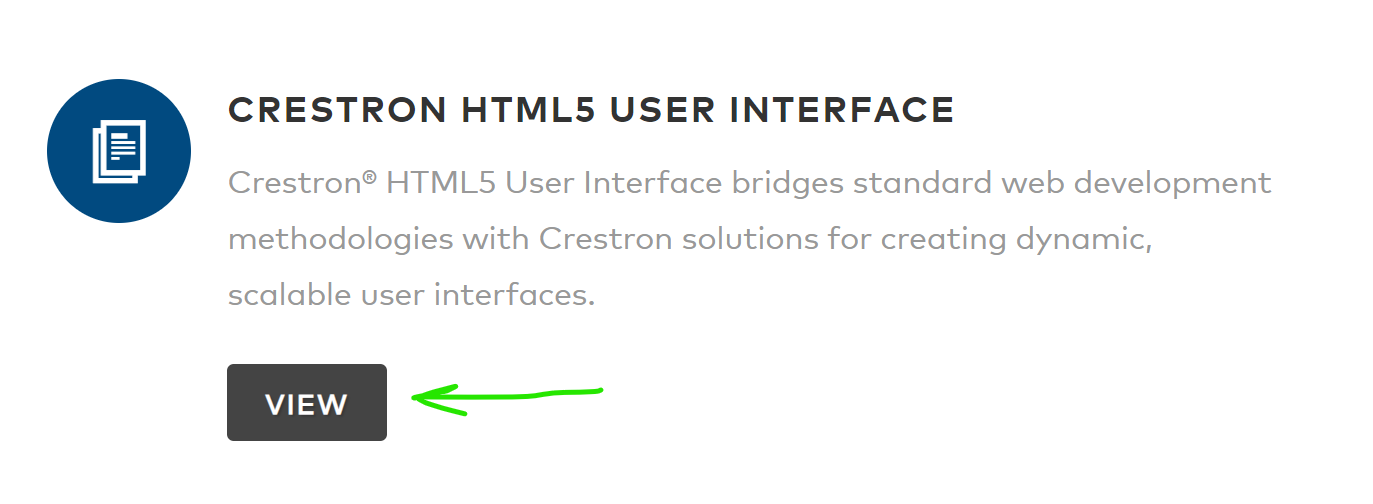
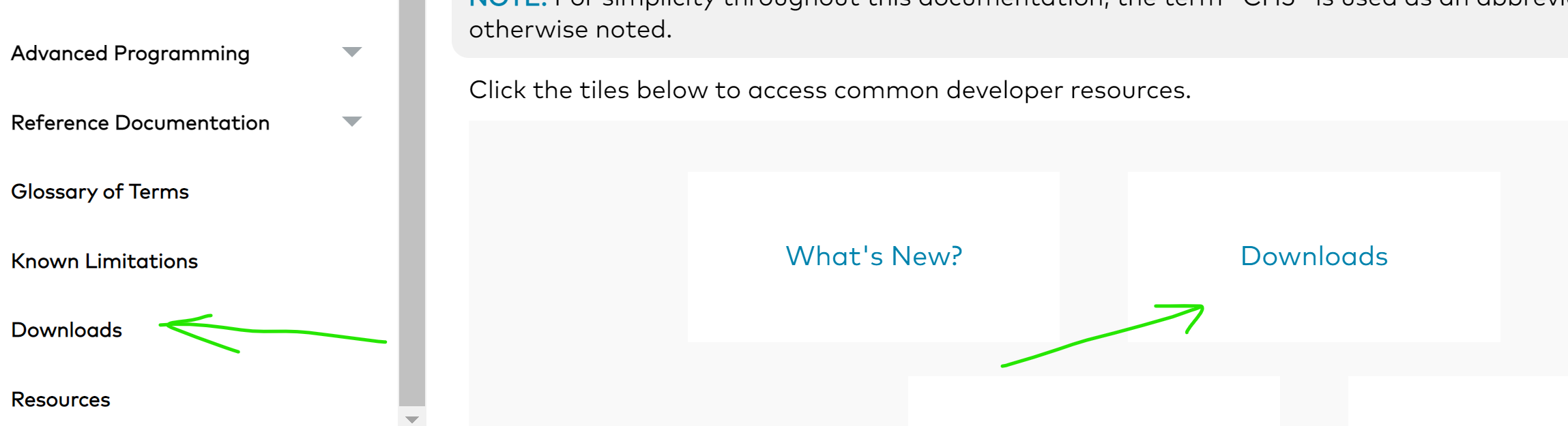
# Step 1a - contents

