Guide to Using the IDE

by Sophia

WHAT'S COVERED

In this lesson, you will learn about the basic features of an integrated development environment (IDE) and be introduced to StackBlitz, a robust web-based IDE that will be used in this class. You will also learn about some of the benefits and advanced features of this IDE.

Specifically, this lesson will cover the following:

1. Integrated Development Environments (IDE)

There are different ways to write your code and run it when creating the program. For many languages, there is the option to use an Integrated Development Environment (IDE) to write the code. Reflect back to a previous lesson where you learned that an IDE is an application that provides a variety of tools and features that make software developers more efficient and accurate. IDEs work similarly to a basic text editor, but include additional features that aid in the coding and development process.

Individuals and organizations may have different preferences of which IDE they like to use. This course uses a web-based IDE called StackBlitz.com. In this lesson, we will cover the basic features that you will find in almost any IDE. Then, we will look at the benefits of web-based IDEs along with the IDE selected for this course.

HINT

StackBlitz.com will be the required tool (IDE) for completing this course. At the end of this lesson you will need to visit StackBlitz.com and create an account and start getting familiar with the interface and layout as we cover

the basic and advanced features.

2. Creating a StackBlitz User Account

HINT

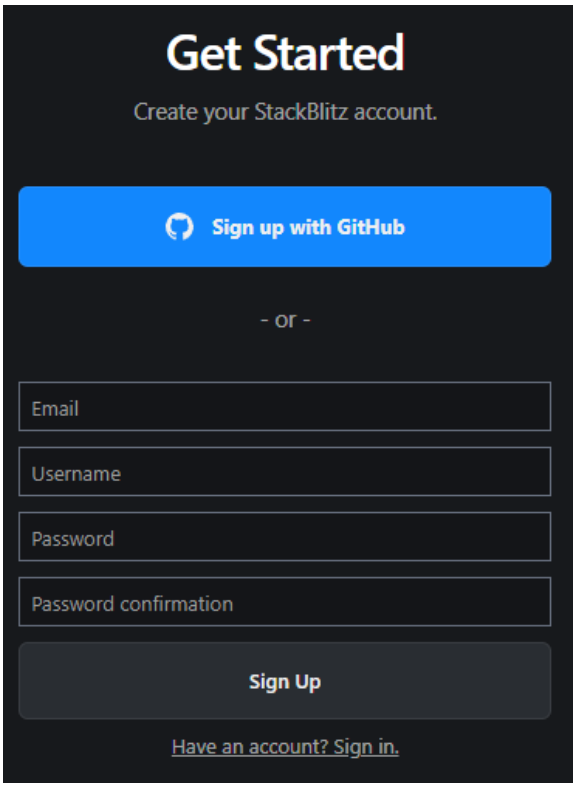
Even though you can use StackBlitz without an account, this use is designed for quick code testing only. You will need to create an account to be able to save projects and also share a project for grading.

STEP BY STEP

Step 1: To get started, access the StackBlitz.com site.

Step 2: To create a StackBlitz user account, select the "Get Started" button in the top right side of the window.

Step 3: Enter your email, a username, and password followed by selecting the "Sign Up" button to create your account.



HINT

We will not cover GitHub and it's uses. GitHub is not covered in this course, so it is highly suggested to create an account using your email.

Step 4: You will receive an email at the address you just created the account with. Make sure to look in your junk or trash if it does not appear in your inbox. The email will confirm your email address. Once your email is confirmed, return to the StackBlitz.com site and log in using the credentials you provided.

TRY IT

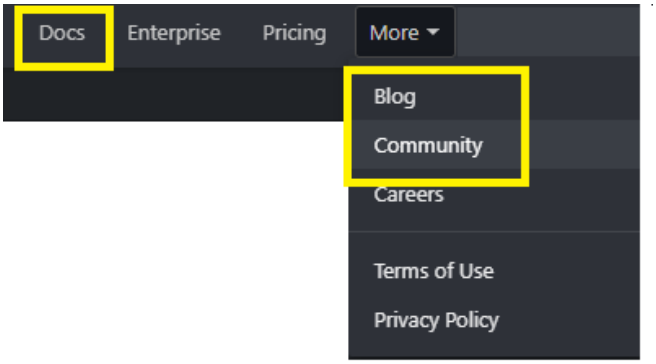
Now that you know more about StackBlitz, it's time to create your account if you have not already.

Directions:

1. Open your browser, preferably Google Chrome or Mozilla Firefox.
2. Visit StackBlitz.com and create a free account.
3. You should now be in your StackBlitz hompage.
4. Create a browser bookmark to the StackBlitz website to add it to your Web Development Toolkit.

