2.4 Mac development environment(optional)

1. Download Kendryte IDE

1.1 Kendryte IDEdownload link:

https://github.com/kendryte/kendryte-ide/releases
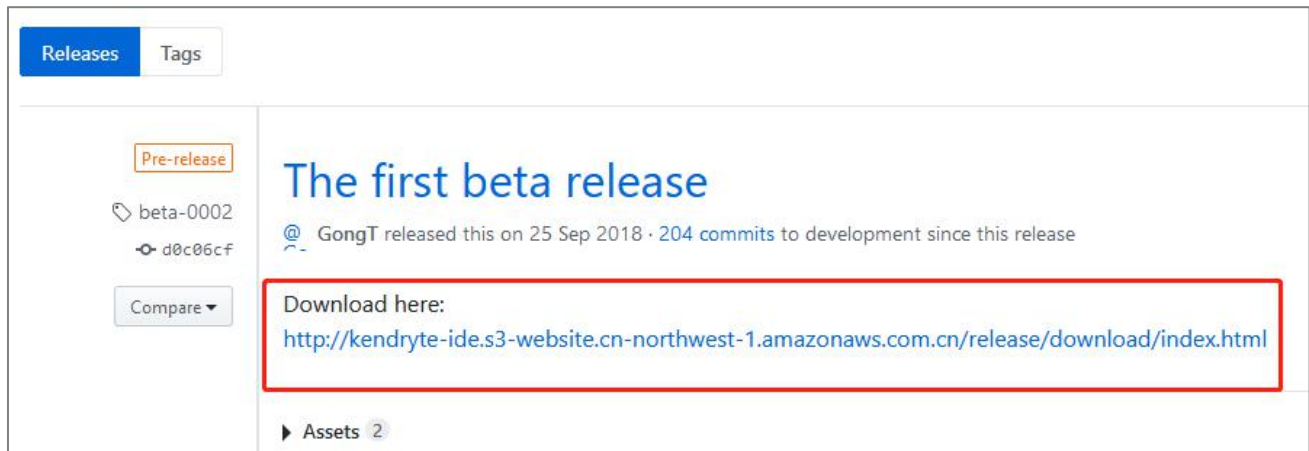
(We have provide this file, you can click [Tools] to get this file.)

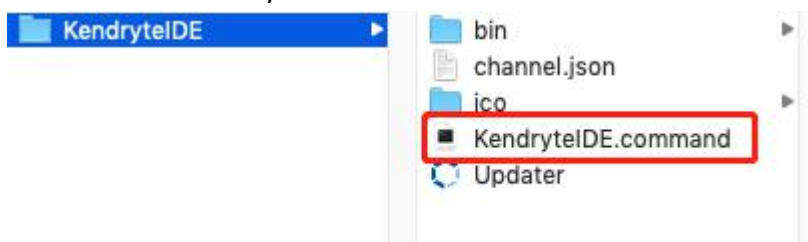1.2 Click Mac version Kendryte IDE.

1.3 After download is complete, extract **.zip** file. You will get a **updater.beta.20190604.4.win32** folder with KendryteIDE folder.

