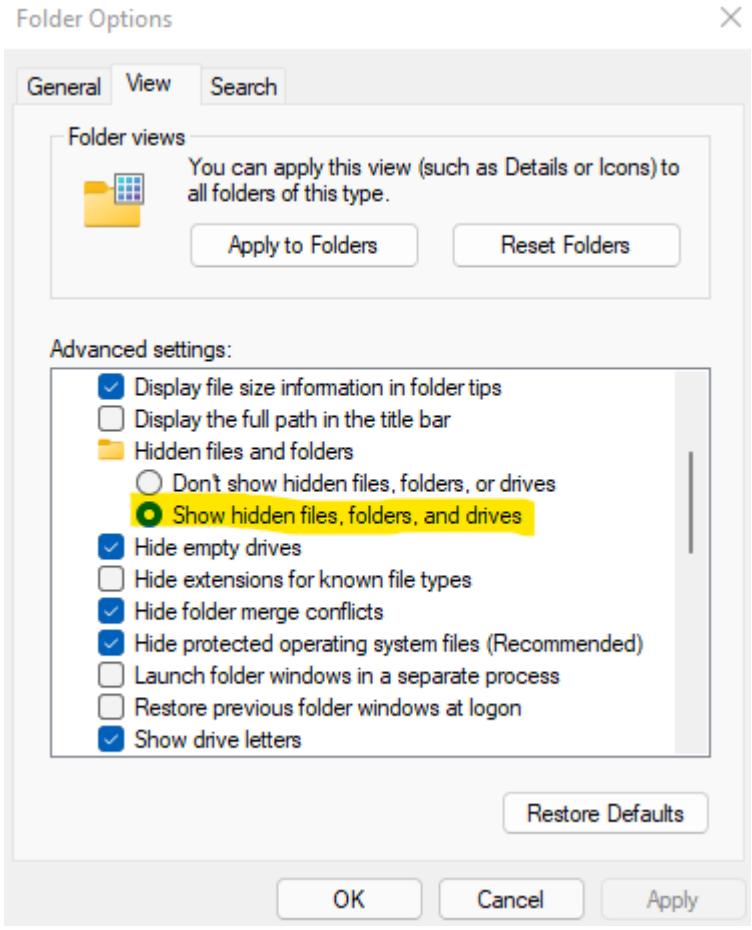
Section xx: How to "Clean" the IDE

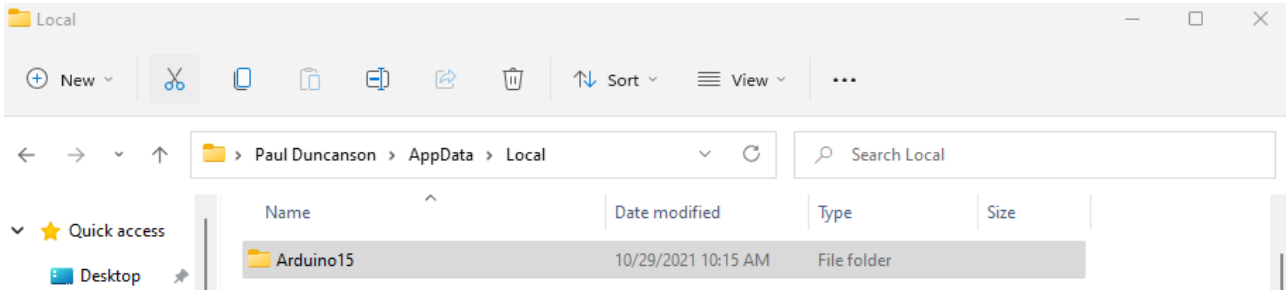
Starting with an Arduino IDE that does not contain any other packages is essential for testing purposes. An IDE with no other packages installed will be considered clean. The IDE appears to have the ability to pull missing variables and tools from other packages that have been installed. This means that testing a new package should be done on a "clean" version of the IDE.

Remove extra packages

1. Make sure the Arduino IDE and any associated files are closed.
2. Locate the Arduino15 folder.
    - Path: "C:\Users\USERNAME\AppData\Local\Arduino15"
    - On most Windows machines AppData will be a hidden folder, to show the folder select folder options and click "Show hidden files, folders, and drives" under the "Hidden files and folders" section.

