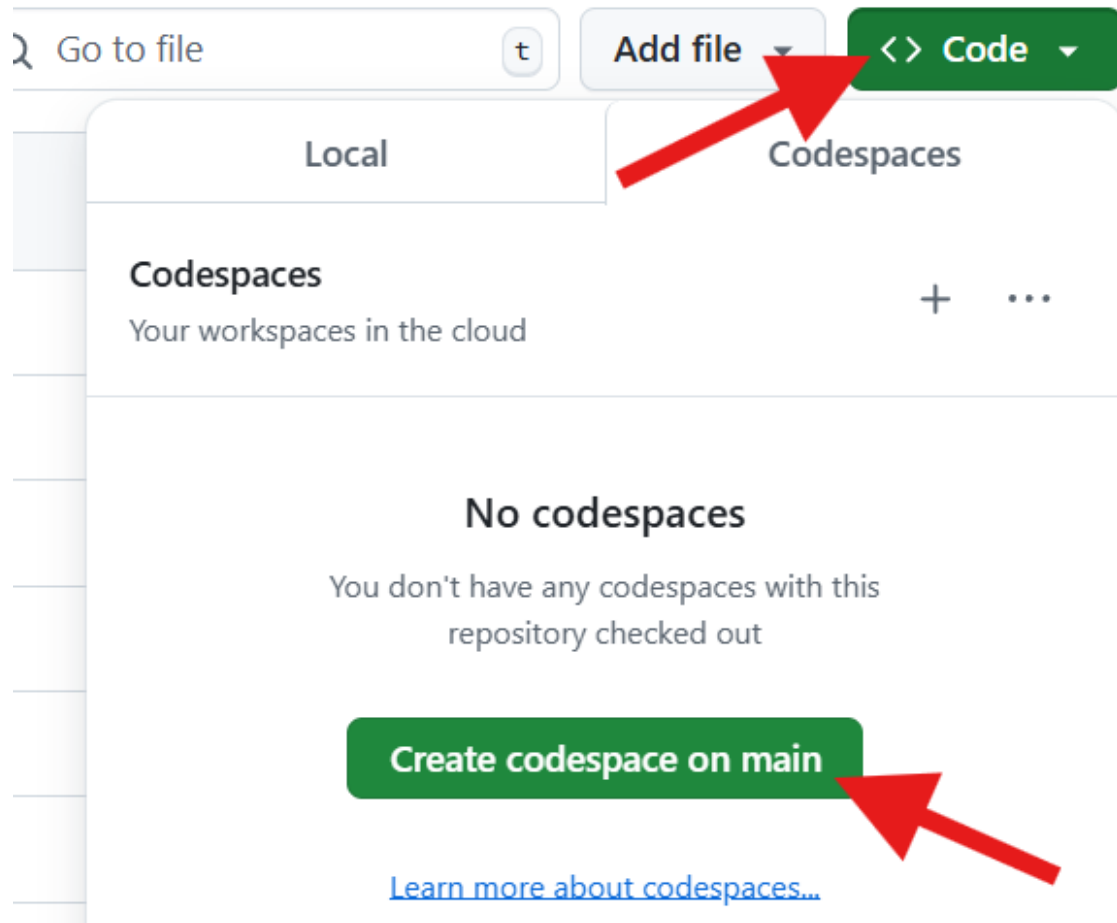


How to Run Flutter App in Codespace

CSCI 3341 SOFTWARE ENGINEERING II

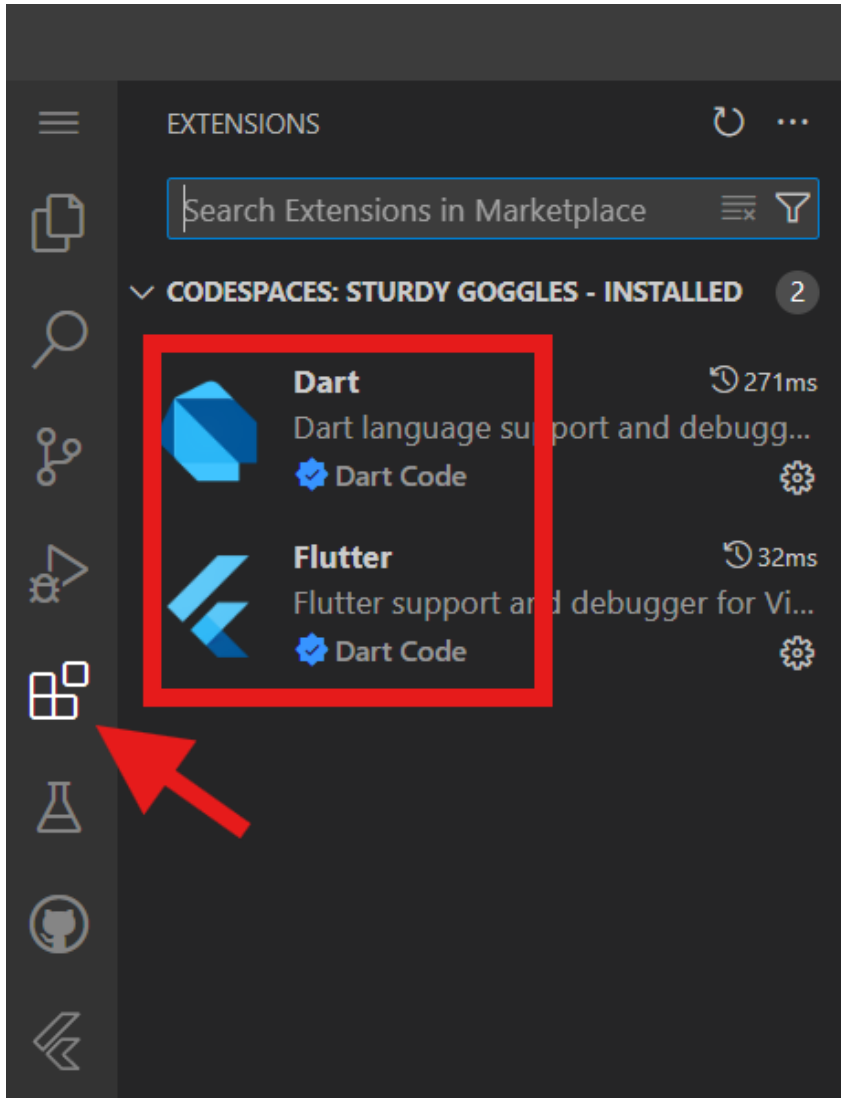


Create your Codespace

The first step is to create a new codespace.

Accept the assignment from Blackboard, then navigate to the project directory and you will see a green **Code** button.

Click on the button, select the **Codespaces** tab, and select **Create codespace on main** button.



