# EiABC
# Professional Computer Aided Design II
Module 2_Lect_2a

The following lecture note and exercises are prepared by the readings of the instructor and references to www.Quest pond.com; web dev simplifier; traversy media, ….

Students may access the exercise files from the exercise file link from their class content site: link
Students are advised to use exercise files as per their instructor's directions.

## Contents

# 1. VS Code and Visual Studio Code


Visual studio Code is a development environment that has different functions integrated together not just editing text like VS code, Eclipse, Bracket, Sublime Text, etc. WordPad also serves as text editor when you are working on HTML. Since you will be working on SVGs, you will want to learn about environments like VS code or Visual Studio Code.

## Explore VS code editor

Vs code editor is created by Microsoft. It is a cross platform code editor which can run on Linux, mac and windows. Things we need for an enterprise development can be the following: Code editor, method of debugging, webservers, databases, code review tools, repository, text tools, etc.

Going through the following steps shall help the student to have a grasp on how to use VS code:(1.1)Get VS code form the web and install it.(1.2)Create a project folder (1.3) Understanding explorer and open editor.(1.4)Extension support for third party. (1.5)Integrated terminal (1.6) Reveal in explorer features.(1.7)VS code preferences.(1.8)Left tool bar.(1.9)Configuring task in VS code.(1.10)Going back and forth, Zoom in, Zoom out.

### 1.1. Get VS code form the web and install it

Vs code editor Get VS code from this link: https://code.visualstudio.com/docs/?dv=win and install it. There is a difference between VS code and Visual Studio Code. The former is just an editor while the latter is IDE (integrated Development Environment).

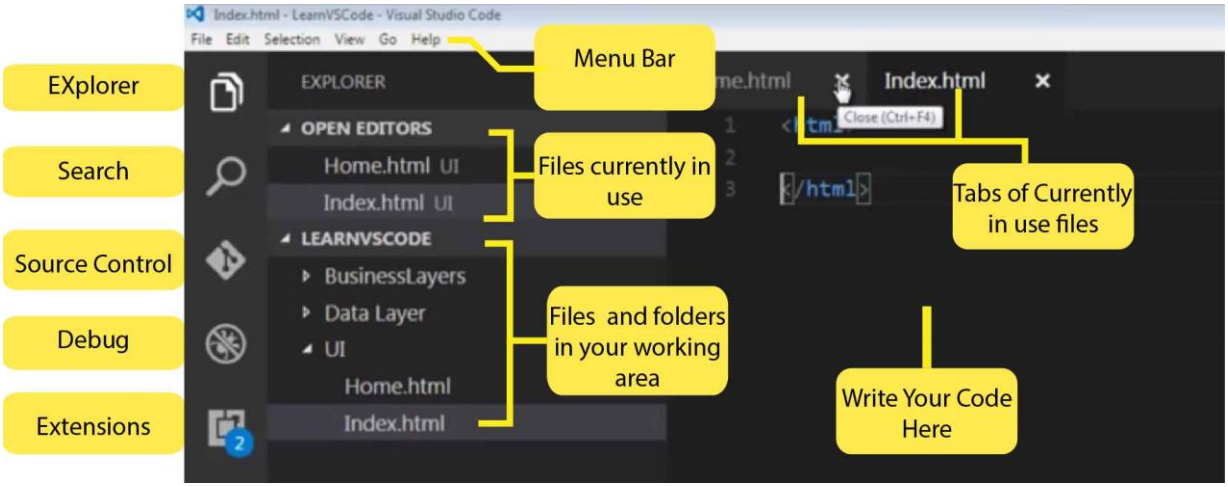


*Figure 1: Interface of VS Code*

## 1.2. Create a project folder:

One needs to create a folder by going in to file menu and create a file inside the folder that serves as a source code. Once you see your folder in the explorer panel of the VS code interface you can add more folders and file by clicking on the new file and new folder icons that have + sign marks on them. Explorer shows you what files and folders you have in your project. In the explorer panel, you can see all your source codes.

## 1.3. Understanding explorer and open editor:

On the explorer panel as described above you can create folders and files (html, SVG, etc.). The explorer panel has two sections: on the lower section: all files and folders in your working folder and on the upper section the files you are currently working on. From the menu bar, you can view + toggle the explorer panel to the right side of the VS code interface. VS code is a generic editor. It does not necessarily lean in to any of the usage of any of the programming languages.