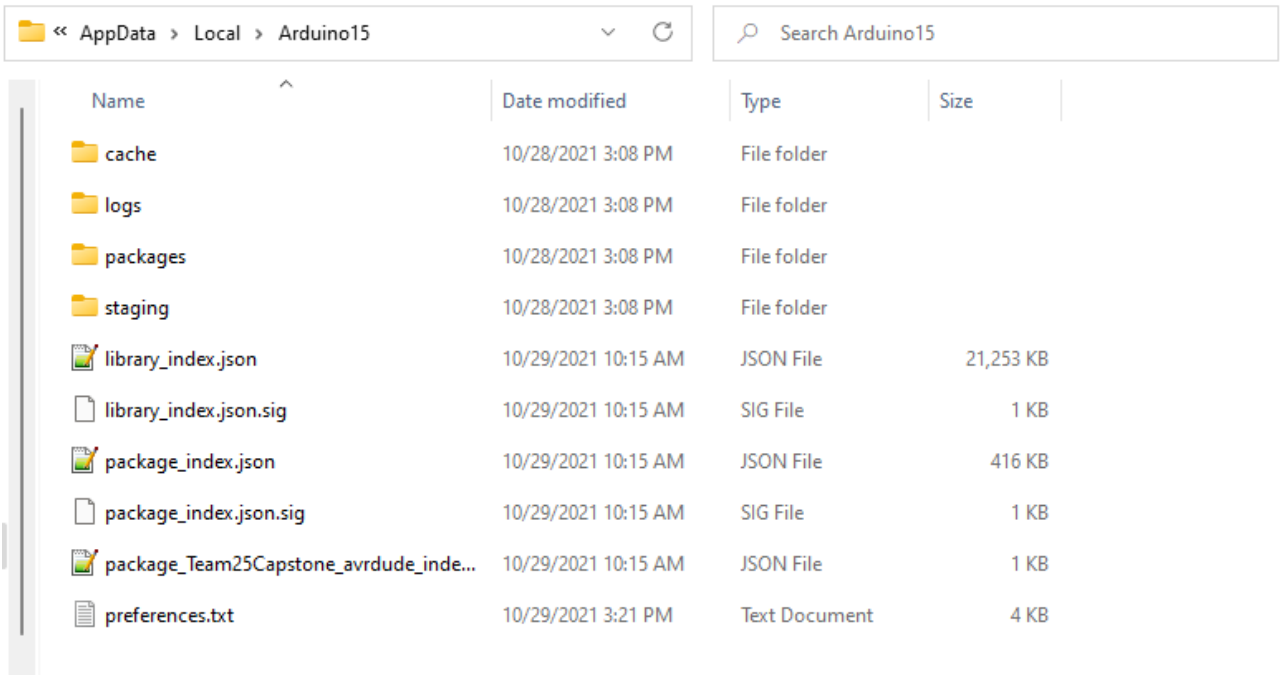


3. Delete the Arduino15 folder.

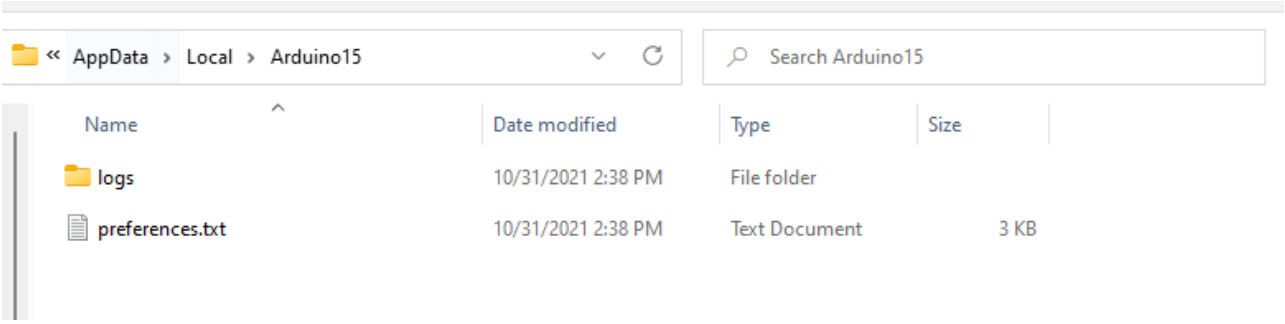


4. Restart the Arduino IDE.
    - The IDE will generate a new empty Arduino15 folder.

Before:



After:

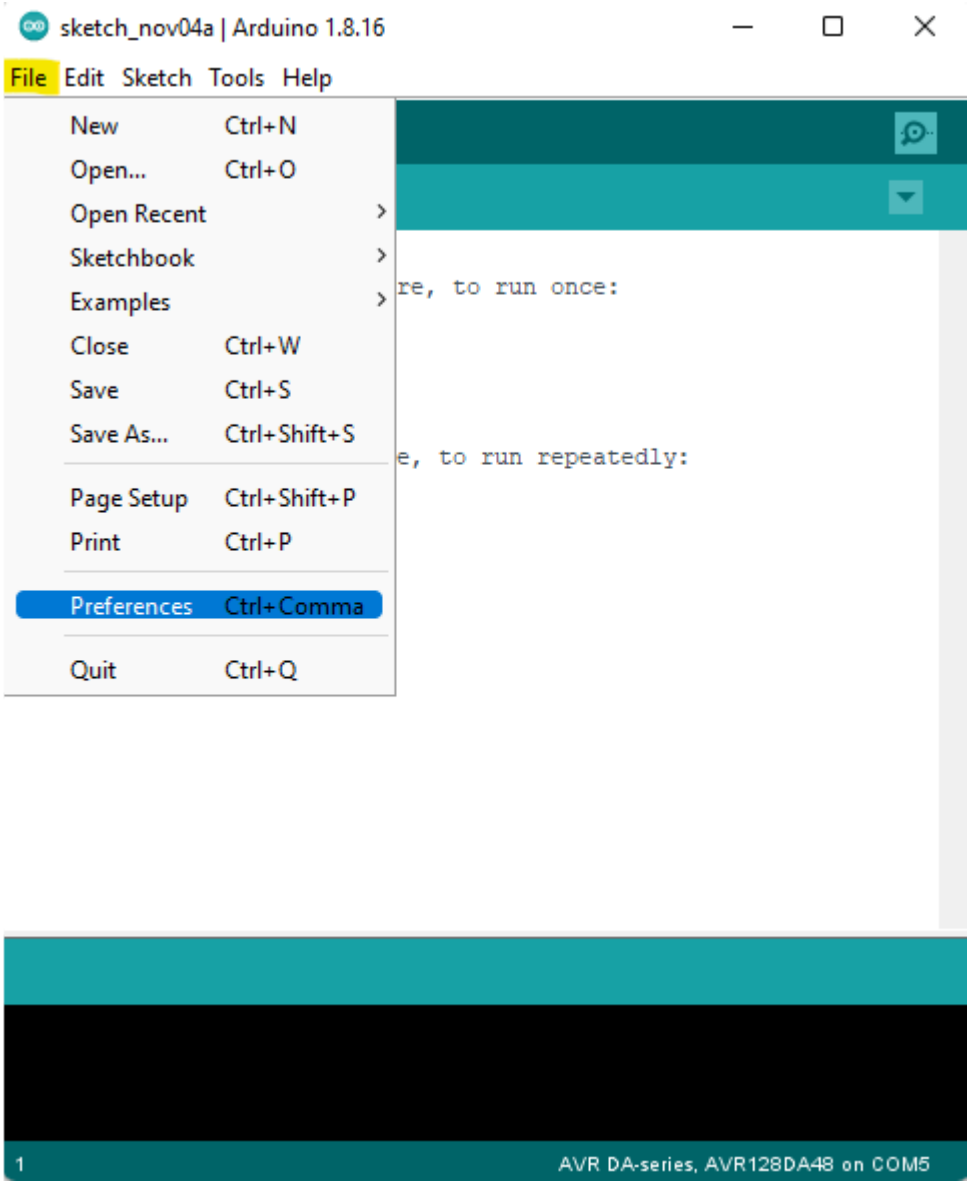


Section xx: How to add a new Package

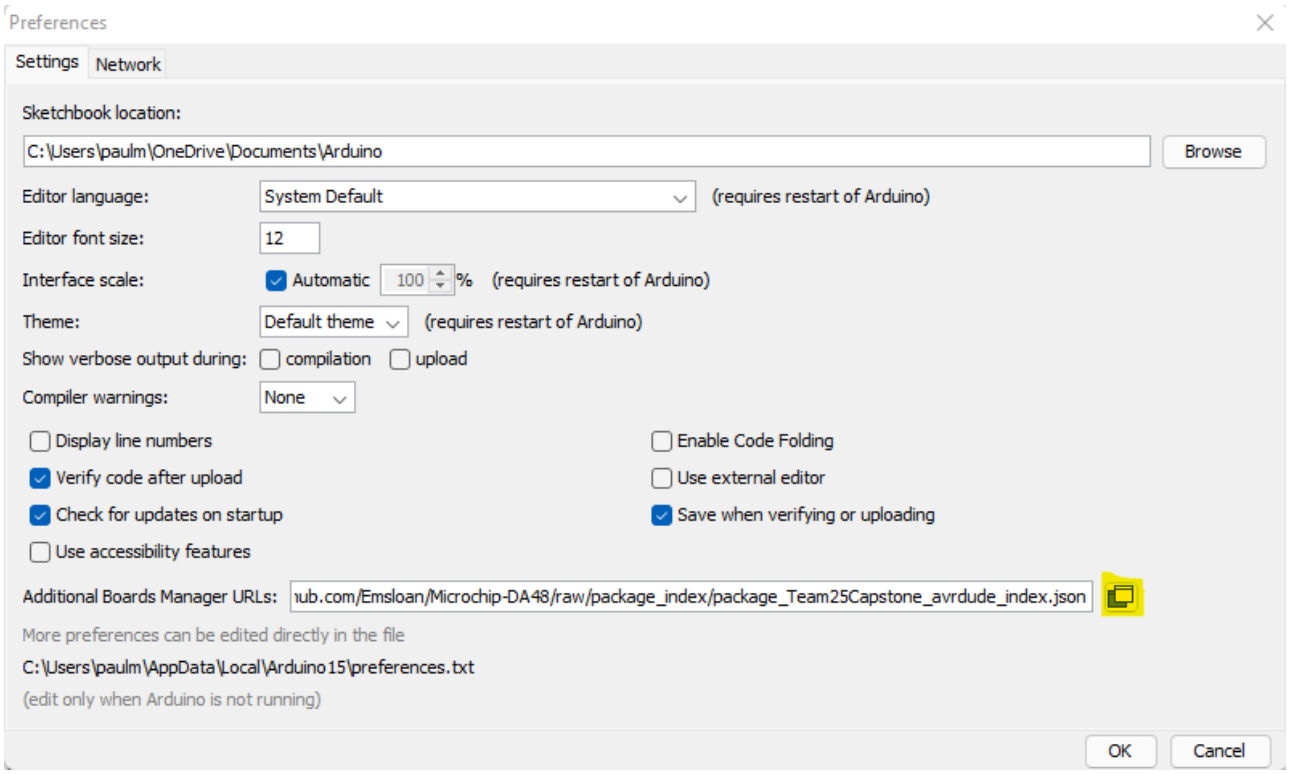
The IDE requires adding URLs pointing to JSON formatted files to install new core packages. The structure of these JSONs is specified by the IDE and will point to necessary tools and information to allow for new boards to interact with the IDE. Please see section xx in this document for more information about creating a JSON file.