Install VS Code Extensions

Wait for the codespace to finish setting up. I advice you to change to a dark theme (it's easier on the eyes).

Select the **Extensions** tab in the left-sidebar and install the **Dart** and **Flutter** extensions (normally they will install by default).

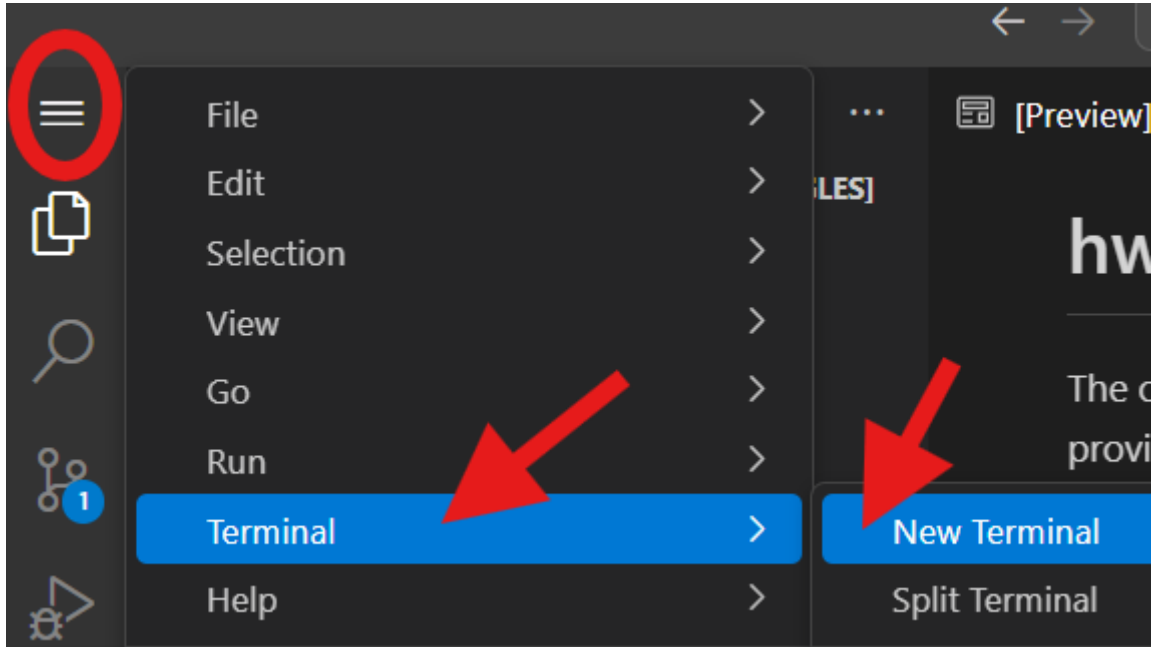
These extensions will provide much needed tools and shortcuts.