1.4 Install Kendryte IDE

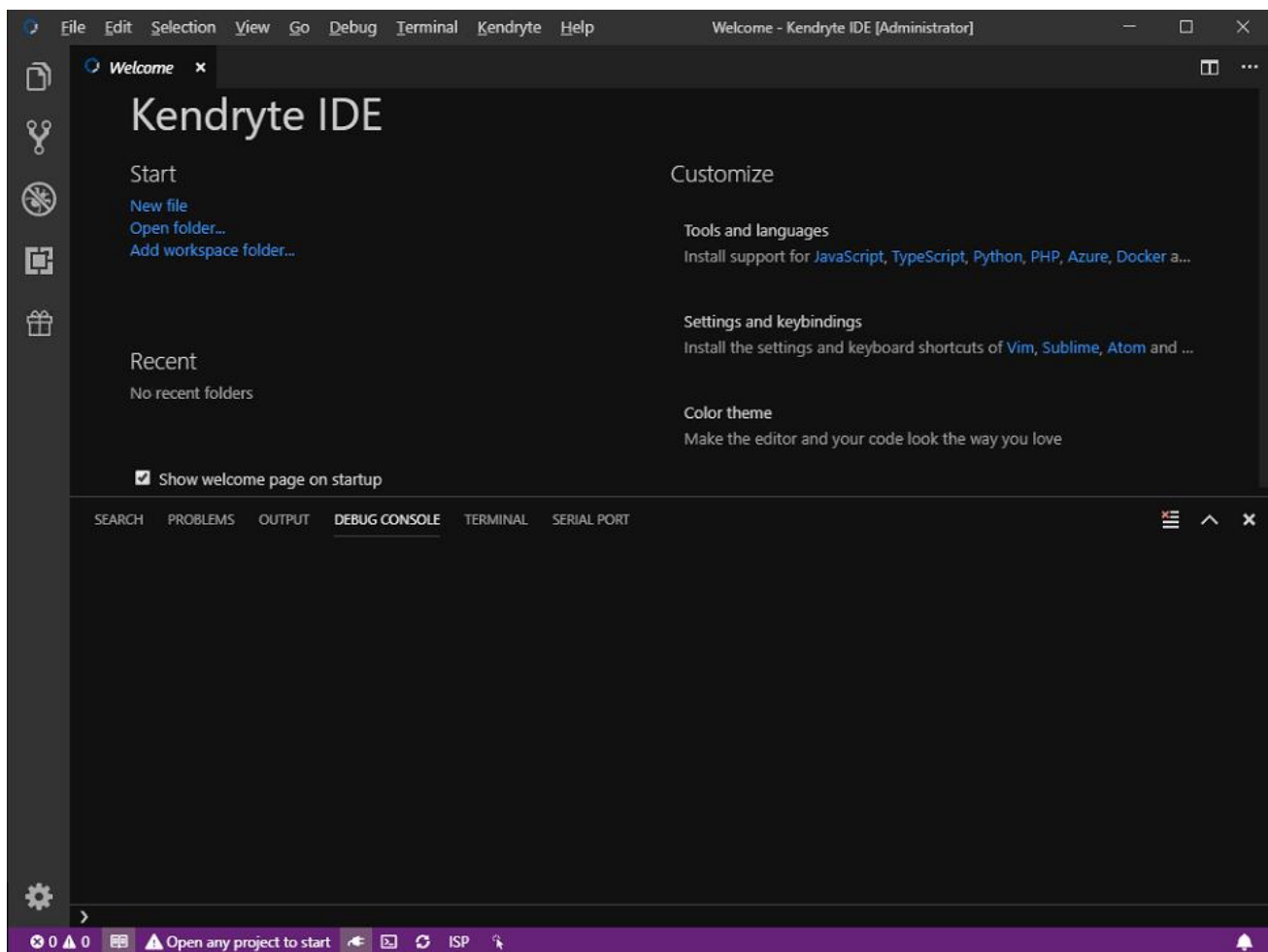**Due to Kendryte IDE is an unsigned software, please do not put it in the application on your MAC.**

Double-click "KendryteIDE.command" to run the installer.

If there are any security prompts on the system, you need to select "OK" and "open"