Unzip the zip file contents into a directory on workstation

# Step 1b – shell template

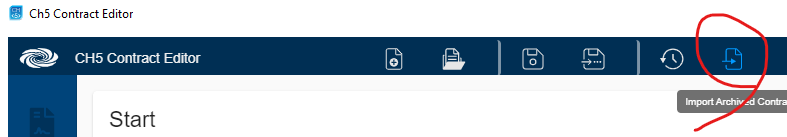
Download the latest [Crestron Template Project](https://siproducts.blob.core.windows.net/ch5-release/ShellProjects.zip) from developer.crestron.com

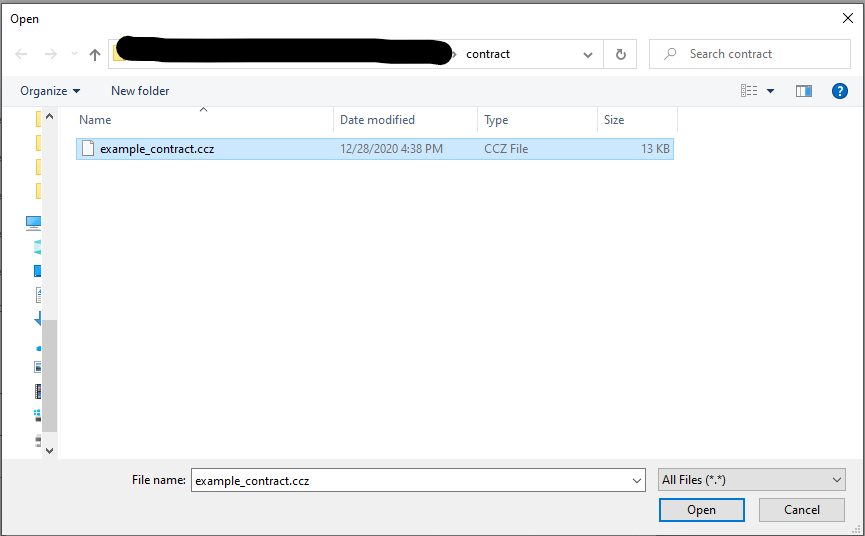
* navigate to Crestron HTML5 User Interface section
* 
* Navigate to downloads
* 
* Click on the “Crestron Template Project” link

Place the downloaded ShellProject.zip file to the project directory created in step 1a

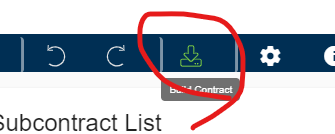
# Step 2 – Import and “build” the contract

1. Download, install, and run the Contract Editor Application <https://www.crestron.com/support/search-results?c=4&m=10&q=contract>
2. Import the .ccz file from the /contract directory



