



You:

I am currently developing an audiobook app for android and ios. I am currently a beginner and I am not very versed with using git and git hub but I want to have a way to store the code to git so I can track the progress, and I also want to make a barebones page for it where I can write blogs on its development, news, updates put screenshots and just showcase the work that im doing. How do I do this ?help me I have no experience



ChatGPT:

I can guide you through the process of getting started with Git, GitHub, and creating a simple webpage for your project. Here's a step-by-step guide:

1. Setting Up Git and GitHub

a. Install Git

- Download and install Git from git-scm.com.
- Follow the installation instructions for your operating system.

b. Create a GitHub Account

- Sign up for a free account at [GitHub](https://github.com).

c. Create a New Repository on GitHub

- Once logged in, click on the "+" icon in the upper right corner and select "New repository."
- Name your repository (e.g., audiobook-app).
- Choose whether to make it public or private.
- Initialize the repository with a README file (optional but recommended).

d. Link Your Local Project to GitHub

1. Open your project folder in your terminal or command prompt.
2. Initialize a Git repository by running:

```
bash
git init
```

3. Add the remote repository you created on GitHub:

```
bash
git remote add origin https://github.com/your-username/audiobook-app.git
```

4. Add your project files to the staging area:

```
bash
git add .
```

5. Commit your changes:

```
bash
git commit -m "Initial commit"
```

6. Push your code to GitHub:

```
bash
git push -u origin master
```

2. Creating a Simple Webpage

a. Using GitHub Pages

- GitHub Pages is a free service that allows you to host a simple static website directly from a GitHub repository.

1. Create a New Repository for Your Website

- Name it `your-username.github.io`.
- Make sure it's public.
- This will serve as your personal website repository.

2. Create a Simple Webpage

- You can start by creating an `index.html` file.
- Use basic HTML to structure your page. Here's a template to get you started:

```
html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Audiobook App</title>
</head>
<body>
  <h1>Welcome to My Audiobook App Development Blog</h1>
  <p>Here you'll find updates, screenshots, and more about the development process.</p>
</body>
</html>
```

3. Deploy Your Website

- Add the `index.html` file to your repository.
- Commit and push it to GitHub.
- Your site will be live at `https://your-username.github.io`.

b. Adding Blog Posts

- To add blogs, you can either manually update the HTML or use a static site generator like [Jekyll](#) (which GitHub Pages supports).
- For simplicity, you can create separate HTML files for each blog post and link them from your `index.html`.

3. Tracking Progress and Adding Screenshots

- For tracking progress, commit and push your changes to GitHub regularly.
- You can add screenshots by uploading them to your repository and referencing them in your HTML files or markdown files (if using a static site generator).

4. Learning Resources

- **Git and GitHub:** Try out resources like [GitHub's Learning Lab](#) for hands-on learning.
- **HTML & CSS:** Websites like [W3Schools](#) or [MDN Web Docs](#) are great for learning web development basics.

This will get you started with version control and creating a basic online presence for your project! Let me know if you need more detailed help with any of these steps.



You:

how do i check if my code was pushed to github?



To check if your code was successfully pushed to GitHub, follow these steps:

1. Check in Your Terminal/Command Prompt

- After running the `git push` command, you should see a confirmation message in your terminal. It usually includes something like:

```
bash
Enumerating objects: X, done.
Counting objects: X, done.
Delta compression using up to X threads.
Compressing objects: 100% (X/X), done.
Writing objects: 100% (X/X), X KiB | X MiB/s, done.
Total X (delta X), reused X (delta X), pack-reused X
To https://github.com/your-username/audiobook-app.git
* [new branch]      master -> master
```

2. Check on GitHub

• Visit Your Repository:

1. Go to [GitHub](#) and log in if you haven't already.
2. Navigate to your repository by clicking on your profile icon and selecting "Your repositories."
3. Click on the repository you pushed the code to (e.g., audiobook-app).
4. You should see your files listed in the repository.

• Check Commit History:

1. In your repository on GitHub, click on the "Commits" link (usually found near the top of the page).
2. This will show you a history of all commits. Your latest commit should be listed at the top, with the commit message you used (e.g., "Initial commit").

3. Check Branches

- If you pushed to a specific branch (e.g., master or main), ensure you're viewing the correct branch on GitHub. You can switch branches by clicking on the branch selector dropdown on your repository page.

4. Check File Contents

- You can click on any file in the GitHub repository to view its contents and confirm that your changes were uploaded.

If everything looks good in your GitHub repository, your code was successfully pushed!



What does the message below mean? Username for 'https://github.com': DaveDevz Password for 'https://DaveDevz@github.com': remote: Support for password authentication was removed on August 13, 2021. remote: Please see <https://docs.github.com/get-started/getting-started-with-git/about-remote-repositories#cloning-with-https-urls> for information on currently recommended modes of authentication. fatal: Authentication failed for 'https://github.com/DaveDevz/audiobook-app.git' andraedavid@Andraes-Air 01.UnheardApp %



The message you're seeing indicates that GitHub no longer supports password authentication for Git operations via HTTPS as of August 13, 2021. This means you cannot use your GitHub account password directly when performing Git operations like pushing or

pulling code. Instead, GitHub now requires more secure authentication methods.

How to Resolve This Issue:

1. Use a Personal Access Token (PAT)

- Instead of using your password, GitHub now requires you to use a Personal Access Token (PAT) for authentication.

Steps to Create a PAT:

1. **Go to GitHub:** Log in to your GitHub account and go to [Settings](#).
2. **Navigate to Developer Settings:** Scroll down and click on "Developer settings" on the left-hand sidebar.
3. **Create a Token:**
 - Click on "Personal access tokens" and then "Tokens (classic)".
 - Click "Generate new token."
 - Select the scopes or permissions you want to grant this token. At a minimum, you'll need repo access for your repositories.
 - Click "Generate token" at the bottom of the page.
 - Copy the generated token. (**Note: This is the only time you will see it, so be sure to save it securely.**)

Use the PAT Instead of Your Password:

- When prompted for a username and password during a Git operation:
 - Enter your GitHub username as usual.
 - Instead of your password, paste the PAT you just generated.

2. Set Up SSH Authentication (Optional)

- Alternatively, you can use SSH keys to authenticate with GitHub, which is another secure method and avoids the need to enter a token each time.

Setting Up SSH Authentication:

1. **Generate an SSH Key** (if you don't have one already):

```
bash
ssh-keygen -t ed25519 -C "your_email@example.com"
```

2. **Add the SSH Key to GitHub:**
 - Copy the content of your public SSH key (~/.ssh/id_ed25519.pub).
 - Go to GitHub and navigate to Settings > SSH and GPG keys.
 - Click "New SSH key," give it a title, and paste your key.
3. **Use the SSH URL for Your Repository:**
 - Replace the HTTPS URL with the SSH URL:

```
bash
git remote set-url origin git@github.com:DaveDevz/audiobook-app.git
```

Summary

To resolve the issue:

- Use a Personal Access Token (PAT) instead of your password when prompted.
- Alternatively, you can set up SSH authentication for a more seamless experience.

Let me know if you need more help with any of these steps!
