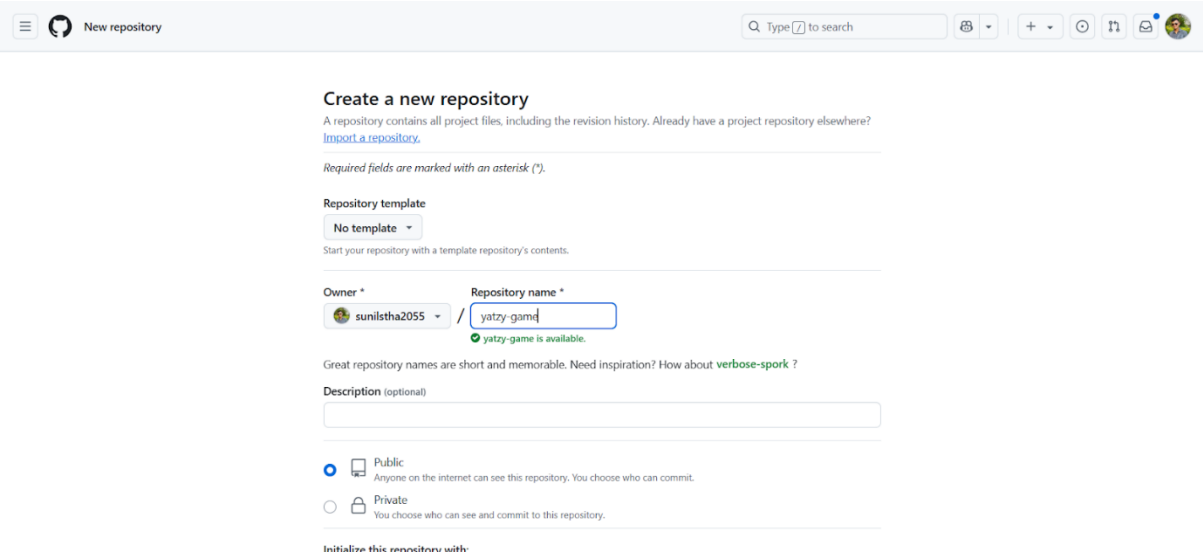
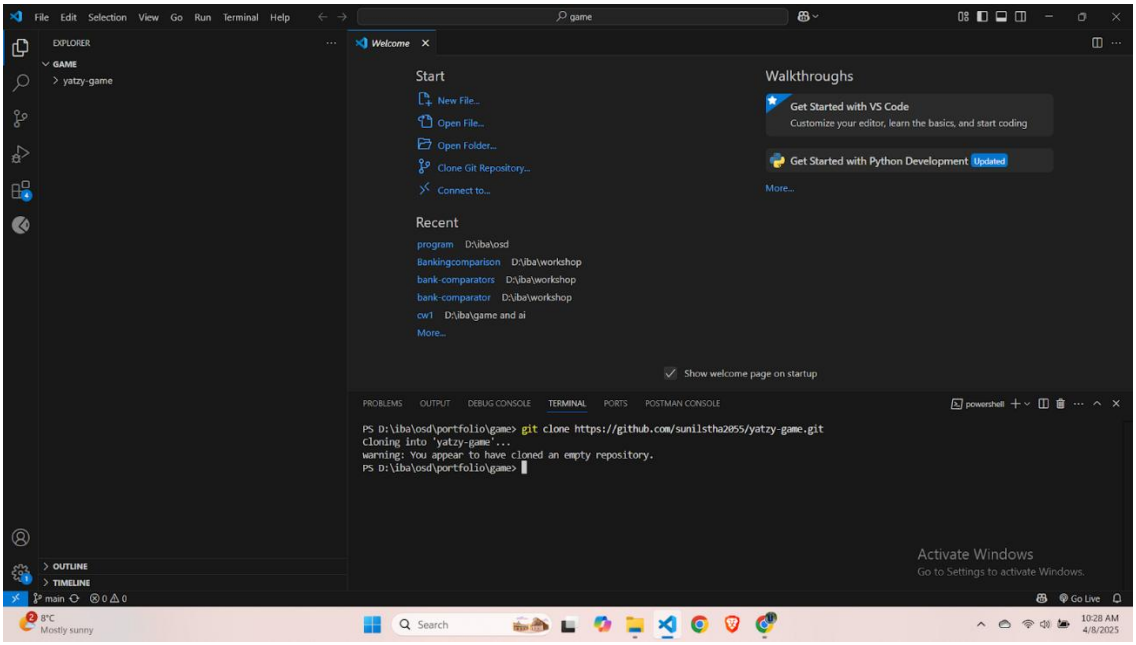
Git and GitHub Workflow Documentation – Yatzy Game