## 1.4. Extension support for third party:

Programmers may use VS code to directly write their HTML codes but as VS code is a generic coding tool, it does not have a built in support for programming languages such as python, Java, C#, etc. In order to code in the above languages the programmer needs to go to menu + view + extensions. From here the programmer can simply search for a plugin for any language he or she wants to use and install it and start writing/ coding in the code area.

## 1.5. Integrated terminal

Menu + view + integrated terminal. This basically integrates your computers command prompt as part of your interface under your coding window. The standard Windows command line is windows power shell but users can change the default command line with other standards such as gitbash by downloading it from [www.git.scm.com](http://www.git.scm.com).

For example, you can go in to the terminal and fire 'npm install http-sever' command to set up a web server for your code or program or design.

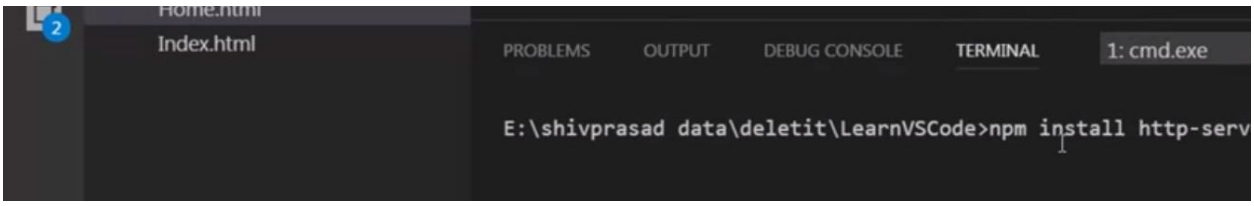


*Figure 2: 'npm install http-server' command*

Once you order the npm command, you will find a webserver folder in your main working folder under the name of 'node_modules'. And for example you can continue and fire 'http-server –p 98' command in the terminal to start running your program on port 98. And you can copy one of the ports in to URL of any browser and display your work and share your work with others globally.

