



SLACK BOTS

TO TEST YOUR THE LOCAL VERSION OF YOUR BOT ON THE COMMAND LINE

For testing changes locally before pushing your code.

1. At the command line, navigate to the main folder for your bot.
2. Type `./bin/hubot` and press **Enter**. This gives a few warning messages that you can safely ignore, as long as the BASH prompt is **not** displayed again after them.
3. Press **Enter** again, after which you should see a prompt such as `hubot>` where you can type commands to your bot.
4. When you're done testing, press **control + C** to return to the BASH prompt. Note that your command prompt interactions reflect the state of your `script.js` file at the time you launched the bot prompt. If you make a change to your `script.js` file and save it while the prompt is running, the change will not be reflected in your interactions with an existing bot prompt. You need to press **control + C** to close the Hubot prompt, then repeat instructions 2 & 3 above to relaunch it.

TO TEST THE HEROKU VERSION OF YOUR BOT ON SLACK

For interacting on Slack with code you have pushed.

1. At the command line, navigate to the main folder for your bot.
2. Push your code to Heroku using the following commands:

```
git add .  
git commit -m "description of changes"  
git push heroku master
```
3. Open a direct message with the bot you're using in the class Slack organization (such as `redbot` or `greenbot`).
4. Send a direct message to interact with your bot.
If you're sharing a bot, remember that the bot's replies may reflect the code of the classmate sharing your bot on Slack, as well as your own code, so you may get multiple replies to a single test message.

TO TEST THE HEROKU VERSION OF YOUR BOT ON THE COMMAND LINE

For interacting at the command line with code you have pushed to Heroku.

1. At the command line, navigate to the main folder for your bot.
2. Push your code to Heroku using the following commands:

```
git add .  
git commit -m "description of changes"  
git push heroku master
```
3. Type `HUBOT_SLACK_TOKEN=TOKEN-GOES-HERE ./bin/hubot --adapter slack` and press **Enter**, where *TOKEN-GOES-HERE* is the Slack token provided by your instructor.
*This gives a few warning messages that you can safely ignore, as long as the BASH prompt is **not** displayed again after them.*
4. Press **Enter** again, after which you should see a prompt such as `hubot>` where you can type commands to your bot.
5. When you're done testing, press **control + C** to return to the BASH prompt.
*Note that your command prompt interactions reflect the state of your `script.js` file at the time you launched the bot prompt. If you make a change to your `script.js` file and push it to Heroku while the prompt is running, the change will not be reflected in your interactions with an existing bot prompt. You need to press **control + C** to close the Hubot prompt, then repeat instructions 3 & 4 above to relaunch it.*