Create a Repository on GitHub



Clone the Repository

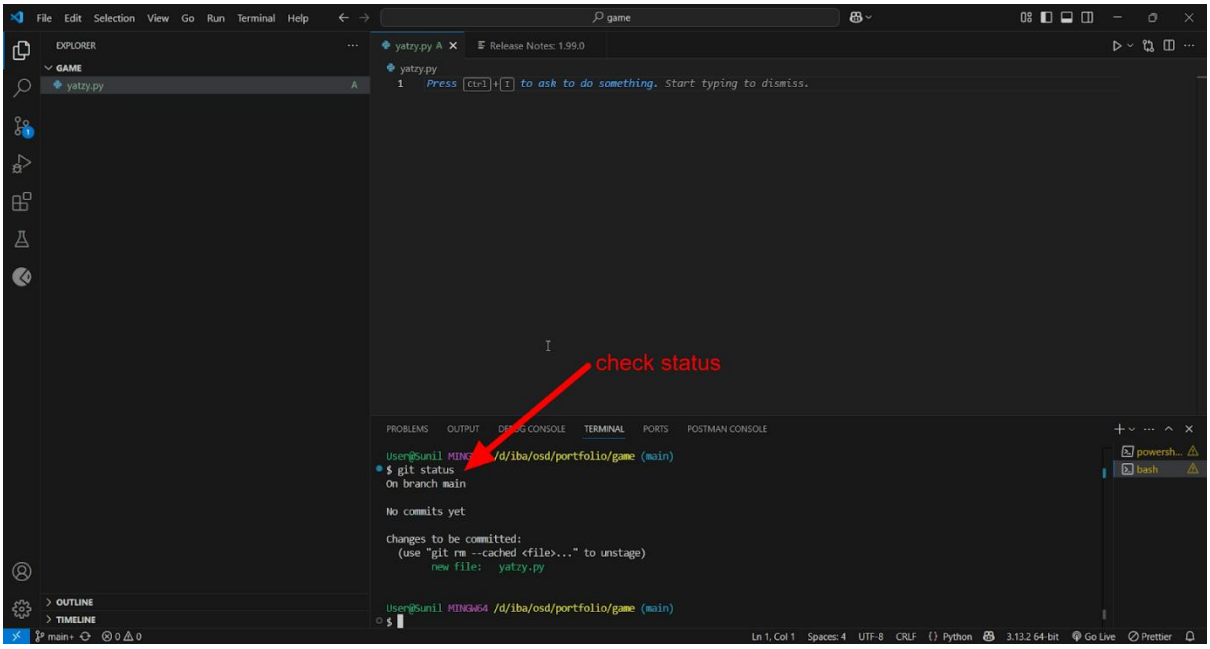
Git Clone is the command we use to create a copy of a remote repository (like one on GitHub or GitLab) on our local machine. It lets us download the entire project, including its history, to work on it locally. When we run git clone, it fetches the project's files from the remote repository and sets up a new local repository on our machine. This way, we have a complete copy of the project to work with, and we can make changes, commit them, and push them back to the remote repository.