Setting up a JSON file in the IDE

1. Click "File" menu and select the "Preferences" option.



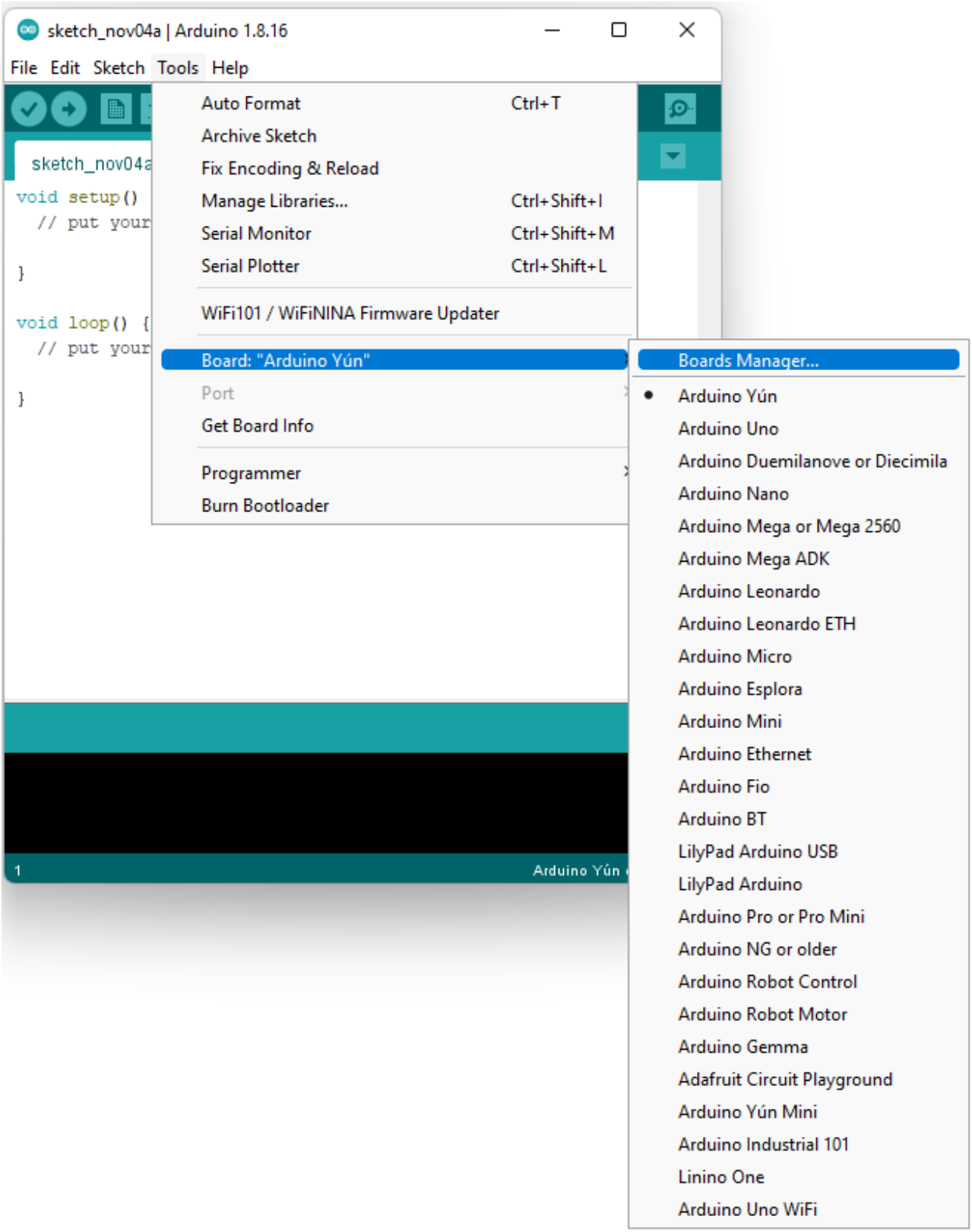
2. In the Preferences menu, click the button to the far right of the "Additional Boards Manager URL" option.