Open Terminal

Next open a new Terminal.

Select the options tab in the left-sidebar, then select **Terminal**, and **New Terminal**.

This will open a new terminal that will appear at the bottom.

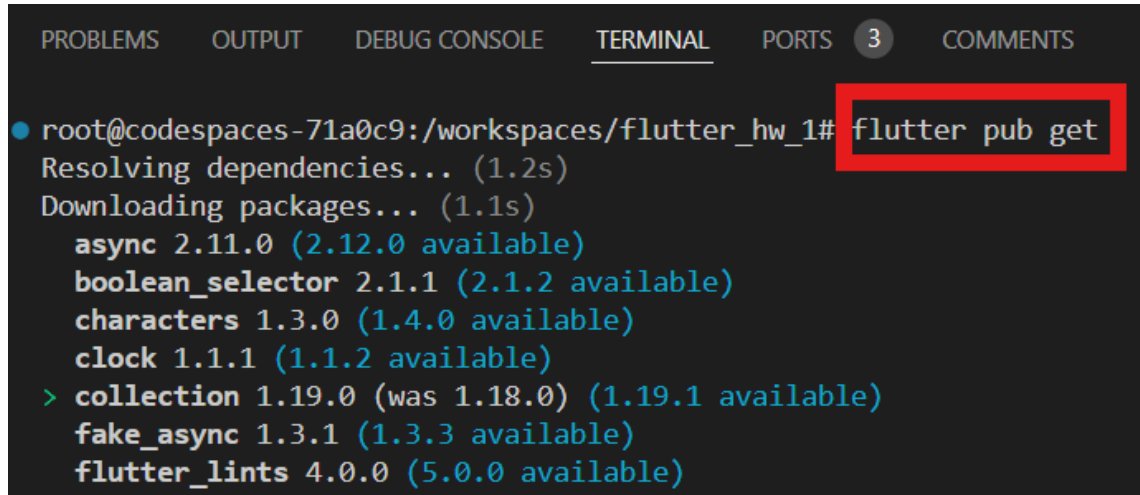


Get Flutter Dependencies

Install the necessary Flutter dependencies from the pubspec.yaml file.

You can do this by executing the following command in the terminal.

- `flutter pub get`



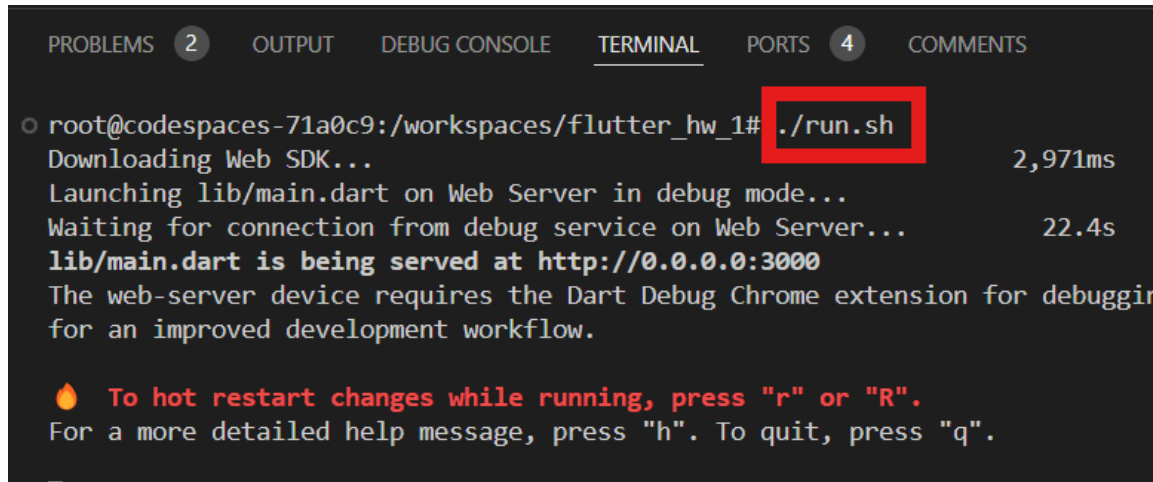
```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS 3 COMMENTS
● root@codespaces-71a0c9:/workspaces/flutter_hw_1# flutter pub get
Resolving dependencies... (1.2s)
Downloading packages... (1.1s)
  async 2.11.0 (2.12.0 available)
  boolean_selector 2.1.1 (2.1.2 available)
  characters 1.3.0 (1.4.0 available)
  clock 1.1.1 (1.1.2 available)
> collection 1.19.0 (was 1.18.0) (1.19.1 available)
  fake_async 1.3.1 (1.3.3 available)
  flutter_lints 4.0.0 (5.0.0 available)
```

Start App Using Script

Next, start the flutter application by executing the run.sh script.