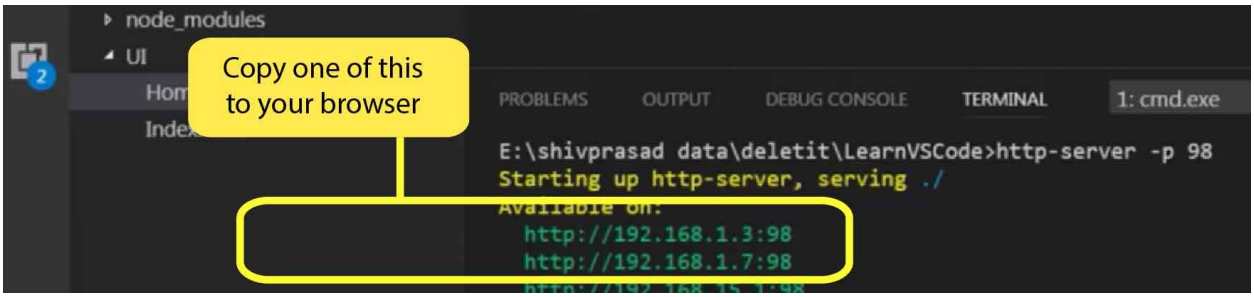


*Figure 3: 'http-server -p 98' Command*

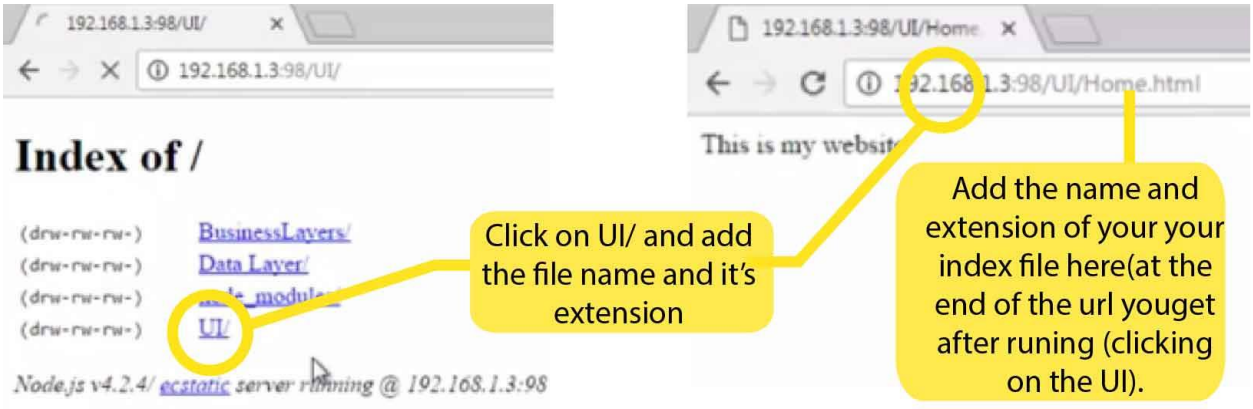


*Figure 4: Browser viewing*

### 1.6. Reveal in explorer features

If you want to open a folder or a file at its location on your computer, right click on the file or folder from the currently working on section and select reveal in explorer from the drop down commands.

### 1.7. VS code preferences

While you are using VS code for your project you will end up creating and generating several files that you will find browsing through complicated therefore you will need to make the process according to your preferences. To do this go to Menu + File + Preferences + Setting. Once you click on the settings, you will end up with the following file setting editing options. Here you will notice that there is a difference between VS code and visual code studio as the user experience design for the visual code studio is more user friendly. See the following figures VS code on the left and visual studio code on the right:

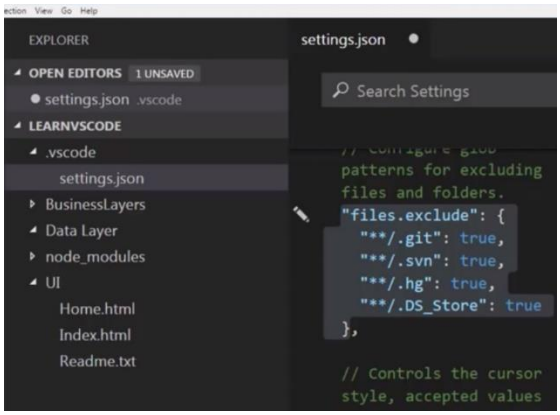


*Figure 5: VS code Preference Setting*

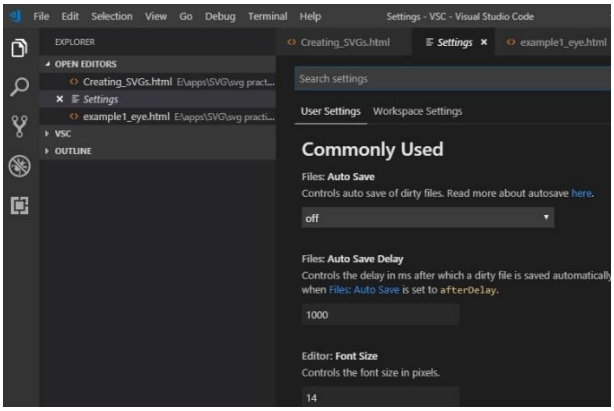


*Figure 6: Visual studio Code preference setting*

### 1.8. Activity bar or Side Bar or Left tool bar.

The bar on the left landside. As showmen on figure 2, this bar comprises: Explorer, search, source Control, Debug, and the Extensions buttons or icons. The Explorer icon allows to contract and expand the explorer panel. The search or global search button or icon allows to search or replace text in any file in any folder. The source controller or version controller icon allows you to connect to GIT repository while the debug button or debugger icon to debug your program. The extension basically lets you work on what we have covered on step 1.4; to add third party extensions.

### 1.9. Configuring task in VS code.

Code review, publish, build, compile file scripts, etc are some tasks a programmer wants to run in VS code but these tasks are not generally recognized by VS code. VS code simply needs to be fed with command lines to carry out this task. The programmer needs to go to the integrated terminal we have seen on step 1.5 and type specific commands like: tsc “supplier.ts” for type script; msbuild to build ms, etc. Short key for build is [ctrl + shift + B]. You need to feed VS code with the specific command because VS code unlike Visual Studio Code is just a text editor not an integrated environment. Crl +shift + B is run a configure tasks.

### 1.10. Going back and forth, Zoom in, Zoom out.

Ctrl – ; ctrl + ; menu + go + back; menu + go + forward; etc. Are for your navigation.