3. Setting up a new JSON requires adding the URL to the next line in the "Additional Boards Manager".

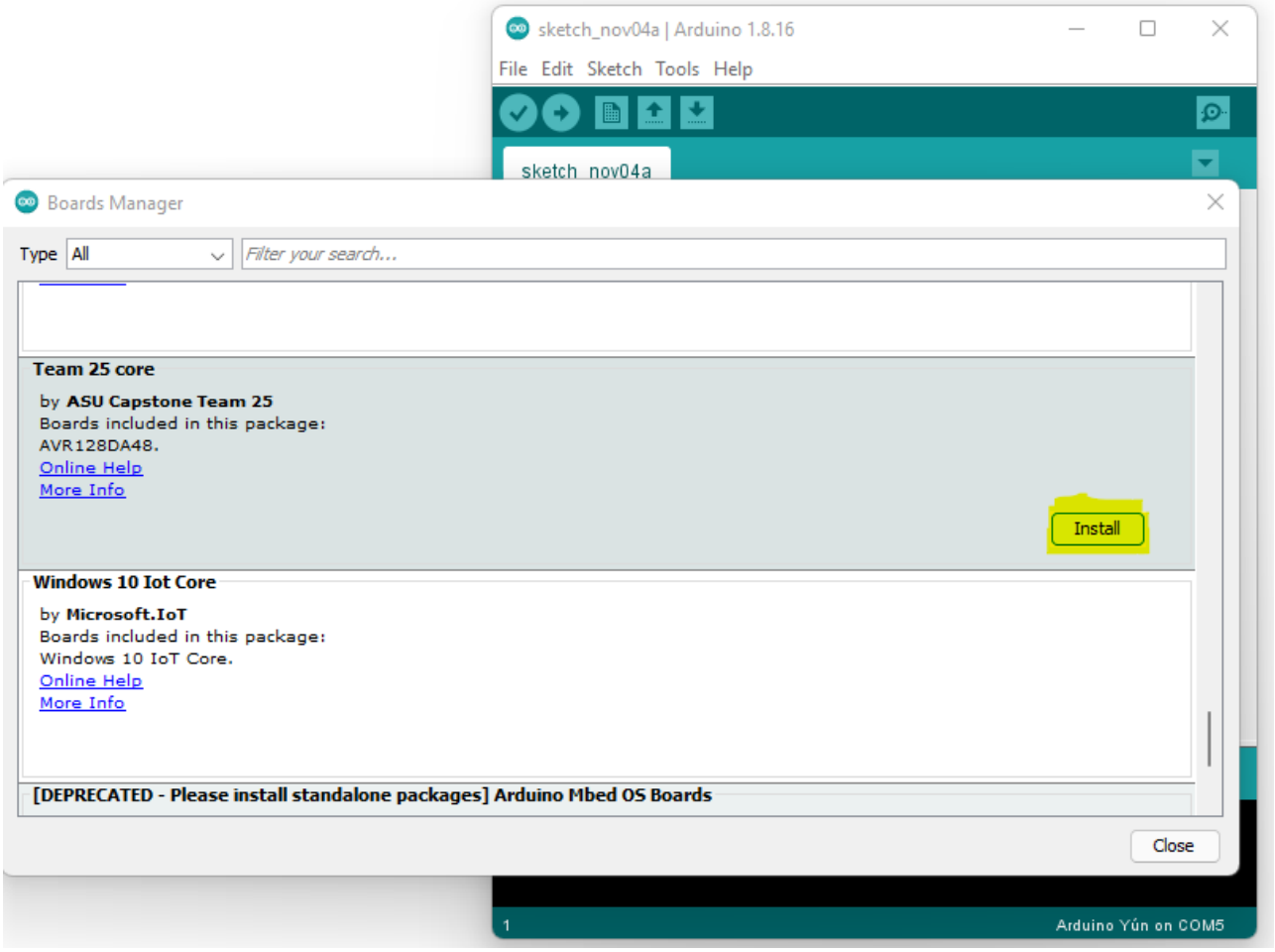
Installing the Package

4. Click "Tools" menu and select "Board" followed by "Boards Manager".

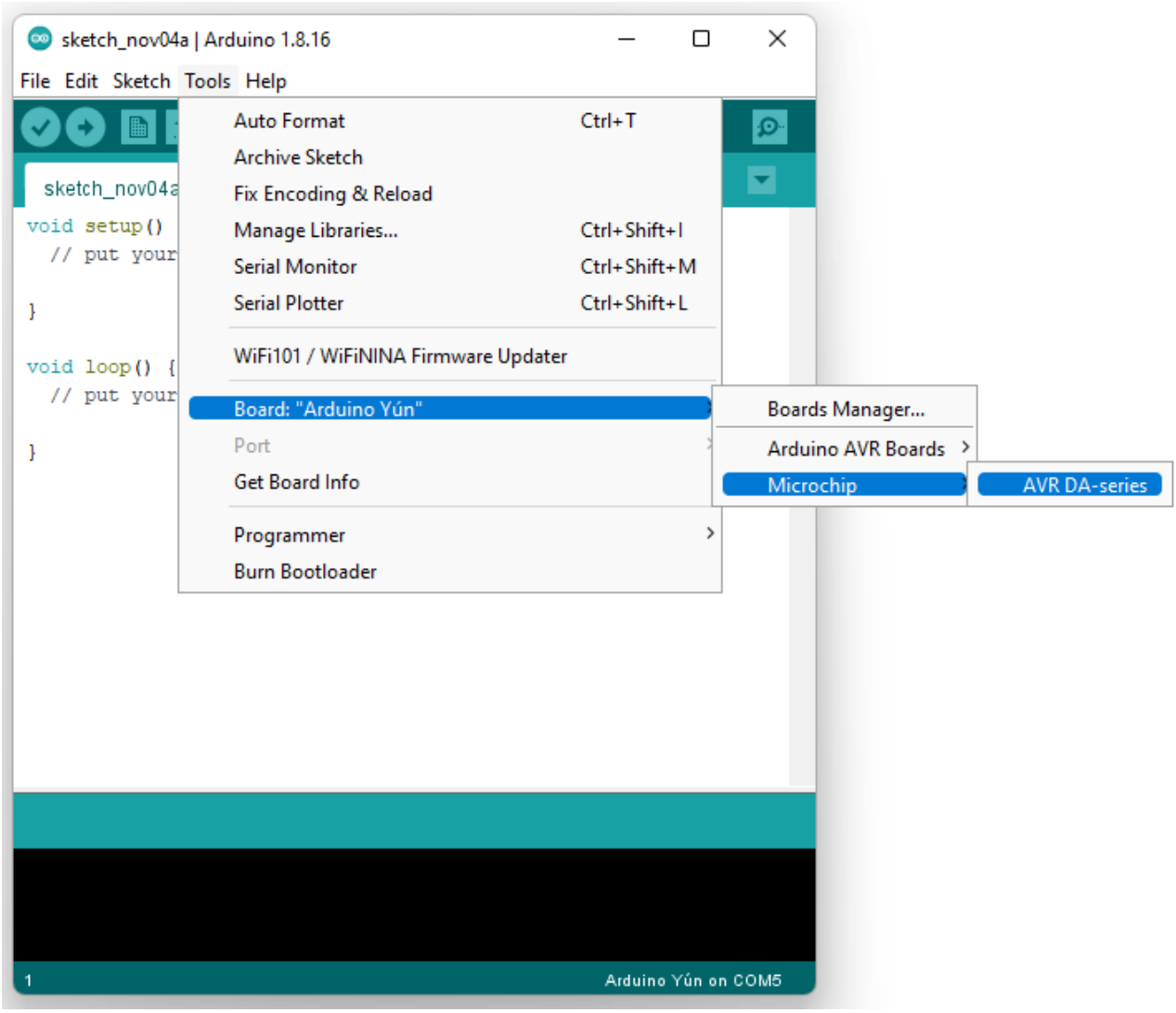


5. The boards manager will update the package options with the new package.

6. Find the added package and select "Install".



7. Installed package will be available under the "Tools" - "Boards" menu.