1.5 Wait for the installation.

1.6 After the installation is complete, open KendryteIDE. You can see following interface.
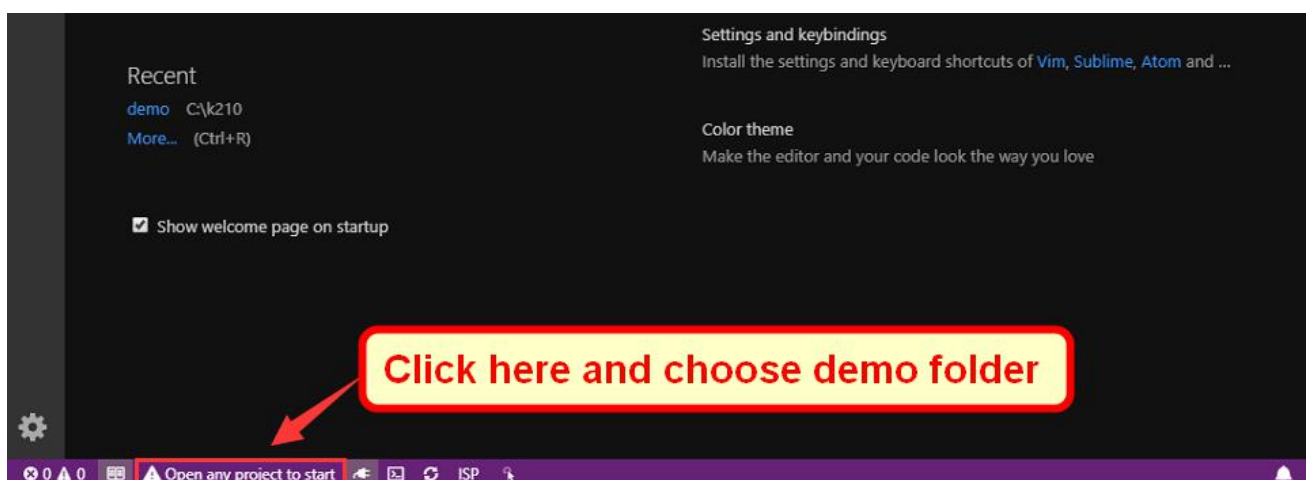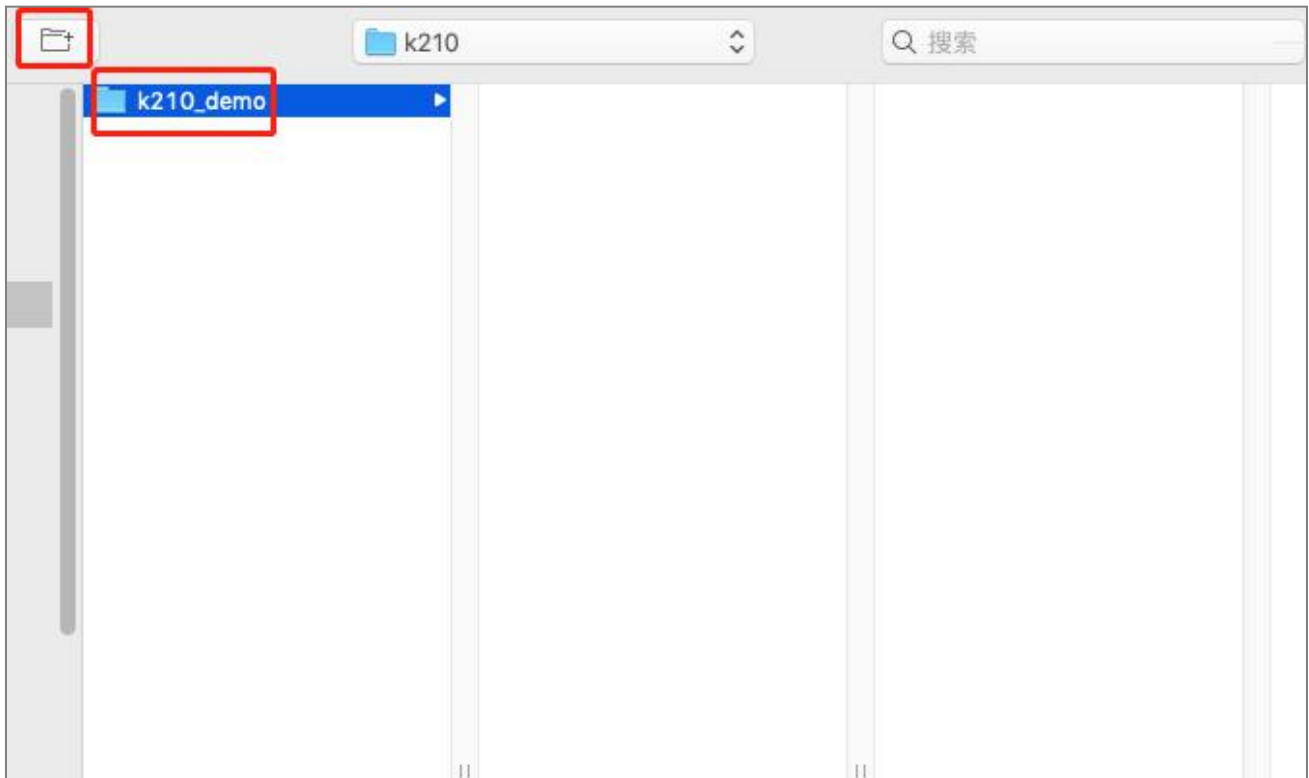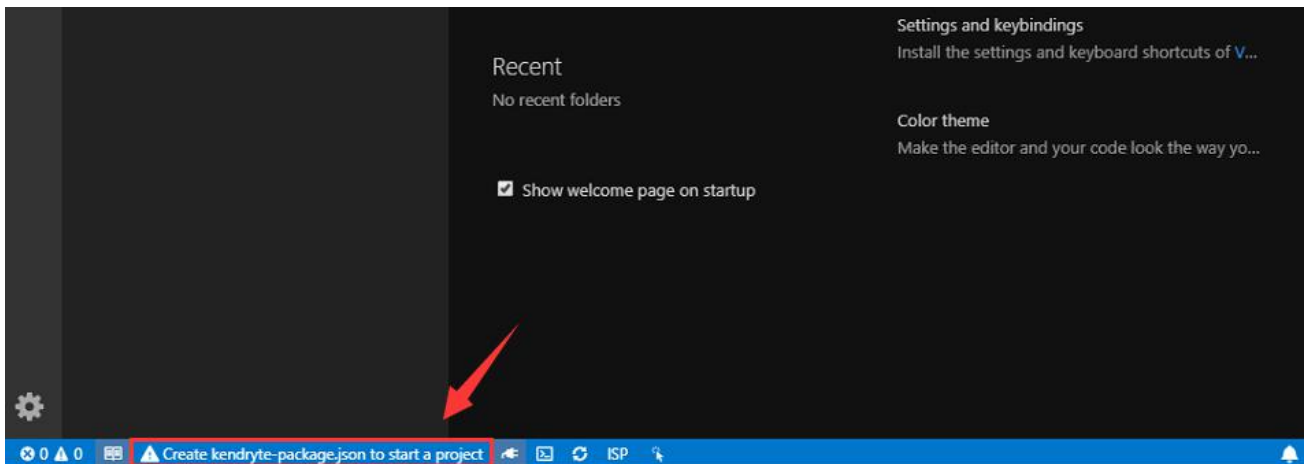

2. New create project

2.1 New create project directory

Click on the open directory in the lower left corner to start a project.

Choose a folder path to save the project, we use k210_demo as an example.

