

## Final Fix: Clean Rename with Git Tracking

Step 1: Rename the file wrong on purpose

In your local machine terminal (or VS Code):

```
mv src/pages/FoodPage.jsx src/pages/temp.jsx
```

Then:

```
git add .  
git commit -m "Temp rename to force case change"
```

Step 2: Rename back with correct casing

```
mv src/pages/temp.jsx src/pages/FoodPage.jsx
```

Then:

```
git add .  
git commit -m "Fix filename casing for FoodPage.jsx"  
git push
```

Why this works:

On Windows, Git often ignores casing changes. But Render (Linux) requires exact match like FoodPage.jsx.

Doing this wrong to correct rename forces Git to track the case difference and fix the error.

Bonus Check:

Make sure in src/App.jsx this is exact:

```
import FoodPage from './pages/FoodPage';
```

No .js, no typo, no extension change.

Once you push this, redeploy on Render it should work.

## Git State and Undo Options

You're currently in a Git state where:

- You have 10 commits that haven't been pushed to origin/main.
- You have unstaged changes in these files:
  - src/App.jsx
  - src/components/Navbar.jsx
  - src/pages/Home.jsx

If you want to undo the local changes (unstaged ones) and go back to the last committed state:

```
git restore src/App.jsx src/components/Navbar.jsx src/pages/Home.jsx
```

Or to undo all unstaged changes at once:

```
git restore .
```

If you want to keep the changes but stage them for commit:

```
git add src/App.jsx src/components/Navbar.jsx src/pages/Home.jsx
```

Or simply:

```
git add .
```

If you want to completely discard ALL local changes and commits (including the 10 commits), and reset everything to match origin/main:

Warning: This will remove all your local commits and changes. Be very careful.

```
git reset --hard origin/main
```

Let me know what exactly you want:

- Undo just the recent edits?
- Remove unpushed commits?
- Keep edits and commit them?
- Push your changes?