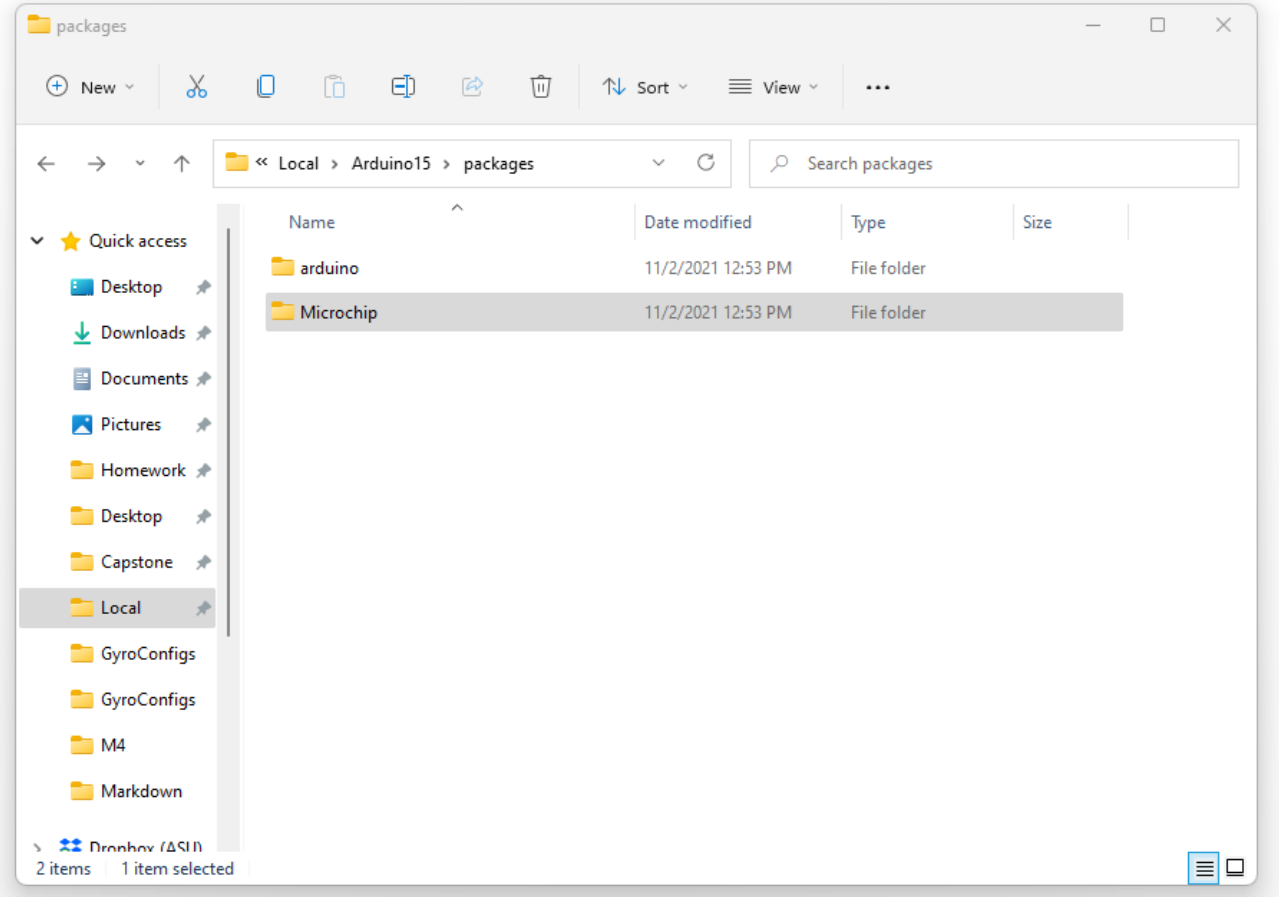


Section xx: How to test a core package within the IDE

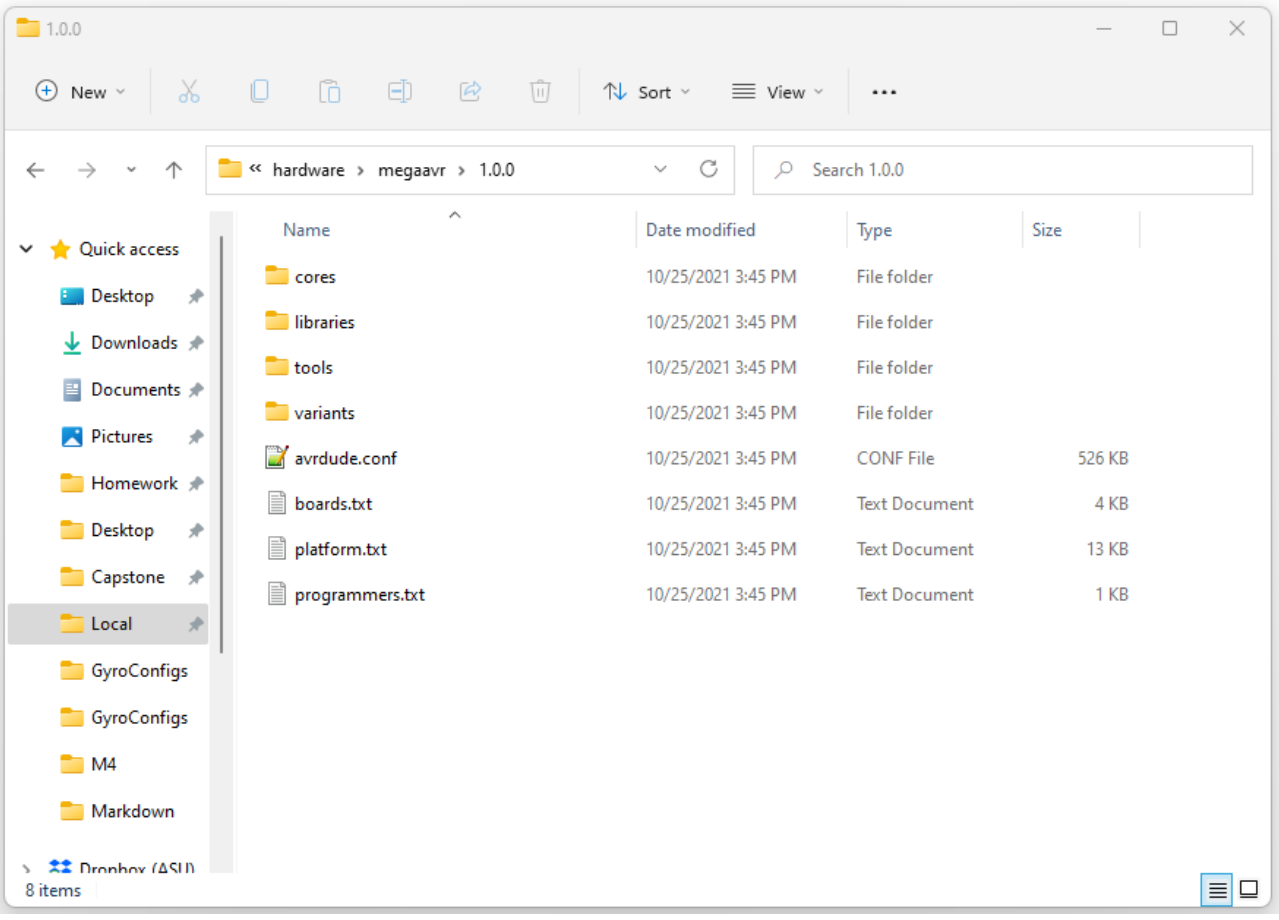
Testing out a new core package can be difficult when every change requires updating the JSON file. These changes require the SHA-256 key to be updated everytime. The IDE will not pull the package's zip file if the SHA-256 code does not match the zip file's source. This means that testing changes made in a package is easier to conduct in an already installed package.