2.2 Click on the lower left corner of the interface to new create **kendryte-package.json** project.


2.3 kendryteIDE will help us create the project, the src folder is used to store our source files and header files.
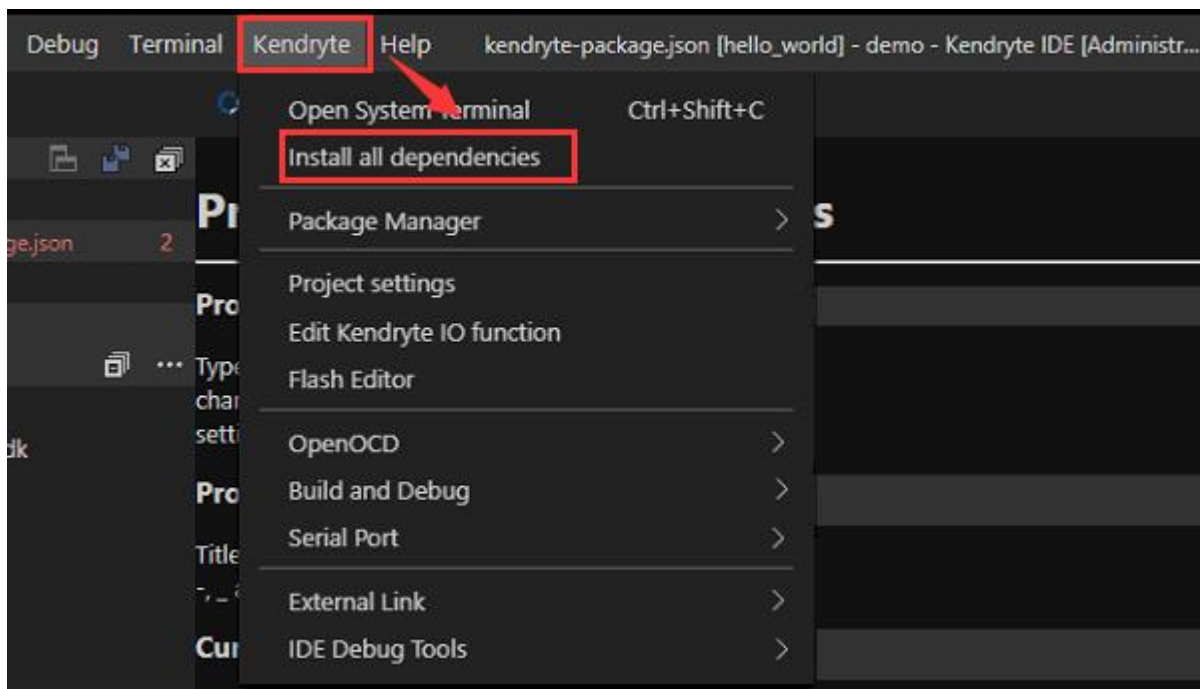
Only the main.c file is by default. The main() function is the entry point of the program, this program function is that print a "Hello World".