---

## 2. Text editor or Integrated Development Environment (IDE):

Which one is better to use? Some programmers like the challenge of using VS code but IDE is better as it is development environment that integrates most needed tools to complete a task in programming. But there might be times when you will need to use text editors: If you are using Just Java script (JS) and HTML, you will need to go to the trouble of using the whole IDE. And If a language (especially if programmers are working (creating) a new one) does not have IDE then you can only use a test editor to write the language and compile. If one is learning how to program, it is advised that they use text editors rather than IDEs as text editors make the student learn more on the fundamentals of putting together and configuring tasks manually than just have an easy way out through the IDEs advantages. But if there are dead lines that are tight and you are an expert, you may use IDEs. An IDE provides all this services in one package. For example visual studio code provides the following tools in one package: Code editor, visual studio debugger, personal webserver, SQL server, file edition, etc. Therefore VS code is only the editor part of visual Studio Code. See the following illustration for a similar explanation.

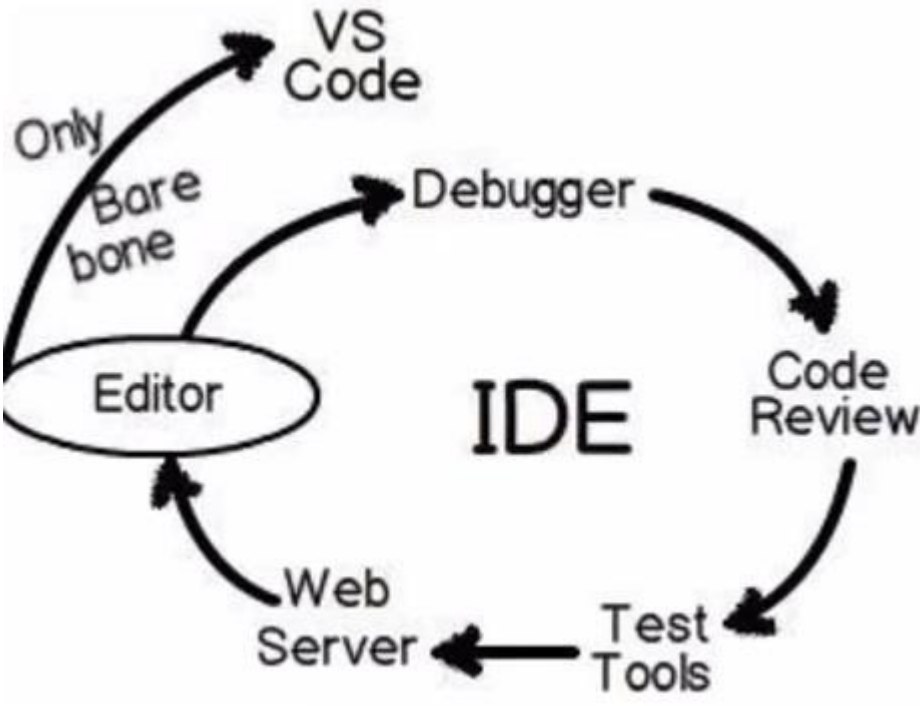


Figure 7: VS code and Visual Studio Code
Source: www.Questpond.com

You can download the software from www.code.visualstudio.com. Visual studio too like VS code is a cross platform that runs on Linux, mac and windows. Initialize repository (push to a remote repository); Intellisense (code completion, drop down windows, parameter info, quick information, etc. things that make your coding efficient) can be considered for extension. The inetlisence of visual studio code woks with: Jyson, TS, JS , sass, less, html and css. To do this one needs to change their settings: menu+ setting file+ preferences + setting. But for python and php one needs to add extension to get the intellicence for them (PHP, python).

Extensions for visual studio code include: Setting Sync (saves on github all your settings, extensions, etc. from your visual studio code environment. It saves it and post it in github under your account and it is easy to setup on your visual studio code. Got to Setting sync+(crl+alt+u)+sign in to github+ add setting synk as your token + select gyst + .. ), SVG viewer, Rest Client (similar to post man but unlike post man Rest client is external and not build into your text editor), Paste Json as Code, ES7 React/Redux/GraphQL/React-Native snippets, Import Cost, indent-rainbow, TODO Highlight, Auto Rename Tag, Prettier-Code formatter.