REFLECT

As you are preparing to explore StackBlitz's IDE, know that there are helpful guides, a blog, and a community of users available. If you look at the top of the dashboard, there is a link to "Docs". This area houses all the help documentation. Using the drop down under "More", you have links to the blog and the community area. You can read up-to-date information from the engineers of StackBlitz in the blog and join the community to ask questions and share information.



3. Basics of the IDE

Once you have created your account and logged in, you will see your StackBlitz dashboard which is empty at the moment as you have not created a project yet. Once you do, your projects will appear in the middle area of the dashboard under "Recent projects".

A few things to note before we start a project, in the side menu you have a few options:

- **Dashboard:** will return you to the dashboard from any of these other option's screens.
- **Projects:** will list out all projects you have created both recent and long term.
- **Collections:** is where you can "collect" projects you create and/or other users projects that have been shared. Note: We will not be creating Collections in this course, but feel free to do so if you wish.
- **Settings:** lastly this an area that you could upgrade your free plan. Remember for this course, you only need to utilize the free plan. Your Profile is available to see/edit here as well. The Variables tab allows for GitHub connections, again we are not covering storage and GitHub use.

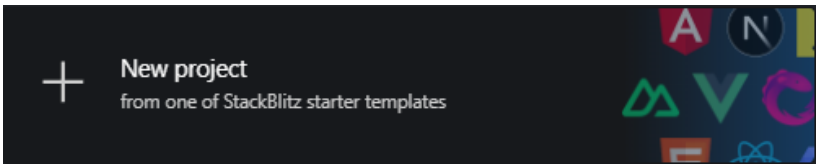
TRY IT

Now that you have an account, let's create a project and explore some of the features of StackBlitz.

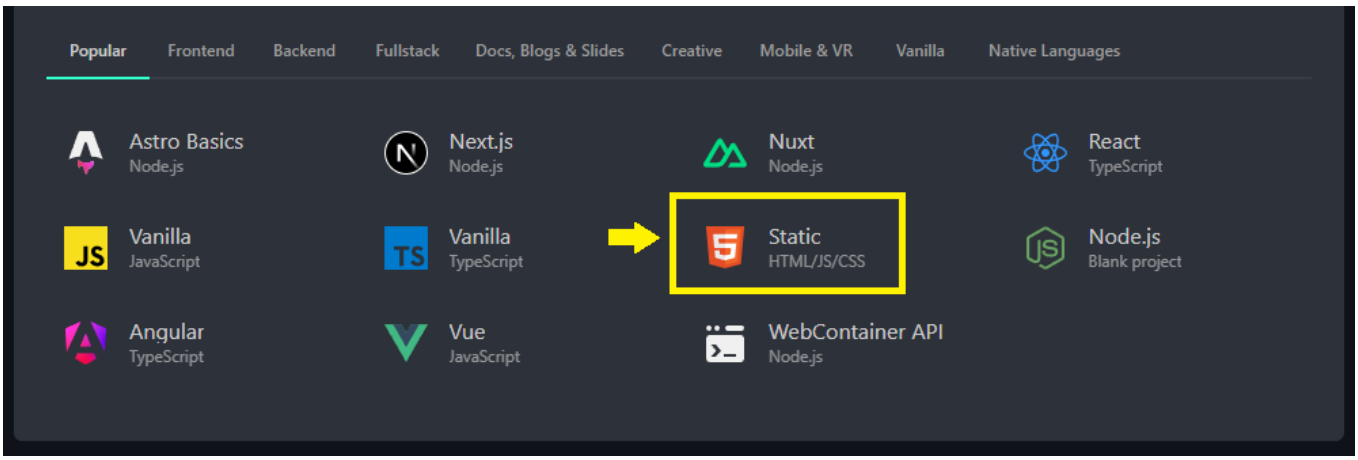
Directions:

Step 1: Log into your account if you haven't already done so.

Step 2: On the dashboard, select the "+ New Project" button.



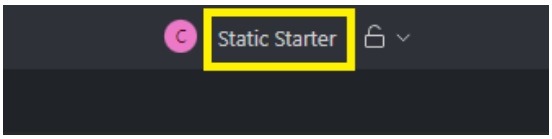
Step 3: In the "Popular" tab of the starter templates, select "Static HTML/JS/CSS".



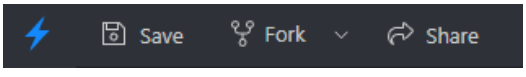
A new project is created. You will notice three distinct panels; we will discuss those in a minute.

First, you see that the HTML/JS/CSS project created contains some files and sample code as a starting point for a web project.

Also, by default, a random project name is generated (located in the top middle).



Step 4: Double click on the name and change the default name to something more meaningful to your project.



Next, there are four links across the top left side that you need to be aware of:

- **Dashboard (lightning bolt):** This link will return you to the main dashboard.