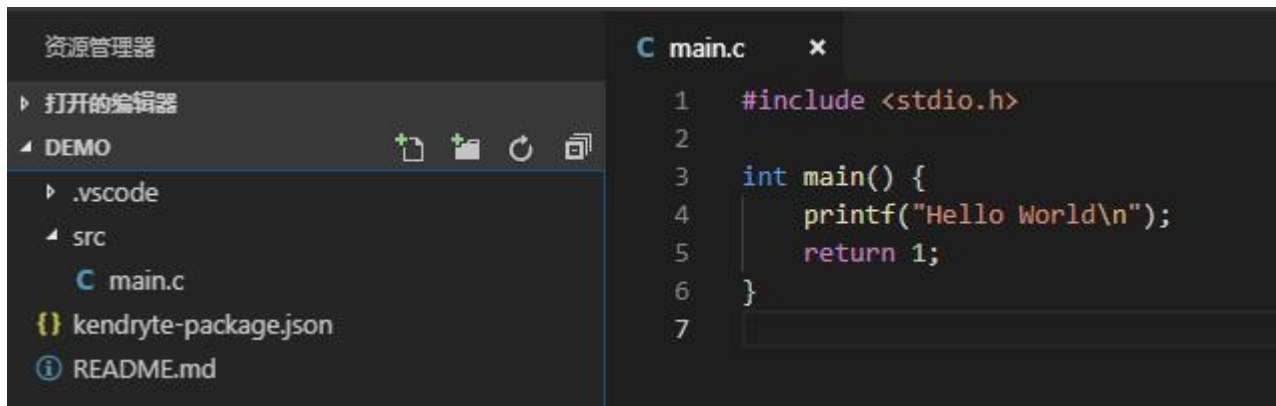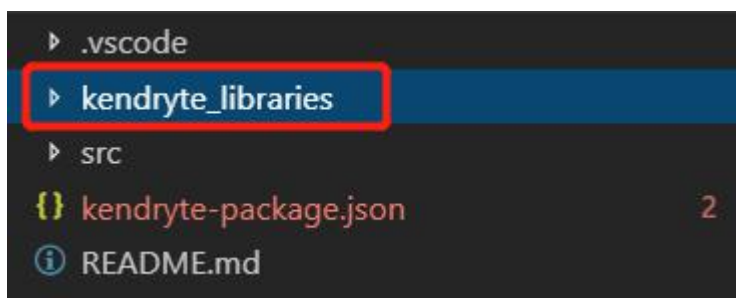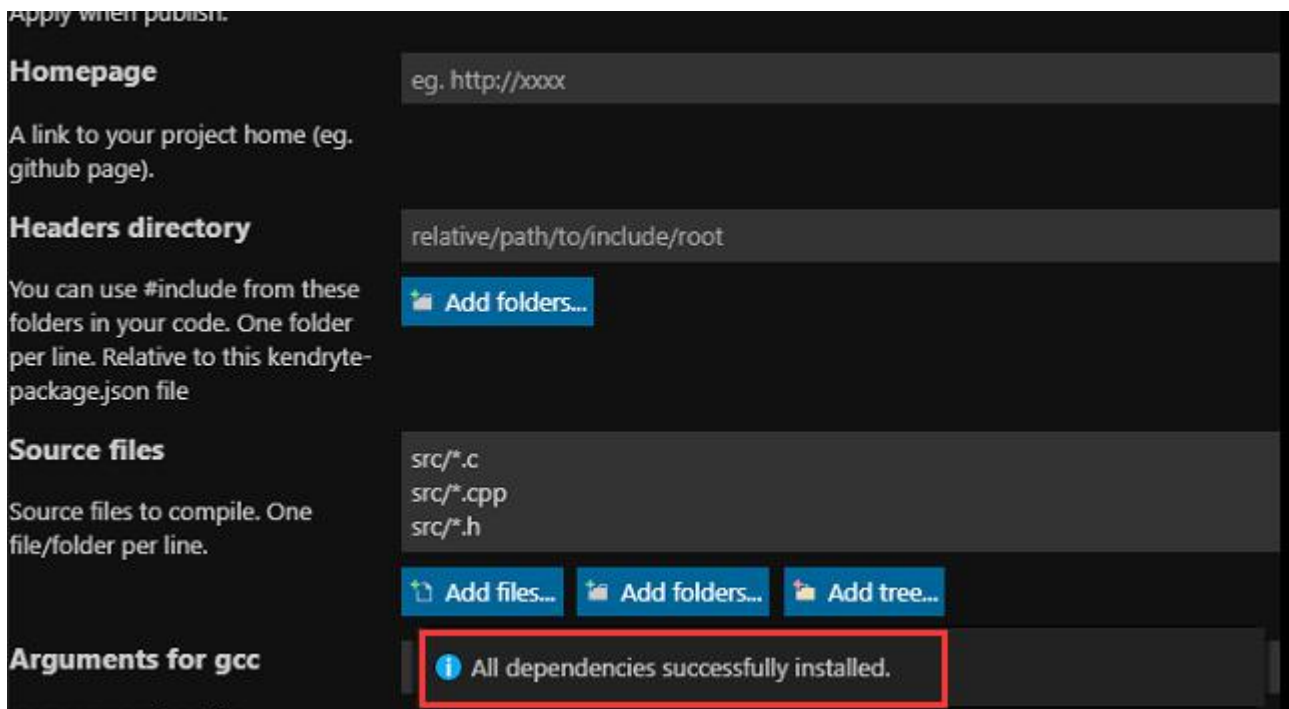
2.4 The **kendryte-package.json** file is the configuration file of KendryteIDE. You can modify the project name, add file paths, and add compilation parameters.
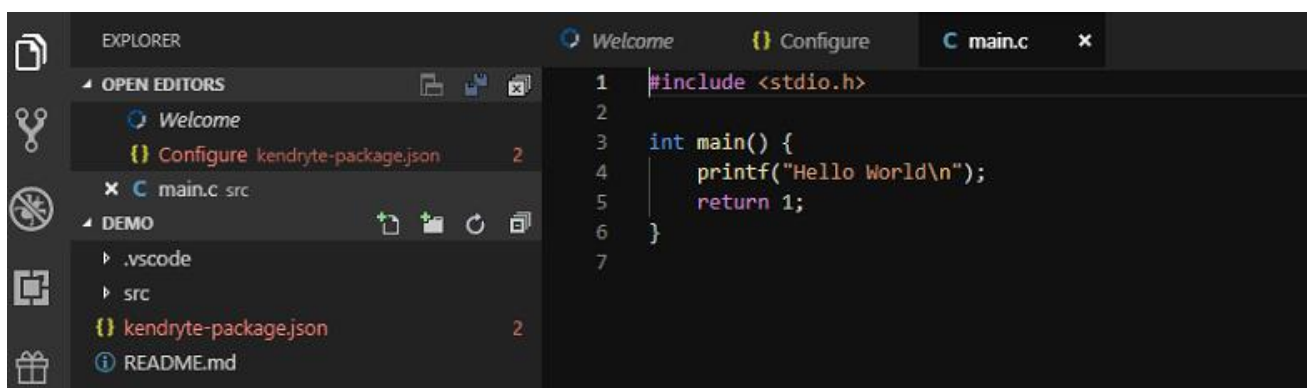
2.5 Click "Kendryte" --> "Install all dependencies" on the menu bar. As shown below.
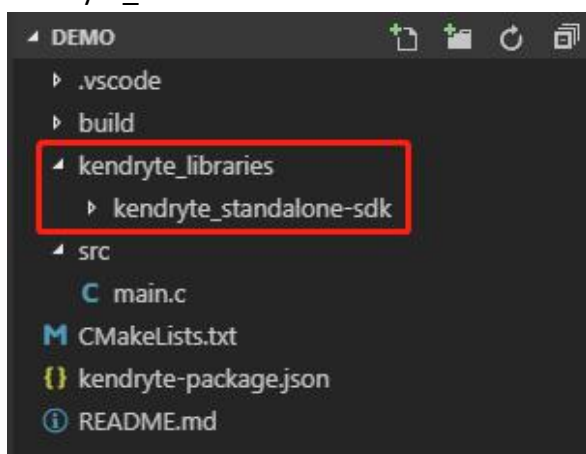

