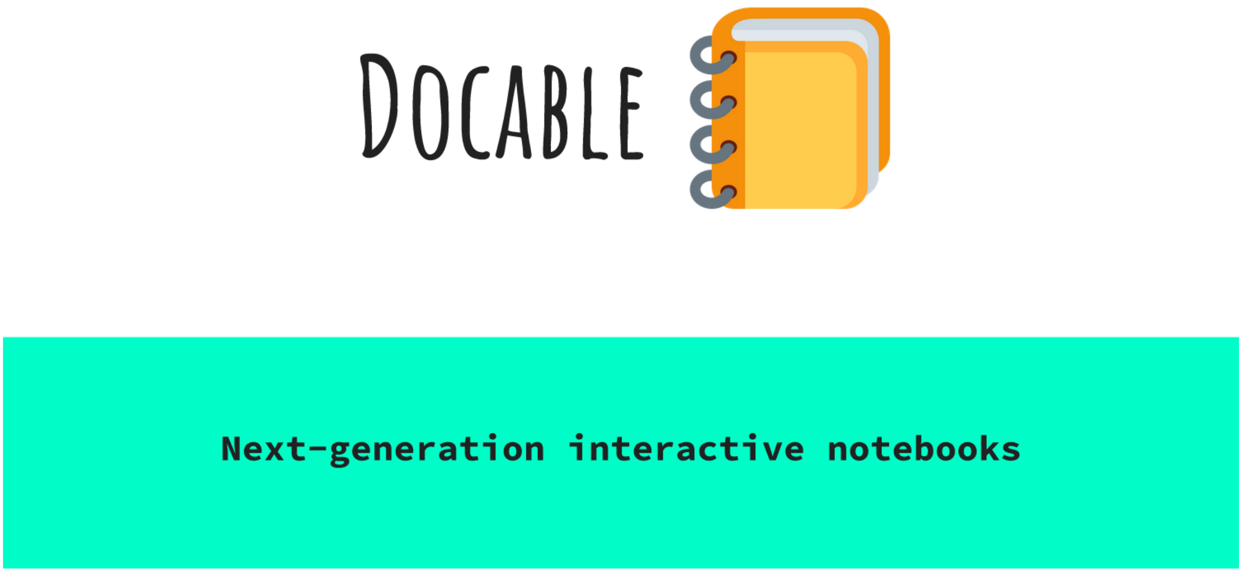
Basics / Getting Started node:12-buster

To successfully build software, you need a properly configured working environment. Creating consistent work environments for a small development teams can be difficult and frustrating, and even moreso for a large class of ~50ish students with various operating systems, personal configurations, coding editor preferences, etc. Thus, for the majority of technical workshops for CS5704 we use Docable, an online system built on Docker to create online educational notebooks, as the standard virtual work environment for everyone to complete activities. With Docable, you can write and run code, embed terminals, quizzes, and more! Here is a brief tutorial on how to use this interactive tutorial system.

Content

Docable notebooks contain items in **cells**. Cells are able to provide a wide variety of features, as we will see later, including embedded content to provide additional information to users like images, slides, or videos. Check out an example in the cell below:



Quizzes

Workshops may include short quizzes to assess your understanding of the content. Quizzes on Docable will consist of multiple choice questions that will notify you if your answers are correct or provide the right answer otherwise. Click the yellow CHECK button at the bottom to submit. Check out examples of quiz format below:

1. Question will be displayed here. Click Option 1.

● Option 1
○ Option 2
○ Option 3

Files

Docable allows users to edit and store files. Below is a sample text file. You may edit the contents of a file by clicking in the cell below and typing whatever you want. Try updating the **`/tmp/docable.txt`** file below. The file is saved automatically after editing.

File Editor `/tmp/docable.txt`

file is auto saved on edit

Scripts

Similarly, Docable is able to provide simple scripts for you to edit and run blocks of code. To edit, change the content within the cell. To run, click on the run button in the top right corner of the cell. The platform supports scripting in JavaScript, C, C++, Java, Python, Ruby, and bash. Modify the script below to print out "Hello Docable" on the first line and add a new line to print "Welcome to CS5704!" Running the script should generate the output below the file editor. Scripts with an error will print the error and StackTrace similar to running buggy code in an IDE or coding environment. Additionally, we are able to run some (very basic) checks to verify the output of your scripts with the yellow Check button at the bottom of the notebook (for example, if you didn't correctly follow the instructions above).

Live Code JS /tmp/script1.js ▷ run (F5)

```
console.log("Hello Docable")
console.log("Welcome to CS5704!")
```

Hint: You must run the script to create the file!

Terminals

Additionally, Docable also provides full interactive terminals for more extensive tutorial activities. These terminals are also connected to the shell commands and files generated throughout this tutorial. For example, run the following command in the terminal below to print “Hello World!” in a file called **/tmp/hello.txt** :

```
echo "Hello World!" > /tmp/hello.txt
```

Run the command **cat /tmp/hello.txt** to view the output.

Final Thoughts on Workshops

1. To receive credit, you will most likely have to submit screenshots or save a pdf of completed notebooks (most browsers have a Print > Save As PDF option in the print menu). Each workshop will have specific instructions on what to turn in.
2. The academic integrity policy applies to all workshops. Do not work with another student unless otherwise specified!

3. If something ever stops working, usually refreshing your browser or resetting the notebook (in the top panel) will fix it. If you run into any issues with a particular notebook, please contact the instructor.
4. Remember, anything covered in workshops (including the following notebooks) are eligible content to be on the exam!

Programming Languages ➔

Your score: 100

1 out of 1 quiz items answered correctly. Environment tests passed. ✓

```
test passed
```

CHECK