- **Save:** This link will save your project. If you return to the dashboard without saving, all your current work will be deleted. Remember to save your work routinely.
- **Fork:** We will not use this function. This allows a copy of a project, so any GitHub repository is not affected with changes to the code.
- **Share:** This allows you to gain a URL link to be able to share your project with others and the Sophia graders during Touchstone submissions.

Step 5: Now is a good time to save your project, select the Save link. Notice that once you save, if you do not make any changes, the Save link will be disabled. It will re-enable once you change anything. This is a good way to determine if you saved the project or not.

REFLECT

Now that you have your first project created and renamed, we can examine the StackBlitz interface where the work happens. Next let's take a look at the different basic components contained in most modern IDE's. There is nothing you need to do to this sample project other than just follow the rest of this lesson as we discuss each of the components. You will have plenty of time in future lessons to do some coding.

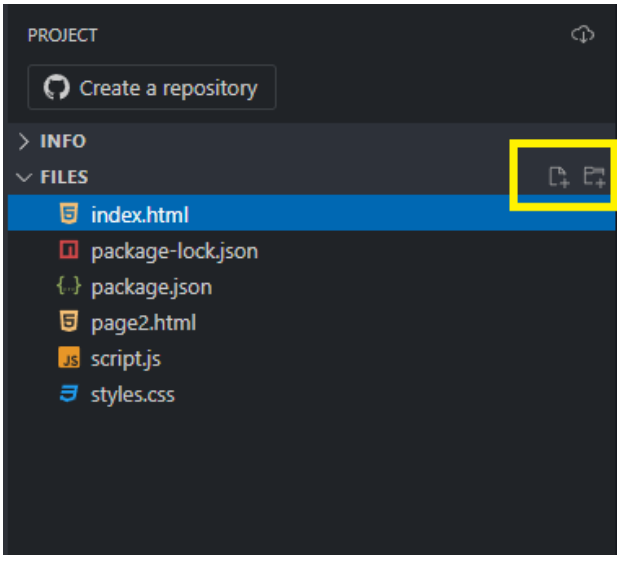
For now, let's cover the three main components you will be utilizing later:

1. Project panel
2. Editor panel
3. Output panel

3a. Project Panel

The first panel (the left panel) is called the project panel.

Here is where all the folders and files for your project exist. You can move between files by simply selecting them here.



For this course, we will be using the index.html, the script.js, and the styles.css files. It is important not to delete or change their file names. We will be adding code to replace the sample code contained in them as the course progresses.

Also notice the highlighted area in the image above. If you hover over the "FILES" row these icons appear. They allow you to create a new file or folder. While folder creation is not required during the course, creation of a new file will be utilized later in the course.

Any files located in this directory are included in the project and can be used or accessed directly. In addition to creating new files, you can delete or rename files and create sub directories or folders to organize the files. As this is an online web-based IDE and the file directory exists on the StackBlitz.com server, if you need to get a file from your local computer into the project, you can drag and drop to this area to upload a file or entire folder to the project. You can also download the entire project directory to your local computer using the small cloud with down arrow icon in the top right side of this panel.

At the bottom of the project panel is an icon that is a light/dark circle. This icon allows for the toggling of light and dark mode. This does not change the third panel (output) however, that is controlled by the code and is light by default.

Note: The code shown in examples in the course were taken from dark mode. In dark mode, the background uses dark colors, and the text and buttons are light. Dark mode is a preference of most software developers as it reduces eye strain and fatigue. Most modern Integrated Development Environment (IDEs) and code editors have some form of dark mode that all new developers are encouraged to utilize.

3b. Editor Panel

The main panel (the middle panel) is the editor panel, as you can see below some sample code was create for us.

index.html

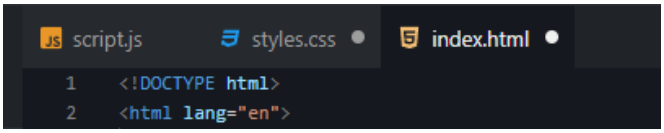
```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Home</title>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width" />
    <!--
      Need a visual blank slate?
      Remove all code in `styles.css`!
    -->
    <link rel="stylesheet" href="styles.css" />
    <script type="module" src="script.js"></script>
  </head>
  <body>
    <nav>
      <a href="/" aria-current="page">Home</a>
      <a href="/page2.html">Other page</a>
    </nav>
    <main>
      <h1>Home page</h1>
    </main>
  </body>
</html>
```

The editor panel contains colorized text signifying what are properties, variables, numbers. etc. This is the main panel and where you will do most, if not all, of your coding. IDE code editing screens have the same features of a basic text editor as well as additional features such as auto code completion, code suggestions, multi-cursor, and advanced find and replace tools. Additionally, much like the spelling and grammar error tools in Microsoft Word, IDE code editors will examine your code and identify syntax and logic errors using underlines. Hovering your mouse over one will open a pop-up containing a description of the problem and in good IDEs may include a suggestion as to how to fix the issue.