Check Status

git status

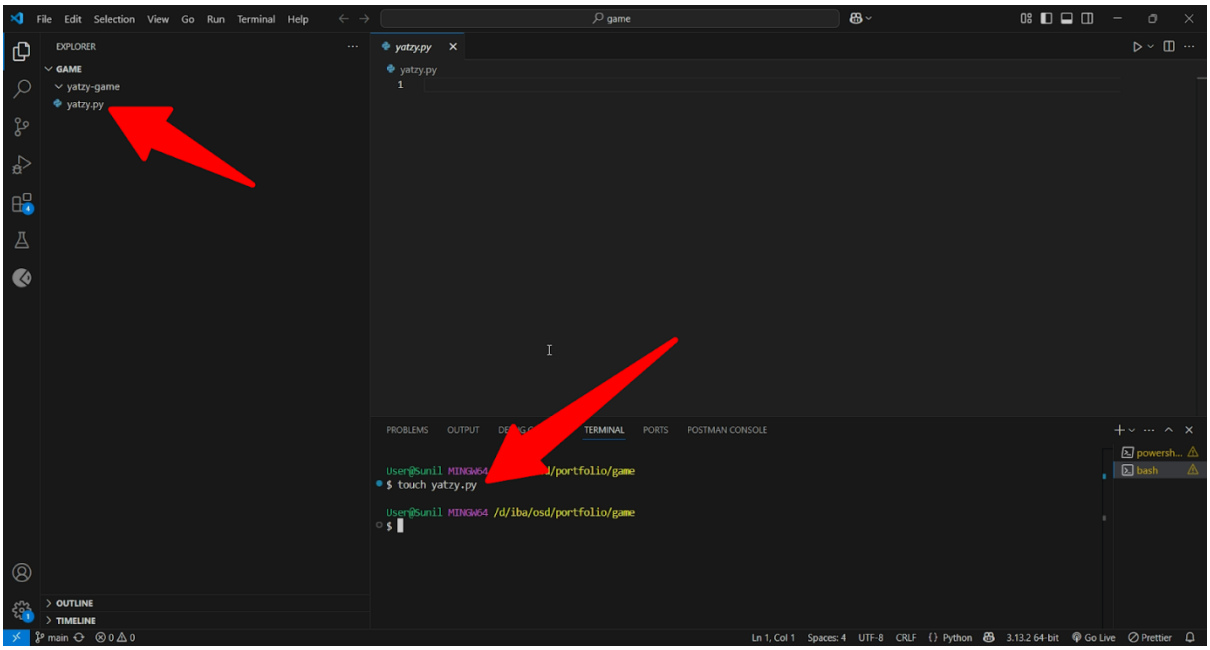
The git status command checks the current state of your Git working directory and staging area. It shows which files have been modified, added, or deleted since your last commit, and whether they're staged for the next commit or untracked (not yet added to Git).



