**💻 Engaging GitHub Learning Task - Hands-on Challenge! 🎯🚀**

This challenge will give students **practical experience** with GitHub while making the session more **interactive and engaging**! They will work on real-world scenarios while watching the videos.

**🎯 Task Overview: Build Your First GitHub Portfolio Project!**

🔹 **Time Required:** 2-3 Hours  
🔹 **Objective:** Students will create and manage a **GitHub repository** where they will store their first **mini-project** using version control principles.  
🔹 **Deliverables:** A **GitHub repository link** with multiple commits, branches, and proper documentation.

**📌 Part 1: Set Up a GitHub Repository (30 min)**

📺 **Watch:** [How to Create a GitHub Repository](https://www.youtube.com/watch?v=RGOj5yH7evk)  
🎯 **Challenge:**

1. Create a **GitHub repository** named **"MyPortfolio"**.
2. Initialize it with a **README.md** and write a short bio.
3. Commit the changes and push them to GitHub.

✅ **Check:** Your repository should now have:

* A README file ✅
* A commit history ✅

**📌 Part 2: Fork & Clone a Repository (30 min)**

📺 **Watch:** [Git Fork vs Clone Explained](https://www.youtube.com/watch?v=lQwsLgrkCzM)  
🎯 **Challenge:**

1. Fork **this public repository**: [Example Repo](https://github.com/octocat/Spoon-Knife).
2. Clone the forked repository to your local machine.
3. Add a new file (yourname.txt), write something about yourself, and commit it.
4. Push the changes back to GitHub.

✅ **Check:** Your repository should now have:

* A new file you added ✅
* A commit log showing your update ✅

**📌 Part 3: Work with Commits & Version Control (45 min)**

📺 **Watch:** [Git Commit and Push Changes](https://www.youtube.com/watch?v=USjZcfj8yxE)  
🎯 **Challenge:**

1. Create a new **branch** called feature-update.
2. Modify your **README.md** by adding a section called “My Skills.”
3. Make **three separate commits** with messages:
   * "Added skills section"
   * "Updated skills list"
   * "Formatted README"
4. Merge the feature-update branch back into main.
5. Delete the feature-update branch after merging.

✅ **Check:** Your repository should now have:

* A **feature-update branch** (merged into main) ✅
* A clear **commit history** with meaningful messages ✅

**📌 Part 4: Push & Pull Requests (30 min)**

📺 **Watch:** [How to Contribute to Open Source](https://www.youtube.com/watch?v=MnIpOrzz5Ck)  
🎯 **Challenge:**

1. Find a friend or classmate and **fork each other's repositories**.
2. Make a small change in their README file (like adding a greeting).
3. **Create a Pull Request (PR)** to suggest the change.
4. Review and merge the PR.

✅ **Check:** Your repository should now have:

* A merged **Pull Request** from a classmate ✅

**📌 Part 5: Fetch vs Pull (30 min)**

📺 **Watch:** [Git Fetch vs Git Pull](https://www.youtube.com/watch?v=3aM8SUGF_mw)  
🎯 **Challenge:**

1. Open your **GitHub repository** in a **local terminal**.
2. Run git fetch and check if any remote updates exist.
3. Run git pull to sync your local repo with GitHub.
4. Modify a file on GitHub directly and **use fetch/pull to update it locally**.

✅ **Check:**

* You successfully **used fetch & pull** ✅

**💡 Final Submission (30 min)**

🎯 **What to Submit?**  
✅ GitHub repository link with:

* Multiple commits & branches
* README file with sections
* Forked repo & pull requests
* Fetch & pull usage