



Machine and Environment Setup

.NET CORE

**“Failure is central to engineering.
Every single calculation that an
engineer makes is a failure
calculation. Successful engineering
is all about understanding
how things break or fail**

”
– Henry Petroski

Contents

[Chat Platform](#) – Slack. This is where we will collaborate and organize.

[Code Editor](#) – Visual Studio Code and Visual Studio. This is where we will create and test code.

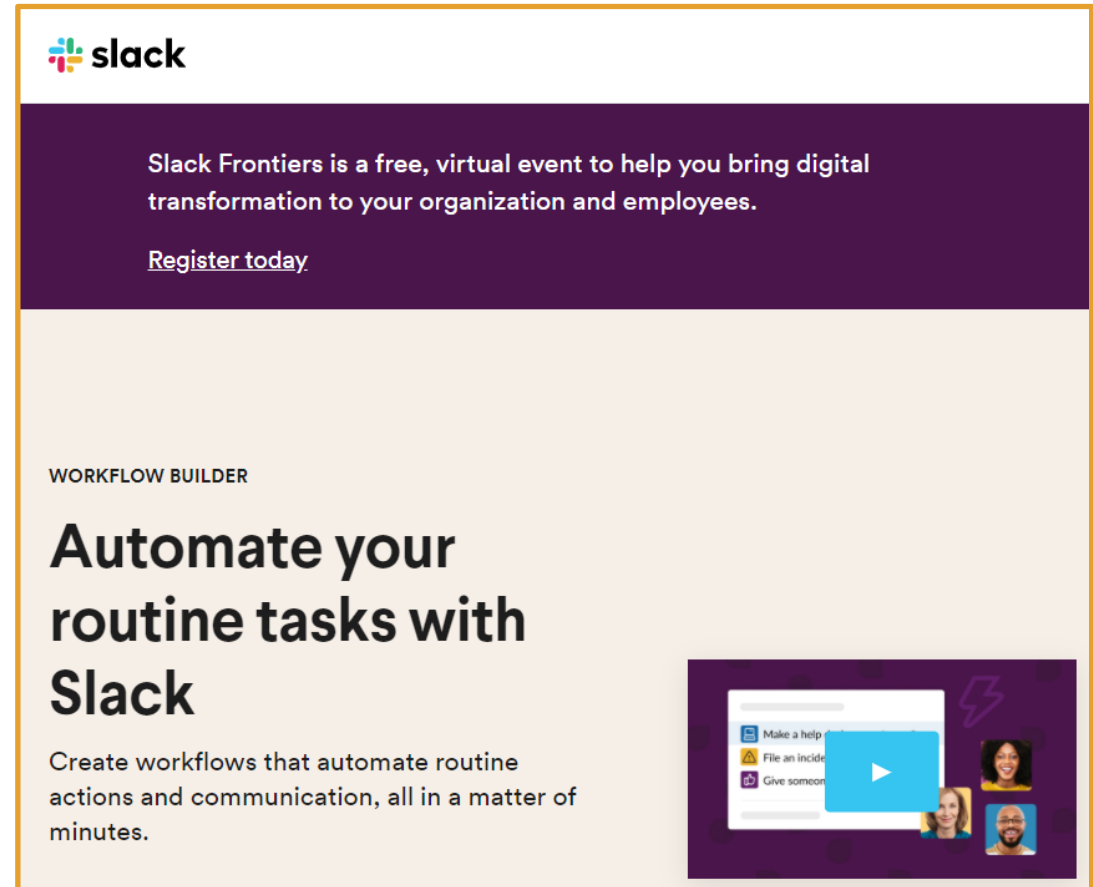
[SDK \(Software Development Kit\)](#) – This is the suite of programs that we download that enables us to create, edit, and test code.

[Version Control](#) – GitHub. This is where we will store our documents and code, record changes, and control which version of our code we use.

Chat Platform – Slack

<https://slack.com/>

1. Download Slack to PC and phone
2. Each associate is responsible for maintaining contact and being up to date on messages on Slack. Especially during 9-5 (CST) hours.
3. Get everyone into the Class workspace. Messages disappear after a few days so copy/download what you want to keep.



