Cloudflare Worker Project Setup: Step-by-Step Commands

1. Create a new Cloudflare Worker project

```
npm create cloudflare -- my-app
```

- When prompted, confirm with `y` to install the required package.
- Follow the interactive prompts:
    - Directory: `./my-app`
    - Start with: `Hello World example`
    - Template: `Worker only`
    - Language: `TypeScript`
- Let it finish copying files, updating `package.json`, and installing dependencies[1].

2. Change to your new project directory

```
cd my-app
```

3. Install and log in to Wrangler (Cloudflare CLI)

```
npx wrangler login
```

- Confirm with `y` to install Wrangler if prompted.
- Complete the OAuth login process in your browser[1].

4. Verify your Cloudflare account and permissions

```
npx wrangler whoami
```

- Confirms you are logged in and shows your account details and permissions[1].

5. Start the development server (optional, for local testing)

```
npm run start
```

6. Deploy your Worker to Cloudflare

```
npx wrangler deploy
```

- This uploads and deploys your Worker. The output will include your deployed Worker's URL [1].

Troubleshooting Common Issues

- **If you see "Missing entry-point to Worker script" error:**
  - Make sure you are inside your project directory (`cd my-app`).
  - Ensure your `wrangler.jsonc` (or `wrangler.toml`) file has the correct `"main"` field pointing to your entry file, e.g., `"main": "src/index.ts"` [1].
- **If you see "Could not read package.json" error:**
  - Run commands inside the project directory where `package.json` exists [1].

Summary Table

Step	Command	Purpose
1	npm create cloudflare -- my-app	Scaffold a new Worker project
2	cd my-app	Enter project directory
3	npx wrangler login	Authenticate with Cloudflare
4	npx wrangler whoami	Verify account and permissions
5	npm run start	(Optional) Run local dev server
6	npx wrangler deploy	Deploy Worker to Cloudflare

References:
All commands and steps are based on your provided terminal session and Cloudflare's interactive prompts [1].

❄

1. paste.txt