Modifying and testing a core package

1. The "Arduino15" folder contains the folders and files that make a core package.



2. Select the desired package from the "Packages" folder.



3. Make updates to desired files in the package and save.
4. Restart the IDE.

5. The IDE will reinitialize installed packages.

6. Updated package will now be in use.

Section xx: How to setup the SHA-256 key

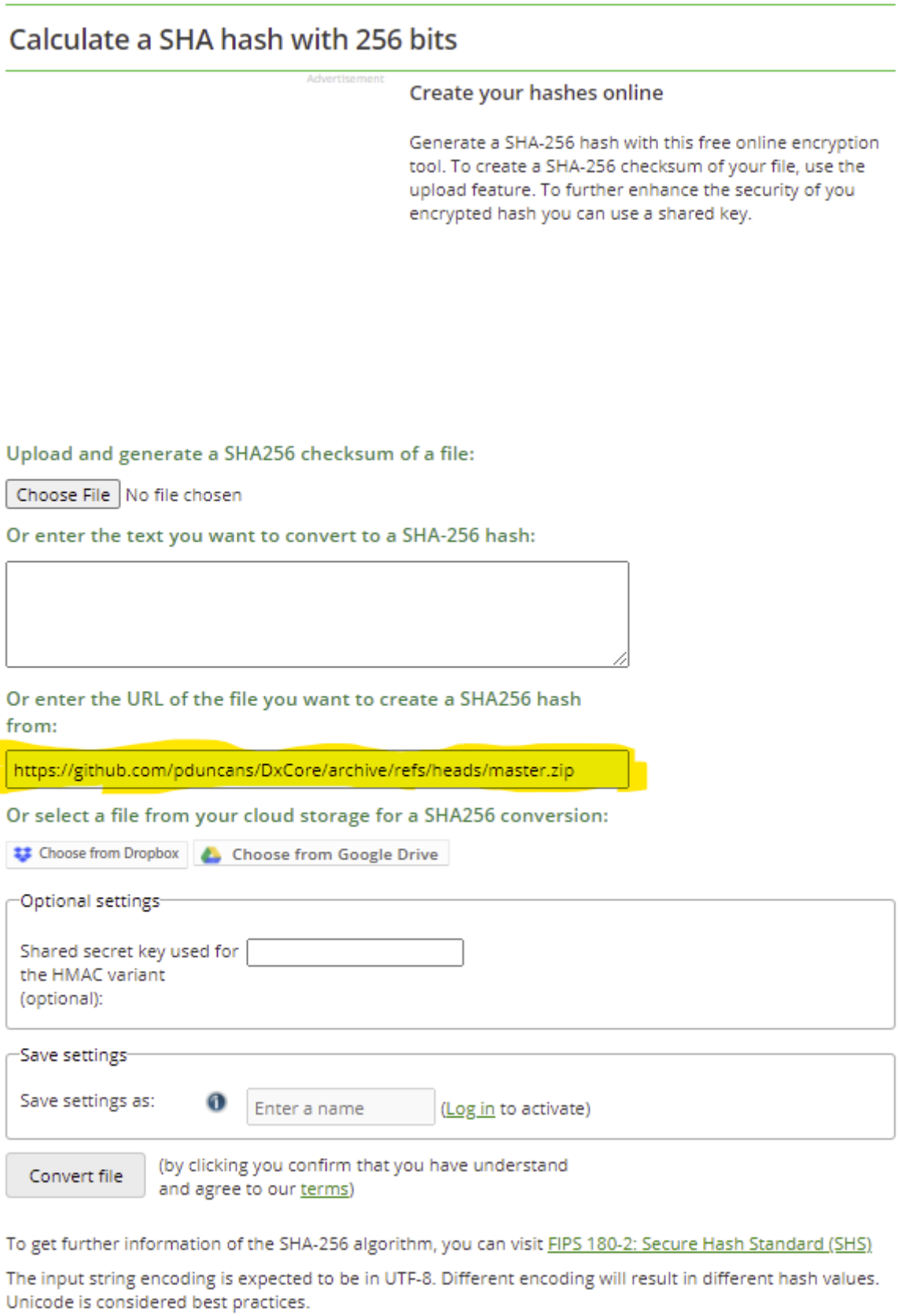
A Secure Hash Algorithm (SHA) is a hashing function that can be used for validation purposes. The Arduino IDE uses the SHA-256 standard as a method of varifying Arduino core packages. This verification process is necessary to install new packages. Package owners generate a unique SHA-256 key from their core repository and must add this key to the JSON file.

