# Git Ignore HOL - 2

## Step 1: Create Test Files and Folders

1. In your Git working directory, create a .log file and a log folder:  
 echo "Sample log data" > debug.log  
 mkdir log  
 echo "App log entry" > log/error.txt  
A screenshot of a computer screen

AI-generated content may be incorrect.

## Step 2: Create .gitignore File

1. Create a .gitignore file in your working directory:  
 touch .gitignore  
2. Add patterns to ignore .log files and log folder:  
 echo "\*.log" >> .gitignore  
 echo "log/" >> .gitignore  
A black background with white text

AI-generated content may be incorrect.

## Step 3: Verify Ignoring of Files

1. Check status to confirm debug.log and log/ are ignored:  
 git status  
2. They should NOT appear in the 'Untracked files' list.  
A black screen with white text

AI-generated content may be incorrect. Step 4: Commit .gitignore Changes

1. Add the .gitignore file to staging:  
 git add .gitignore  
2. Commit the changes:  
 git commit -m "Added .gitignore to exclude .log files and log folder"

A black screen with white text

AI-generated content may be incorrect.  
3. Push to remote:  
 git push origin master  
A computer screen shot of white text

AI-generated content may be incorrect.

## Step 5: Verification after Commit

1. Run git status to ensure working directory is clean.  
2. Verify on the remote repository that .gitignore is present but debug.log and log/ folder are not pushed.  
A screenshot of a computer

AI-generated content may be incorrect.