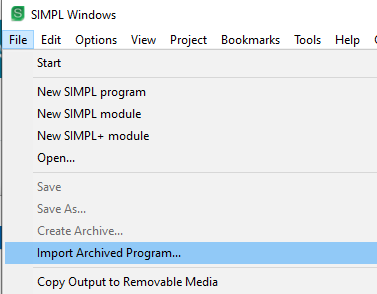


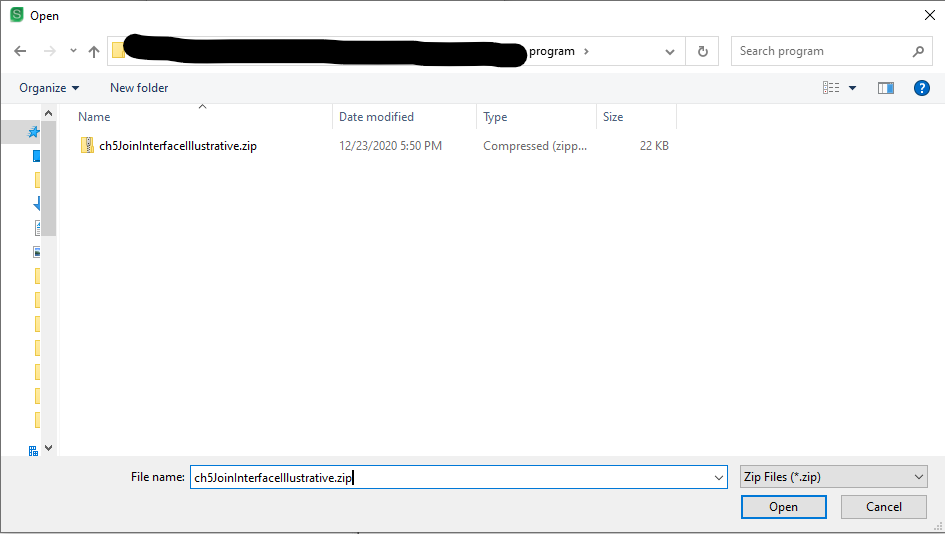
1. Build the contract

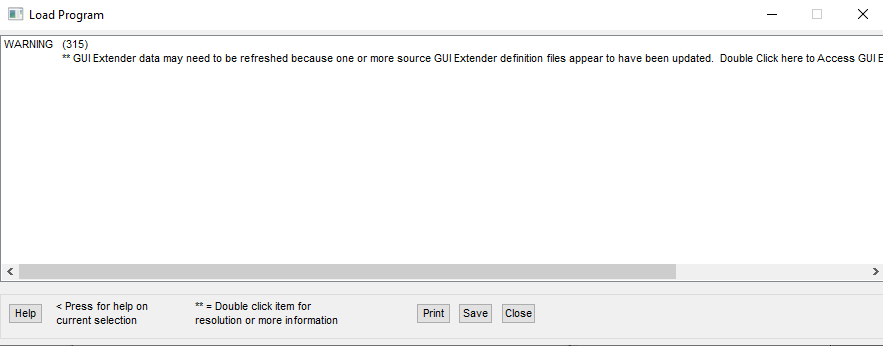


# Step 3 – import the SIMPL program, manage the extenders, build and deploy the program to a control system

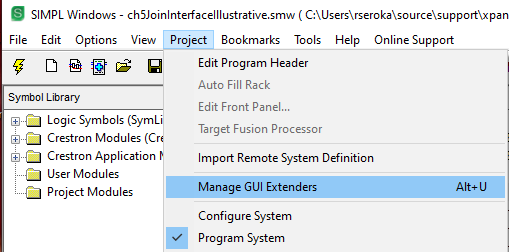
1. run SIMPL Windows
2. Import the SIMPL program, ch5JoinInterfaceIllustrative.zip, found in the /program directory