Some IDEs also allow you to have multiple code files open at once. You can have multiple files open in tabs and easily swap back and forth between the tabs. This is a helpful feature when working in web development as we often have code that interacts directly with each other but are abstracted out into their own files, as is often done with HTML, CSS, and JavaScript.

Note: When you open a file in the Project panel, it will open in a separate tab at the top of the editor panel.

EXAMPLE



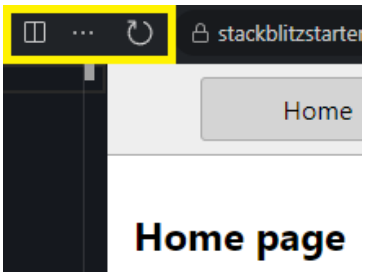
KEY CONCEPT

Included in the editor panel but below the editing area is the Terminal information. This area identifies status, connection to the server information, and can present project errors that may occur across files.

3c. Output Panel

The output panel is the panel that shows you the results of your code.

IDE output typically comes in two forms, HTML content rendered to a page or textual content posted to a console. The StackBlitz IDE renders HTML to a page. Any changes that you make in the editor panel will automatically change in the output. This makes testing easy as you can see the effects of code adjustments live and in the panel to the right as you make the adjustments.



Located near the top of the output panel are three buttons:

- Split Editor (two panes): this button allows you to open more than one file at a time in the editor panel.
- More Actions (triple dots): this button allows you to close all active (open) files in the editor panel. It can also close only the saved files.
- Refresh (circular arrow): - you will use this button in this course. Certain code changes are displayed automatically in the output panel. If you are working on a supporting page or in the .css/.js files, you may need to select the refresh button to see the changes take place in the output panel.

3d. Other Basic Features

In addition to the main interface components, some of the basic functions of an IDE include:

Function	Description
IntelliSense	IntelliSense is a feature of modern IDEs wherein the IDE will continuously examine your code to identify problems. The IDE does this not just for the code file being

	worked on, but does this for the entire project. Basically, the IDE is pre-rendering and pre-executing your project in the background while you work. Not only does this allow it to easily identify syntax errors, but it allows it to identify logic errors and access violations almost in real-time.
Error messages	Error reporting is a key feature of an IDE used for software development. When the IDE detects an error through IntelliSense or when attempting to execute the code, the IDE will produce an error message visible in the Terminal area of the Editor panel. No programmer is perfect and having the IDE constantly looking over your shoulder is a great advantage.
Code Suggestion / Auto Completion	Another benefit of IntelliSense is that it allows the IDE to predict what your next step will likely be and will provide a suggestion of how to complete the code. This can be anything from completing a single word, to building out a complete code structure based on what you did previously.
Running the code	IDEs are typically equipped to execute or run the code project. This includes having the necessary libraries and applications to compile the code and show its results. Many times, an IDE will provide additional helpful insight and metrics into the program's performance when executing the code.
Debugging the code	An IDE should provide the tools needed to debug the code. Generally, debugging is the process of carefully examining the code and how it executes to identify and resolve problems or errors with the code. When debugging, the IDE executes the application, but keeps the execution closely tied to the actual code files and IDE. This way, while the program is running, you can examine the contents of the program's data in RAM, step through the program one line of code at a time, some IDEs even allow you to manually manipulate data in RAM to see the results.

In addition to lots of other smaller features, IDEs provide a wide range of tools and utilities for making developers more efficient and more effective. However, the realm of web development encompasses a wide range of technologies that have to work together seamlessly. As such, StackBlitz IDE provides a set of advanced tools that enable developers to be even more effective at prototyping, testing, and deploying web software.

HINT

Only if you feel comfortable, feel free to test the functionality of Stack Blitz's IDE. Remember, if you make a mistake with your code, you can use the shortcut ctrl + z (or command + z if on an Apple device) to redo anything that was changed.

MAKE THE CONNECTION

You will create a need to have a StackBlitz account to work on Touchstone Task 2.1: Creating HTML Pages. Now is a great time to create your account. Don't forget to add the IDE to your developer toolbox!

SUMMARY

In this lesson, you learned about the basic features of a modern **Integrated Development Environments (IDE)**. You also learned about how to get started by **creating a StackBlitz user account**. Lastly, you learned the **Basic tools of an IDE** while reviewing the essential panels of the StackBlitz IDE (**Project, Editor, Output**) and identifying **other basic features** that exist with most modern IDEs.

Source: This Tutorial has been adapted from "The Missing Link: An Introduction to Web Development and Programming " by Michael Mendez. Access for free at https://open.umn.edu/opentextbooks/textbooks/the-missing-link-an-introduction-to-web-development-and-programming. License: Creative Commons attribution: CC BY-NC-SA.