2.6 When the dependency installation is complete, we can see the prompt as shown below in the lower right corner.

Apply when publish.

Homepage

eg. http://xxxx

A link to your project home (eg. github page).

Headers directory

relative/path/to/include/root

You can use #include from these folders in your code. One folder per line. Relative to this kendryte-package.json file

📁 Add folders...

Source files

src/*.c
src/*.cpp
src/*.h

Source files to compile. One file/folder per line.

📄 Add files...   📁 Add folders...   📁 Add tree...

Arguments for gcc

ℹ All dependencies successfully installed.

▸ .vscode

▸ kendryte_libraries

▸ src

{} kendryte-package.json   2

ⓘ README.md

we can see that an additional **kendryte_libraries** folder.
This folder stores the K210 SDK, which contains various drivers.

2.7 Introduction of the kendryte_libraries folder structure.
The kendryte_libraries folder stores different K210 library files, the default is kendryte_standalone-sdk.

◢ DEMO   ⁺📄 ⁺📁 ↻ 🗗
  ▸ .vscode
  ▸ build
  ◢ kendryte_libraries
    ▸ kendryte_standalone-sdk
  ◢ src
    C main.c
  M CMakeLists.txt
  {} kendryte-package.json
  ⓘ README.md

If you need to download other library files, you can click "Kendryte" -> "Package Manager" ->
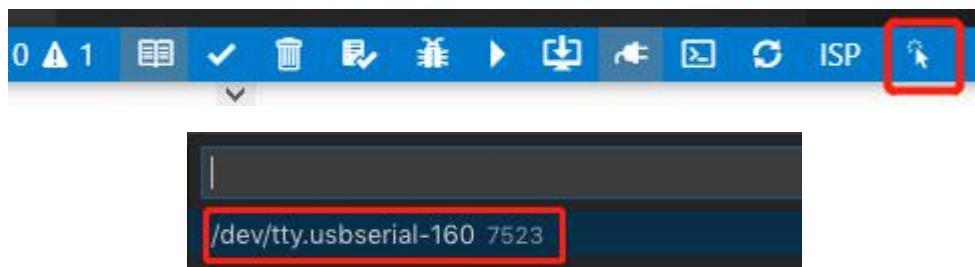
"Explore online packages" to download by yourself.
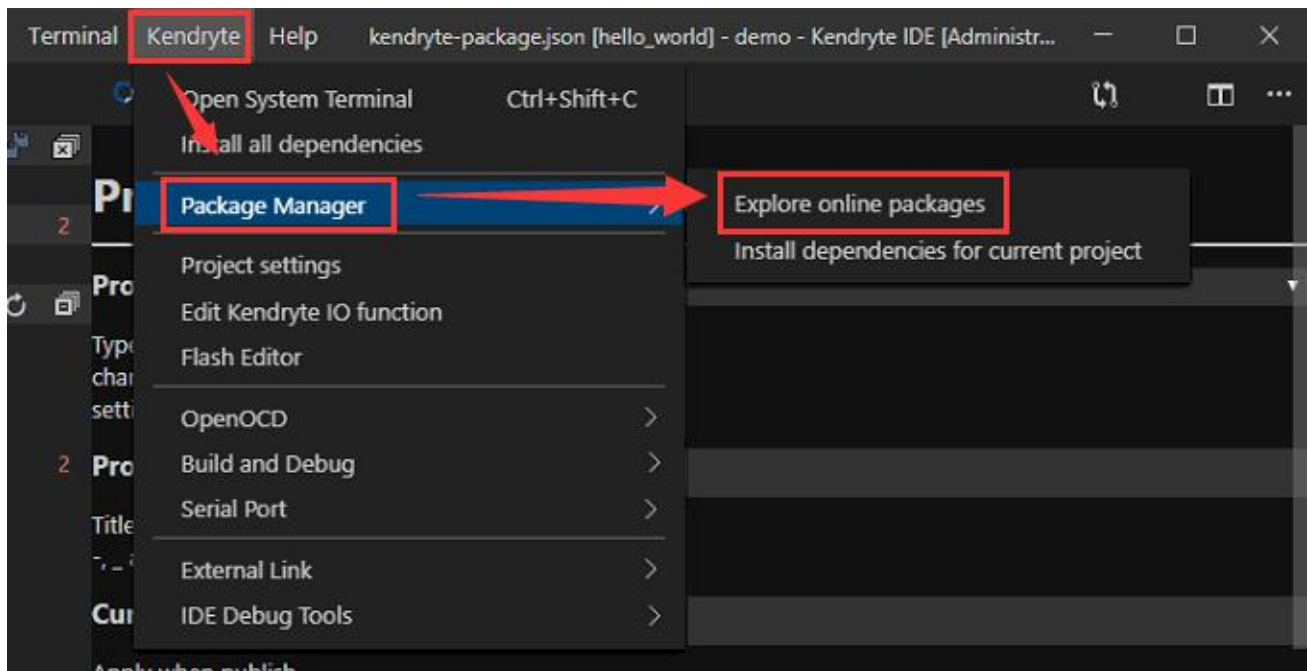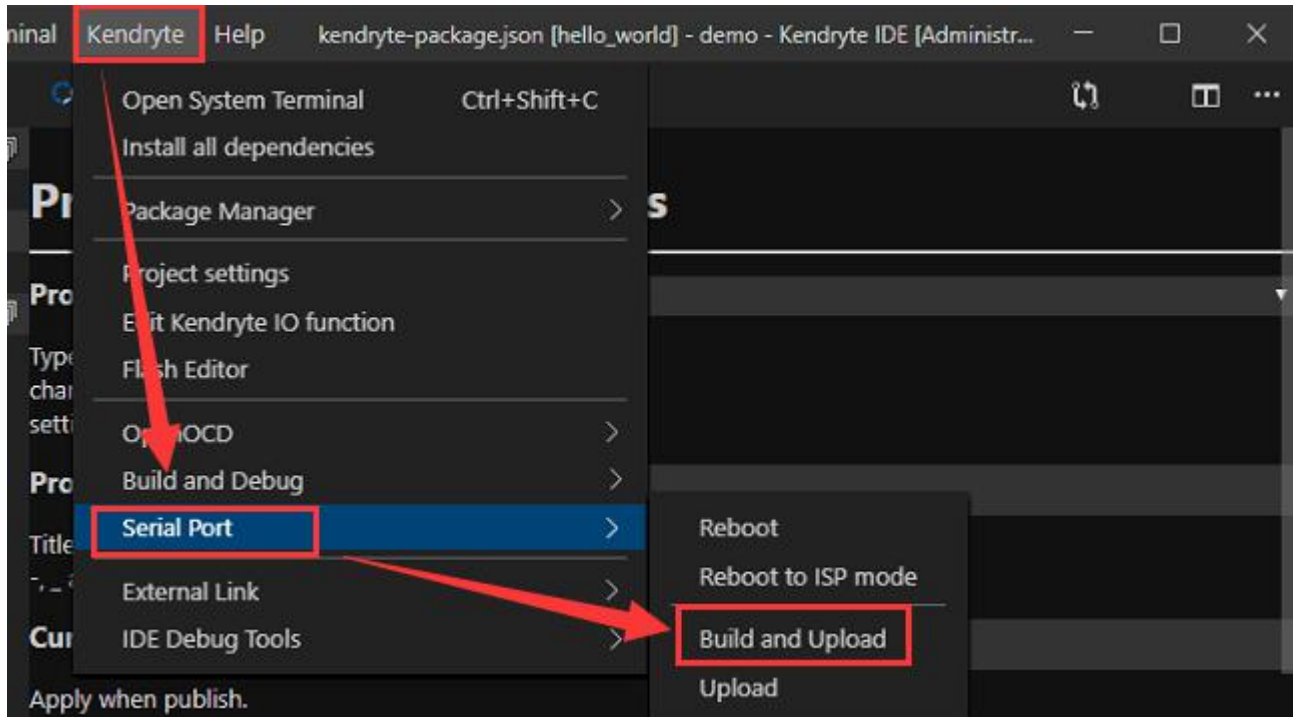

3.Compile and write program

3.1 Connect the K210 board to the computer, and **open the power switch of K210 board.**

Click the indicator button in the bottom column to select the serial port.

3.2 Build and Upload

Choose "Kendryte"--> "Serail Port" --> "Build and Upload".

3.3 At the first time, compilation will take a long time, please be patient.

After the compilation is completed, it will be directly uploaded to the K210 chip flash.