The image is a promotional banner for Slack. At the top left is the Slack logo. Below it, on a dark purple background, is the text "Slack Frontiers is a free, virtual event to help you bring digital transformation to your organization and employees." followed by a "Register today" link. The bottom half of the banner has a light beige background. It features the text "WORKFLOW BUILDER" in small caps, followed by the headline "Automate your routine tasks with Slack" in large, bold font. Below this is the text "Create workflows that automate routine actions and communication, all in a matter of minutes." In the bottom right corner, there is a small graphic showing a workflow builder interface with a play button and three user avatars.

slack

Slack Frontiers is a free, virtual event to help you bring digital transformation to your organization and employees.

[Register today](#)

WORKFLOW BUILDER

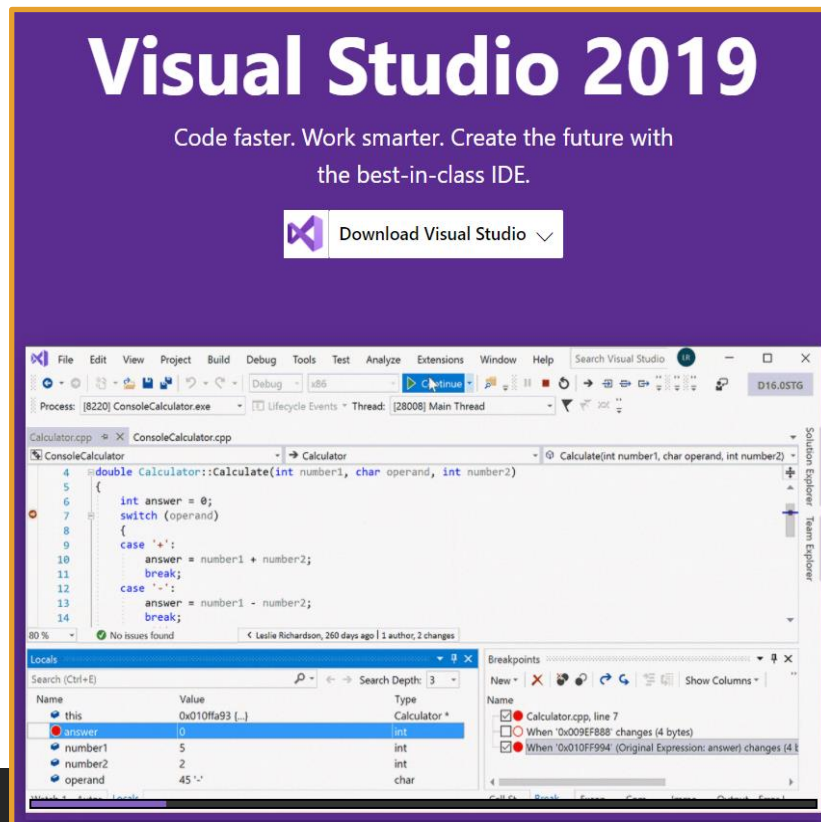
Automate your routine tasks with Slack

Create workflows that automate routine actions and communication, all in a matter of minutes.