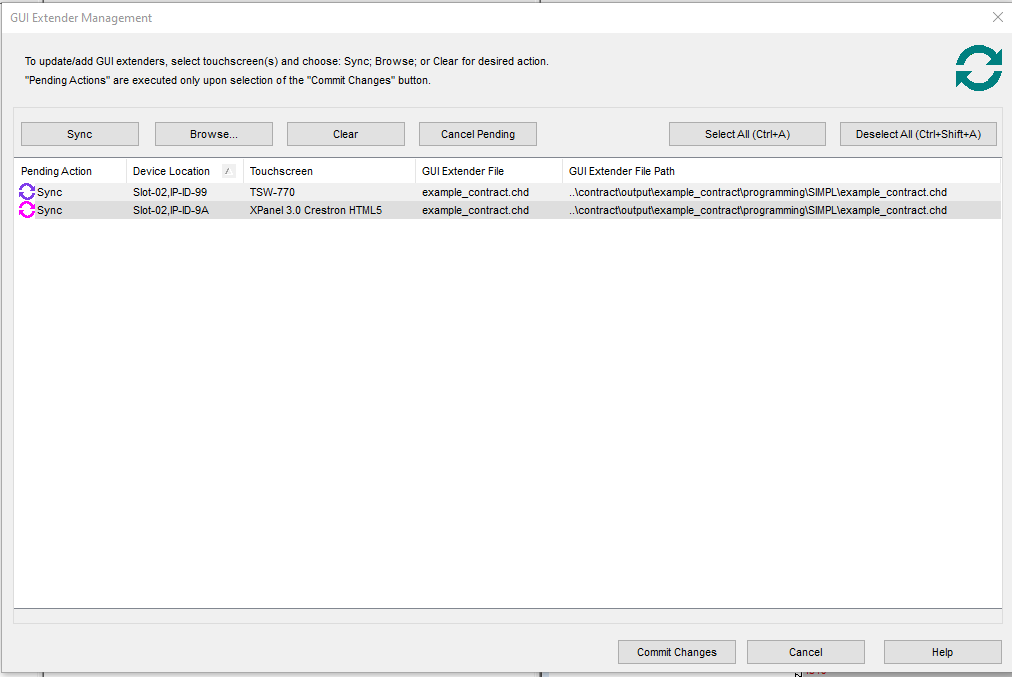


  
This result is expected

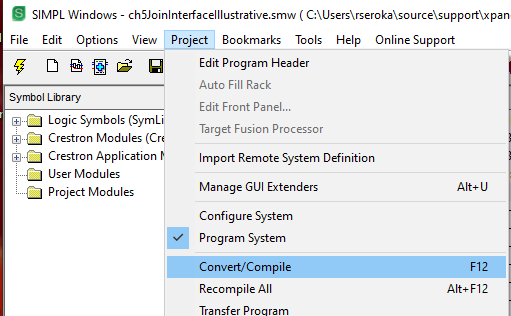


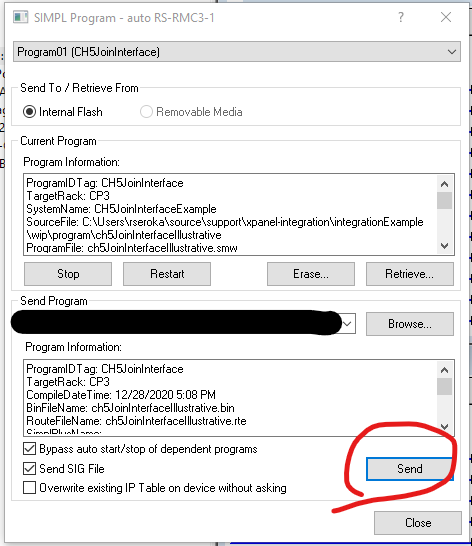
1. Browse to correct .chd file and “sync” the newly built contract output.





1. Send result of compilation to a 3 or 4 series control system that has been updated with appropriate level of firmware.

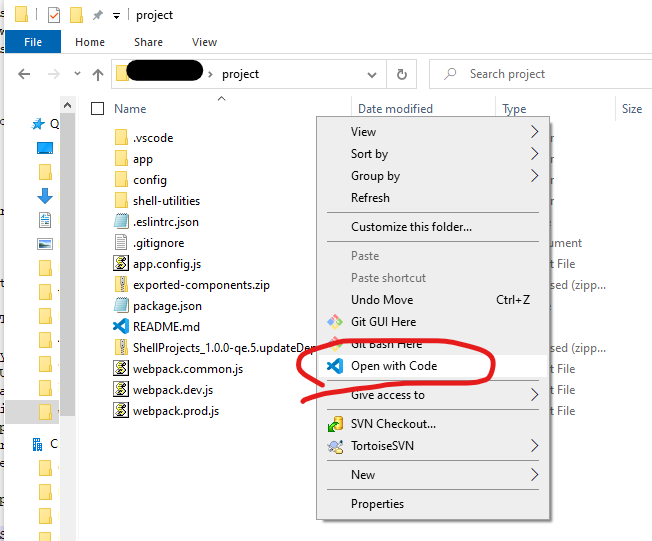




# Step 4 – update project with contract, import example components, build and deploy project

## Unzip the contents of ShellProjects.zip file to the project directory

## Open VSCode in the directory containing the contents



## Customize the project

### Optional – Customize the project name

In package.json file,

* update the “name”: parameter (example “name”: “shell-template” => “name”: “example-project”)
* update the “scripts”/”build:archive” : “….-p shell-template” to “build:archive” : “…. -p example-project”
* update the “scripts”/”build:deploy” : “…. dist/prod/shell-template.ch5z” to “… dist/prod/example-project.ch5z”

or simply search and replace shell-template to example-project in the package.json file

### Update the target host and type of device you will deploy the project to

Find the “scripts”/”build:deploy” entry.

