/fr/page/gestion-de-projet-avec-kanban-sur-github)

Ces guides expliquent pas à pas comment créer et personnaliser un tableau Kanban dans GitHub Projects, ajouter des cartes, des étiquettes, et lier des issues ou pull requests.

**6 — Astuce bonus**

Tu peux **lier ce Kanban à un dépôt vide** juste pour ton apprentissage.  
Chaque carte peut contenir :

* Les liens vers les cours (DataCamp, RealPython, LinkedIn Learning, etc.)
* Les gists que tu crées pendant ta formation
* Les repos de test sur lesquels tu pratiques

Si tu veux, je peux te préparer **un fichier .csv prêt à importer dans GitHub Projects** avec toutes les cartes déjà remplies pour tes 8 semaines, afin que ton tableau soit opérationnel en 2 minutes.  
Veux-tu que je te prépare ce fichier ?

Parfait, Emmanuel !  
Voici ton **fichier CSV prêt à importer dans GitHub Projects** pour créer automatiquement ton tableau Kanban avec toutes les cartes correspondant à ton plan d’apprentissage Git/GitHub/Gists/Workspaces.

**📄 Contenu du CSV**

Tu peux copier-coller ce texte dans un fichier plan\_apprentissage\_git.csv (UTF‑8), puis l’importer dans un **nouveau projet GitHub** via l’option **Import CSV**.

Title,Description,Status,Labels

Semaine 1 — Bases de Git,"🎯 Objectif : Maîtriser les commandes essentielles et le flux local → distant.\n\n📚 Ressources :\n- Introduction à Git – DataCamp : https://www.datacamp.com/fr/courses/introduction-to-git\n- Pro Git Book (FR) : https://git-scm.com/book/fr/v2\n- Pro Git Book (EN) : https://git-scm.com/book/en/v2\n- Git Essential Training – LinkedIn Learning : https://www.linkedin.com/learning/git-essential-training-the-basics\n\n✅ Pratique :\n- Créer un repo local\n- Commit\n- Push vers GitHub\n- Tester branch, merge, stash, rebase",To Do,"Git,GitHub"

Semaine 2 — GitHub en profondeur,"🎯 Objectif : Comprendre toutes les fonctionnalités de GitHub (issues, projects, actions, wiki).\n\n📚 Ressources :\n- Fondements de GitHub – DataCamp : https://www.datacamp.com/fr/tracks/github-foundations\n- GitHub Docs – Guide de démarrage : https://docs.github.com/fr/get-started\n- GitHub for Developers – LinkedIn Learning : https://www.linkedin.com/learning/github-for-developers\n\n✅ Pratique :\n- Créer un repo avec README, wiki, tableau de projet\n- Configurer une GitHub Action simple",To Do,"GitHub,Actions"

Semaine 3 — GitHub Workspaces / Codespaces,"🎯 Objectif : Utiliser GitHub Codespaces comme IDE cloud.\n\n📚 Ressources :\n- Présentation Codespaces : https://docs.github.com/fr/codespaces/overview\n- Configurer un projet Python dans Codespaces : https://docs.github.com/en/codespaces/setting-up-your-project-for-codespaces/adding-a-dev-container-configuration/setting-up-your-python-project-for-codespaces\n- Tutoriel vidéo Codespaces Python : https://www.youtube.com/watch?v=ozuDPmcC1io\n\n✅ Pratique :\n- Ouvrir un repo dans Codespaces\n- Configurer requirements.txt\n- Tester le dev collaboratif",To Do,"Codespaces,Python"

Semaine 4 — Gists et snippets,"🎯 Objectif : Partager et réutiliser du code efficacement.\n\n📚 Ressources :\n- Docs GitHub Gist : https://docs.github.com/fr/get-started/writing-on-github/editing-and-sharing-content-with-gists\n- RealPython – Python Gists : https://realpython.com/python-gists/\n\n✅ Pratique :\n- Créer un gist public et un gist secret\n- Intégrer un gist dans un README ou site",To Do,"Gist,Snippets"

Semaine 5 — Git avancé,"🎯 Objectif : Maîtriser les commandes et stratégies avancées.\n\n📚 Ressources :\n- Advanced Git – LinkedIn Learning : https://www.linkedin.com/learning/advanced-git\n- Pro Git Book – Chapitres avancés : https://git-scm.com/book/en/v2\n- RealPython – Git Hooks : https://realpython.com/python-git-hooks/\n\n✅ Pratique :\n- Rebase interactif\n- Cherry-pick\n- Bisect\n- Hooks Git",To Do,"Git,Advanced"

Semaine 6 — Collaboration & workflow d’équipe,"🎯 Objectif : Travailler comme dans un vrai projet open source.\n\n📚 Ressources :\n- Collaborating with GitHub – LinkedIn Learning : https://www.linkedin.com/learning/collaborating-with-github\n- GitHub Flow : https://docs.github.com/fr/get-started/quickstart/github-flow\n\n✅ Pratique :\n- Forker un projet open source\n- Créer une branche\n- Pull request\n- Code review",To Do,"Collaboration,Workflow"

Semaine 7 — Automatisation & CI/CD,"🎯 Objectif : Automatiser tests, déploiements et intégrations.\n\n📚 Ressources :\n- GitHub Actions Docs : https://docs.github.com/fr/actions\n- RealPython – CI/CD avec GitHub Actions : https://realpython.com/python-github-actions-ci-cd/\n\n✅ Pratique :\n- Créer un workflow pour tests Python\n- Déploiement auto sur Replit ou autre",To Do,"CI/CD,Automation"

Semaine 8 — Projet final complet,"🎯 Objectif : Mettre en pratique tout ce que tu as appris.\n\n✅ Projet proposé :\n- Application Python (Flask/FastAPI)\n- README complet\n- Wiki\n- GitHub Actions\n- Codespaces configuré\n- Gists liés\n- Collaboration avec au moins une autre personne",To Do,"Projet,Final"

**📌 Comment l’importer dans GitHub Projects**

1. Crée un **nouveau projet** dans GitHub (type **Board**).
2. Clique sur **…** (menu) → **Import CSV**.
3. Sélectionne ton fichier plan\_apprentissage\_git.csv.
4. Les cartes seront créées automatiquement dans la colonne **To Do**.
5. Tu pourras ensuite déplacer les cartes dans **In Progress** et **Done** au fur et à mesure.

**💡 Astuce**

Tu peux enrichir chaque carte avec :

* Des **liens vers tes dépôts** ou gists créés pendant la formation
* Des **checklists détaillées**
* Des **dates d’échéance** pour te motiver

Si tu veux, je peux aussi te préparer **une version améliorée de ce CSV** avec directement les **checklists intégrées** dans chaque carte, pour que tu puisses cocher les sous-tâches dans GitHub Projects.  
Veux-tu que je te prépare cette version ?