Create Python File

touch yatzy.py

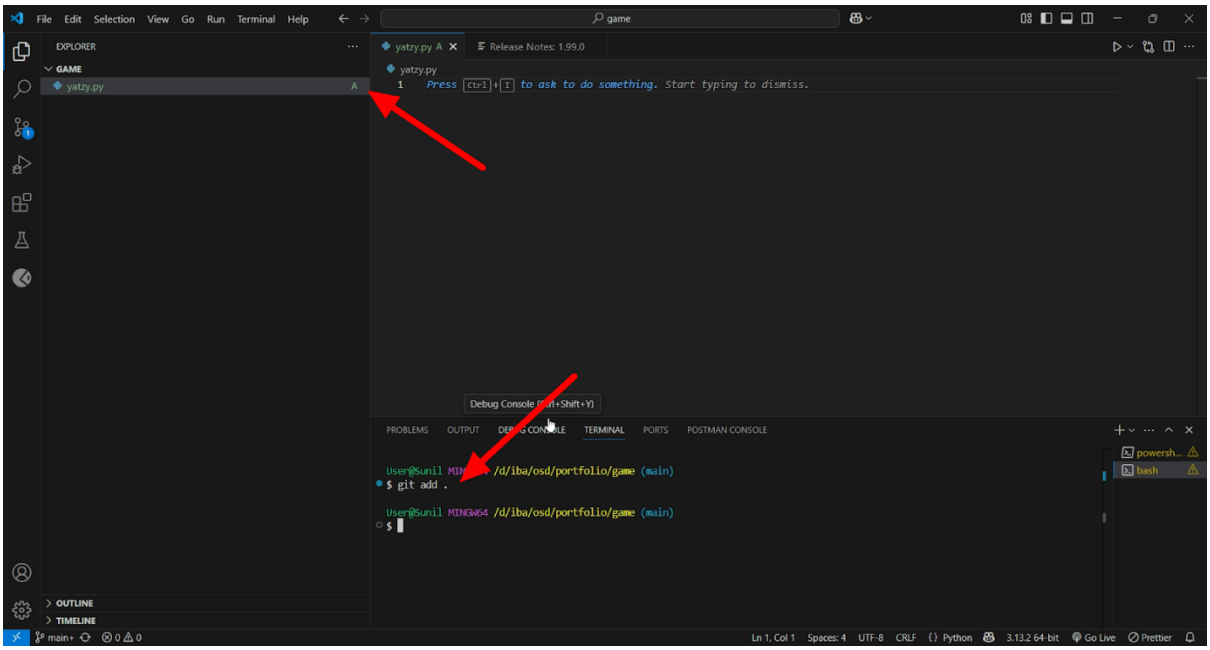
The touch yatzy.py command creates a new, empty file named yatzy.py in your current directory (assuming you're using a Unix-like system such as Linux or macOS). In the context of Worksheet 2, this is the initial step to create the Python file where you'll implement the Yatzy class with its dice and scoring methods.



Stage Changes

git add .

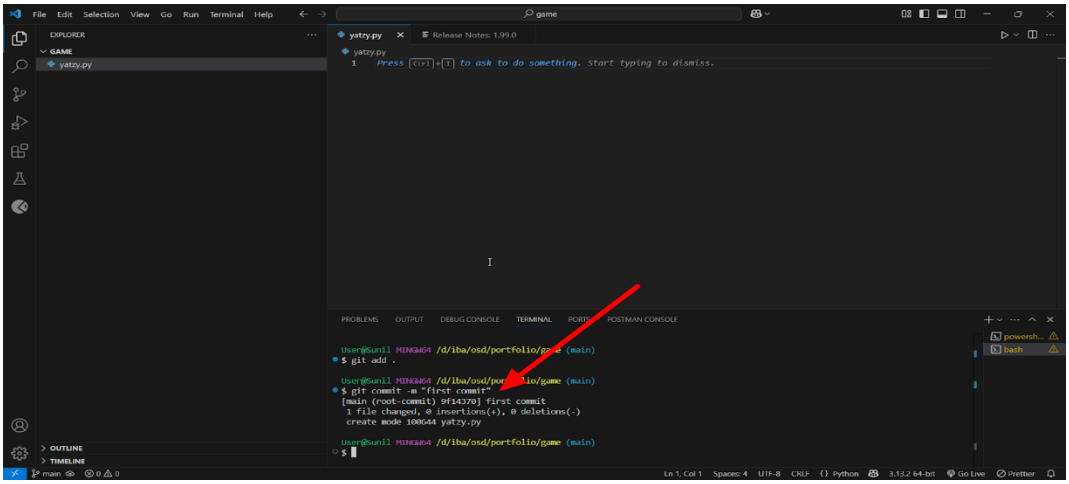
Git add is a command we use to tell Git which changes we want to save in your project. When we make changes to files in your project, we use to select the specific files or modifications we want to keep. Once we've added them, they are staged and ready to be saved.