Type the following command in your terminal:

- `./run.sh`



The screenshot shows a terminal window with a dark background. At the top, there are tabs for 'PROBLEMS', 'OUTPUT', 'DEBUG CONSOLE', 'TERMINAL' (which is active), 'PORTS', and 'COMMENTS'. The terminal output shows the command `./run.sh` being executed, which is highlighted with a red box. The output includes: 'Downloading Web SDK...' (2,971ms), 'Launching lib/main.dart on Web Server in debug mode...', 'Waiting for connection from debug service on Web Server...' (22.4s), and 'lib/main.dart is being served at http://0.0.0.0:3000'. A message at the bottom states: 'The web-server device requires the Dart Debug Chrome extension for debugging for an improved development workflow.' Below this, there is a red flame icon and text: 'To hot restart changes while running, press "r" or "R". For a more detailed help message, press "h". To quit, press "q".'

```
PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS 4 COMMENTS
root@codespaces-71a0c9:/workspaces/flutter_hw_1# ./run.sh
Downloading Web SDK... 2,971ms
Launching lib/main.dart on Web Server in debug mode...
Waiting for connection from debug service on Web Server... 22.4s
lib/main.dart is being served at http://0.0.0.0:3000
The web-server device requires the Dart Debug Chrome extension for debugging
for an improved development workflow.

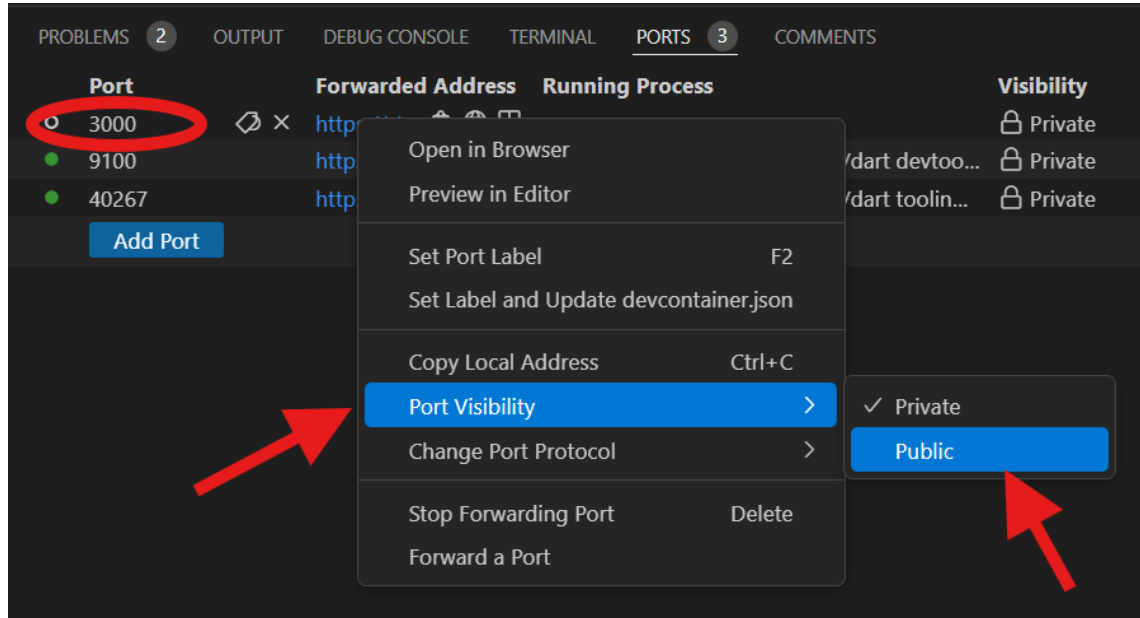
🔥 To hot restart changes while running, press "r" or "R".
For a more detailed help message, press "h". To quit, press "q".
```

Make Port 3000 Public

To view your application, go to the **PORTS** tab next to the **TERMINAL** tab.

Right-click on the **3000** Port and select **Port Visibility**, then select **Public**.

This will make the link where your app is being hosted visible to you.



Open Link in Browser

Lastly, right-click on the **3000** Port again and select **Open in Browser**.

This will open a new tab in your browser where you'll be able to see your app.

Keep in mind, it may take a couple of minutes to load.