Change the -H hostname to be -H ipaddress or hostname of control system you wish to deploy your web xpanel project to or touchscreen you whish to deploy your touchscreen project to

If deploy to a control system for a web xpanel deployment, change the -t parameter from touchscreen to web

Add a -p parameter to prompt for credentials,

Example:

    "build:deploy": "ch5-cli deploy -p -H 192.168.1.197 -t web dist/prod/example-project.ch5z",

see ch5-cli deploy help on CH5 developer.crestron.com for more information on how to use the ch5-cli for deploying projects

### Customize the location of the contract file interface

#### Update the archive to include correct file

Find the “scripts”/”build:archive” entry.

Update the -c parameter to point location of the built contract interface file.

../contract/output/example\_contract/interface/mapping/example\_contract.cse2j

Example (all on one line)

"build:archive": "yarn build:prod && ch5-cli archive -p example-project

-d dist/prod/Shell -o dist/prod

-c ../contract/output/example\_contract/interface/mapping/example\_contract.cse2j",

#### Optional – update the local copy of the contract if you are using the “yarn start”/”npm run start” interactive development

There are built in scripts to Interactive development is available through a local webserver serving up the project called “yarn start” or “npm run start”. If you use this feature, you’ll need to provide the contract by copying the ../contract/output/example\_contract/interface/mapping/example\_contract.cse2j file to the ./config/contract.cse2j file contents of the project.

## Install the dependencies

From integrated terminal in VSCode or external terminal in the same directory as the package.json file, enter the command “yarn install”

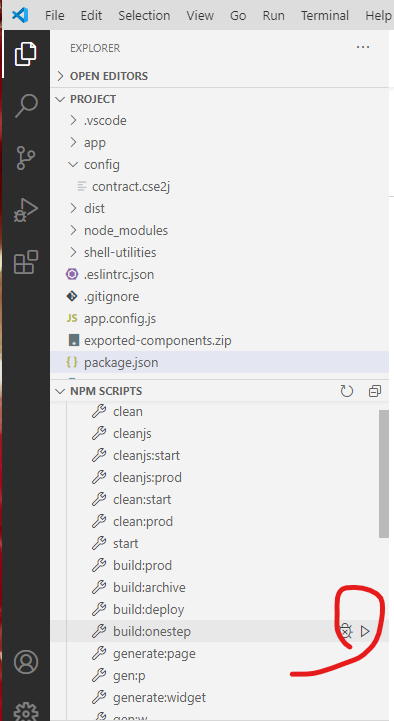
## Install example page components

Example pages are provided in the file called exported-components. To install run the following command line command from the integrated terminal in VSCode or external terminal in the same directory as the package.json file.  
“yarn import:components -z exported-components.zip –all”

When prompted “Do you wish to overwrite the files?” select Yes

## Build and Deploy

If you already updated the build:archive and build:deploy sections of the package.json file as directed in the “customize the project” section, you can now build and deploy the project in one script that is available by typing “yarn build:onestep” on the console or pressing the button in VSCode shown below.



## Run the project

### Touchscreen

If deployed onto a touchscreen, change the IPID of the touchscreen to 99

Example

TSW-1060>iptIP Table:CIP\_ID Type Status DevID Port IP Address/SiteName RoomId 99 Gway ONLINE 41794 192.168.1.179

### Web XPanel

If deployed onto a control system as a web xpanel project, use the URL

<https://controlsystem/example-project/index.html?ipId=9A>

where

controlsystem is the hostname or ipaddress of the control system, and   
example-project is the name of project deployed

note the ?ipId=9A is the ipId setup in the control system program for this touchscreen.

See documentation on how to configure within the project if desired.

Also note that control systems with self signed certificates will prompt that “connection is not private” or similar warnings. This is expected.

See CH5 documentation “Run the HTML5 Web XPanel” for more information.