When you see the prompt as shown in the figure below, it means that the program has been successfully uploaded.
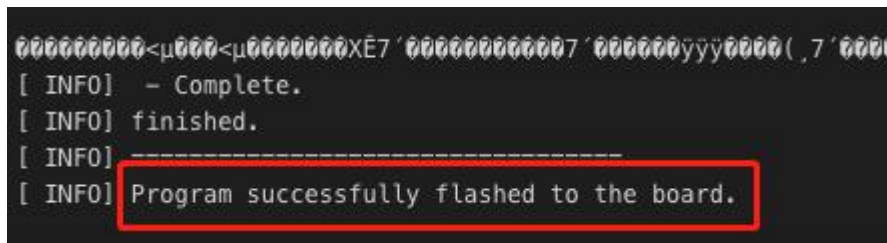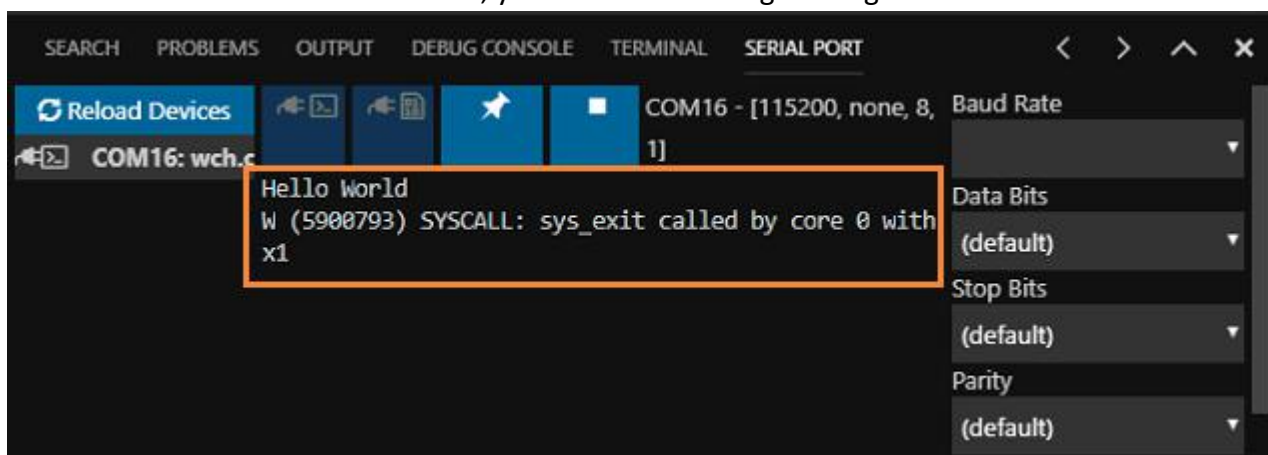

3.4 Open serial port, as shown below.

①Click "SERIAL PORT".
②Click "Reload Devices".
③Double-click "COM port"(my port is COM16, choose your own port).
④Click "Open serial terminal".
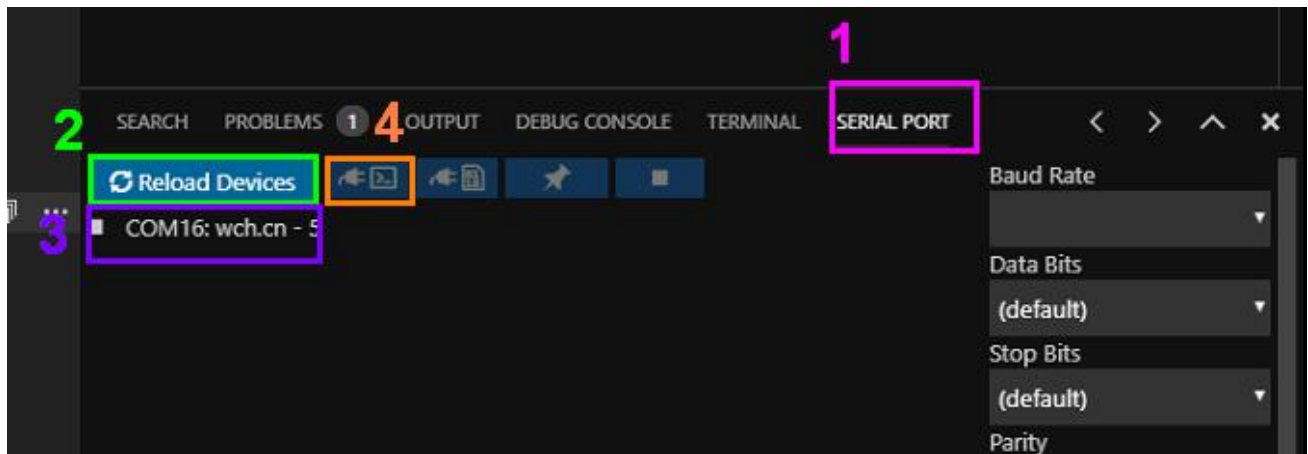
3.5 Press reset button on K210board, you can see following message.

!Tip: Before you compile and upload at the next time, you need to close port, Otherwise it will fail to flash.