Commit Changes

git commit -m "first commit"

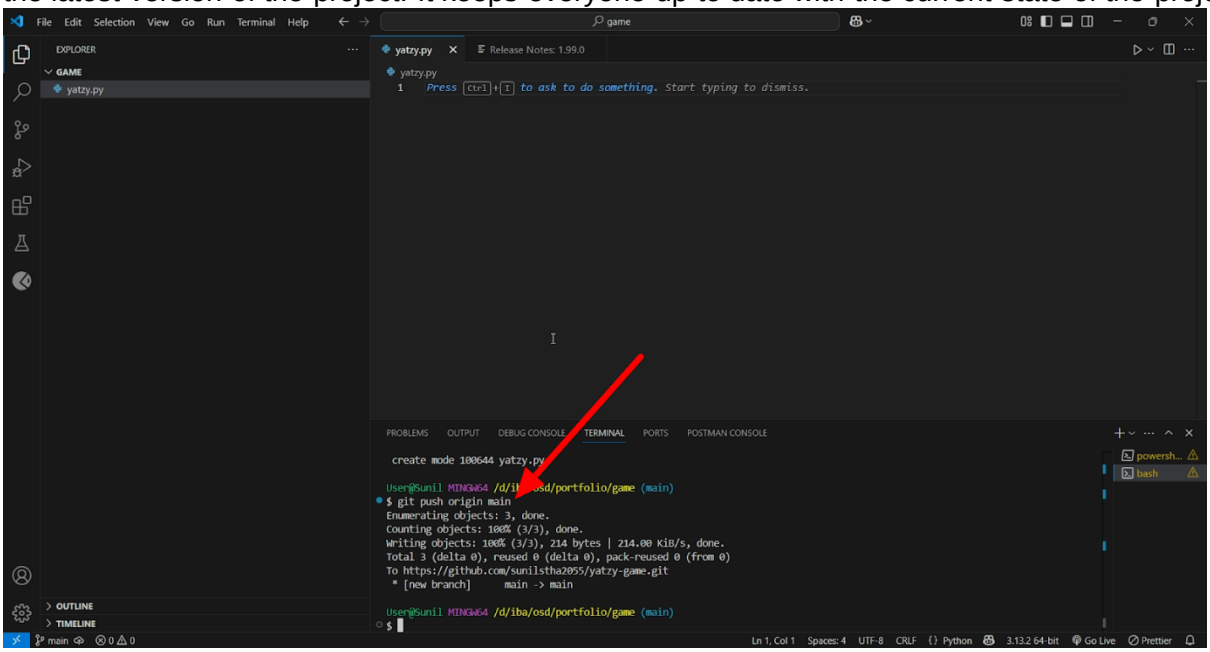
When we use the command git commit, we are telling Git to permanently save the changes we've made and staged with git add. This means the current state of our 4 files, including the changes we've selected, will be saved as a new version or snapshot of our project. Each commit includes a message to describe the changes, making it easier to track the history of our project and understand what was changed and why. Once committed, these changes are safely recorded in the project's history and can be revisited or reverted to if needed.



Push to GitHub

git push origin main

Git push, is the command we use to send our local project changes to a remote repository (like GitHub or GitLab). After we've made changes to our project, staged them with `git add`, and saved them with `git commit`, we use `git push` to share those changes with others. This command uploads our commits to the remote repository so that other team members can see and access the latest version of the project. It keeps everyone up to date with the current state of the project.



Pull Changes

git pull origin main

Git pull is the command we use to download changes from a remote repository (like GitHub or GitLab) to our local project. When other team members make updates to the project and push their changes to the remote repository, we can use git pull to get those updates and apply them to our local files. This keeps our project up to date with the latest changes made by others. It's a way to make sure our local copy of the project is in sync with the remote repository.