1. The "checksum" section of the JSON contains the unique SHA-256 key.

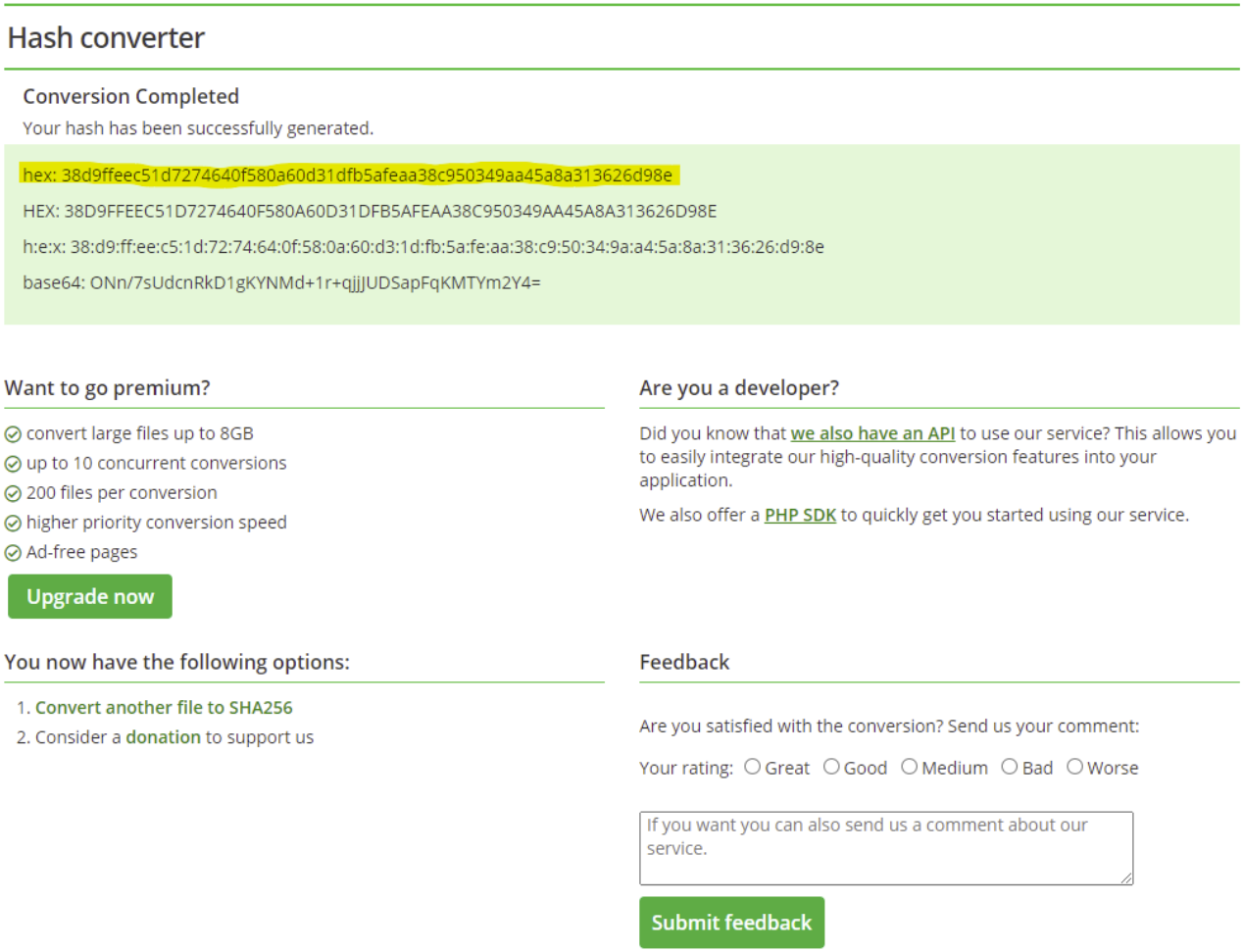
```
{
    "packages": [
        {
            "name": "Microchip",
            "maintainer": "ASU Capstone Team 25",
            "websiteURL": "https://github.com/Emsloan/Microchip-DA48",
            "email": "emsloan@asu.edu",
            "help": {
                "online": "https://github.com/Emsloan/Microchip-DA48"
            },
            "platforms": [
                {
                    "name": "Team 25 core",
                    "architecture": "megaavr",
                    "version": "1.0.0",
                    "category": "Contributed",
                    "url": "https://github.com/Emsloan/Microchip-DA48/archive/refs/heads/master.zip",
                    "archiveFileName": "Microchip-DA48-master.zip",
                    "checksum": "SHA-256:87ddb9416625115d108e3318925cdd07b9fda34f096456b34c0830afbd38b27d",
                    "size": "133000",
                    "boards": [
                        {
                            "name": "AVR128DA48"
                        }
                    ],
                    "toolsDependencies": [
                        {
                            "packager": "arduino",
                            "name": "avrdude",
                            "version": "6.3.0-arduino18"
                        }
                    ]
                }
            ],
            "tools": []
        }
    ]
}
```

2. The SHA-256 key can be generated online here:
    - https://hash.online-convert.com/sha256-generator

3. Add a link to the repo's URL and click "Convert file".



4. Copy the "hex:" value into the "checksum" section of the JSON



5. The SHA-256 key must be updated everytime the package repository is changed.

6. If the checksum key doesn't match the zipped package signature, the IDE will throw an error.