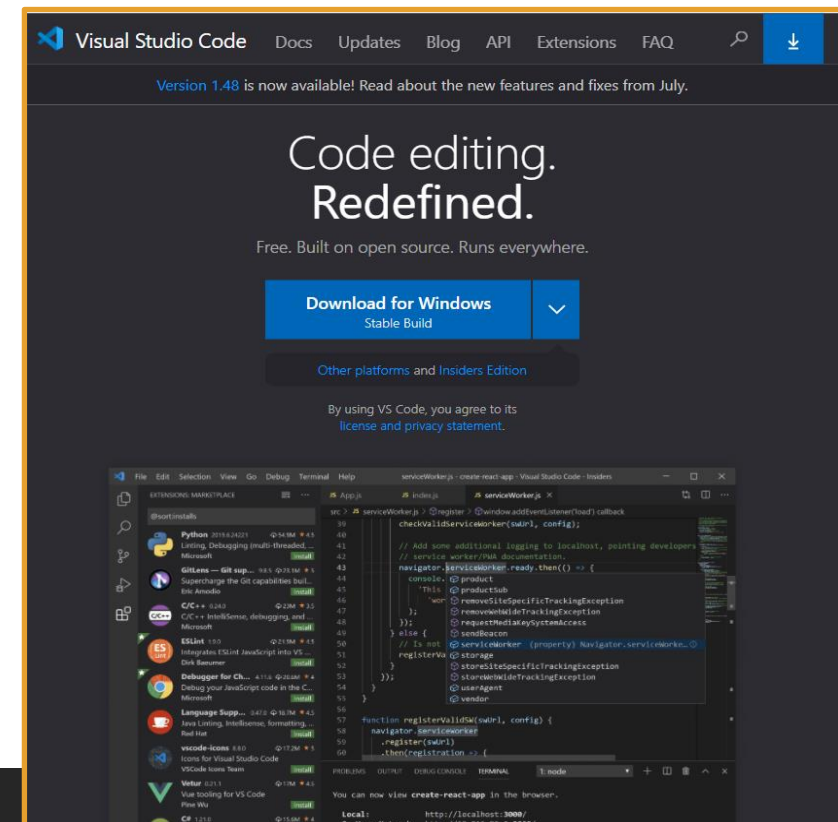
Code Editor – Visual Studio

<https://visualstudio.microsoft.com/vs/>
<https://dotnet.microsoft.com/download>
<https://code.visualstudio.com/>

Visual Studio



VS Code



SDK (Software Development Kit)

https://en.wikipedia.org/wiki/Software_development_kit#:~:text=

A **Software Development Kit (SDK)** is a collection of software development tools in one installable package. They have compiler, debugger and perhaps a software framework. They are normally specific to a hardware platform and operating system combination.

- An **SDK** is required for developing a platform-specific app.
- The development of an Android app on Java platform requires a Java Development Kit.
- For iOS applications (apps) the iOS SDK is required.
- For Universal Windows Platform (UWP) the .NET Framework SDK is used.

Some **SDKs** add additional features and can be installed in apps to provide analytics, data about application activity, and monetization options.

.NET Core

.NET Core 3.1

.NET Core is a cross-platform version of .NET for building websites, services, and console apps.

Run Apps ⓘ

[Download .NET Core Runtime](#)

Build Apps ⓘ

[Download .NET Core SDK](#)

Advanced ⓘ

[All .NET Core downloads...](#)

gitBash

<https://gitforwindows.org/>

1. *Git for Windows* focuses on offering a lightweight, native set of tools that bring the full feature set of the [Git SCM](#) to Windows while providing appropriate user interfaces for experienced Git users and novices alike.
2. Go to <https://gitforwindows.org/>



Version Control – GITHUB.COM

<https://gitforwindows.org/>

1. Open Command Line (Terminal). Run `'git -version'`.
2. Accept your invite to the class repo.
3. On the class repo, create a personal Repo of the format `'MooreMark'`.
4. Clone your remote repo from your local gitBash.
5. Create a text doc in your cloned repo folder.
6. Push a simple text doc.
 - `git add .`
 - `git commit -m "message to self"`
 - `git push`
 - Create new upstream branch, if necessary.
 - Verify it worked by looking at your account online.



GitHub

Basic Git Life Cycle

<https://education.github.com/git-cheat-sheet-education.pdf>

1. `git clone [url]` or `git pull` from master branch.
2. `git checkout -b [branchName]` (Create a new “feature” branch and switch to it.)
3. Make changes to add a feature.
4. `git add .` (Add all changes made to tracking)
5. `git commit -m “This message tells what the changes are”` (Add to staging)
6. `git pull` (to be 100% sure no changes were made to the master branch while you were working)
7. `git push`
8. Go online and make a pull request (PR) to master.

Simple (NO-CONFLICTS) Github Workflow