Now that you are starting to code further you may have to use Emmet intensively to effectively write your code. Study the list of hot keys for emmet to get yourself accustomed to it. Bracket, IDEs such as Visual studio code have the intellisense of EMMIT that will allow you to use abbreviation+ tab to complete your functions or your markups for instance as in HTML: h1 + tab for <h1></h1>, for a class on h1 you can type h1.the class + tab you will get <h1 class= <"the class"></h1>, for IDS you can type h1#the id + tab you will get <h1 id= "the id"><h1>. If we want to just work with div for IDs and classes, we do not have to specify an element. We just go and put a # and . In front of our ID and Class respectively. For example: #the ID +tab this yields . For class: . the class + tab this yields <div the class = "the class"> . For list li*4 + tab yields:

```
<li></li>
<li></li>
<li></li>
<li></li>
```

And ! + tab yields:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>Document</title>
</head>
<body> </body>
</html>
```

Templates such as edge, react, etc. do not work with emmet by default therefore one needs to go to preferences and include them.

Among the various types of extensions, students should look in to Live Server extension. From the extension icon students can search for it by the key word 'live server'. Live server allows programmers to have an environment where they can work on their project by viewing it live. After opening it in extension, install it. And reload. Go to the html file and right click and from the drop down list, select open with live server. Another extension for doing the same thing can be open in browser. Other extensions to look in to can be: VS code icons (allows programmers to display files and folders in the explorer with different icons); VScode-faker (allows programmers to create pseudo addresses, other formal standards for test data purposes); PHP Intellisense (if a programmer is using a PHP code in visual studio code, can use this

extension for better performance); angular v5 Snippet (this gives you code highlighting, code hinting, etc.); JavaScript (ES6) code snippets ; React-Native/React/Redux snippets for es6/es; vue 2 snippets; python.

# 3. Remote repository for your project on Github

Creating github repository and pushing your content there can be a bit hairy. Take a look at the following texts carefully and follow your instructor's demonstration to get it under control.

To initialize a repository and push your work to github: Go to your github account + click the plus sign on the menu of the window + new repository + fill out all the prompts and push create repository + we can copy the code from the created repository to our command line (terminal) or since we have initialize repository functionality integrated, we can do the remaining work in Visual studio Code: Clique on version control (source control) and clique on the icon on the top of the source control panel; on the pop up window, press initialize repository (this creates .git folder); the files you will see now in your source control panel are upstaged files listed with "U" mark; click on the "….." icon on the top of the panel and select "stage all changes". This changes all the files in to stage ADD with "A" mark on them. After that we go back to the "….." and select "commit all". We write out comment for example with a text as "initial commit" and we press enter. After this, all lists of files in the version control panel will be committed. Whenever we change any element in our project, we go through the staging and committing process over again to commit the change: Stage all changes, commit staged (we write comment here) at "….." icon. Close the version control (source control) panel. All things done till now are on our local environment. To push our content to the github repository, we need to copy "git remote add original https://github.com/.....git "from our github account repository created earlier and paste it on to visual studio code integrated terminal window (command line) and press enter;


*Figure 8: Remote Github repository on CMD (integrated terminal) of visual studio.*

This links us with github as a remote repository. After that, we go to "…." Icon on visual studio code and we select "push to" on the provided prompt line we select the added remote repository. This action will push all of our files to the remote repository on github. We can go to our github repository and reload the page and we can view all files pushed there (uploaded). We can open our index file see the html code uploaded from the github repository itself. If there are further changes in your project to be done in visual studio code you have to again go through the steps: Make your change + notice the file changed will be displayed in the version (source) control list with M mark + select stage changes from a right clique on the file at Version control panel (this can also be done at "….." icon) + at "……." icon select stage changes + add comment at the provided prompt line and press enter + go back to "….." icon and select "push to" and pick the directory to github selected earlier. Now you can go and see the changes made in your github account on your index file. We can create a .ignore file in visual studio code and add files to

it. Files we do not want to push to our directory. To do that after we created the .ignore file we then go to version control panel from there we notice the file we want to ignore and right clique on it, from the drop down list select add to ignore. (1) Stage all changes; (2) commit all (add comment after this one) ; (3) push at "…." (4) Select the github repository at the prompt line. Now you can go to your github account and see that the file you needed to be ignored is not added. Browse through the repository in your github account and see the number and types of commits you made.

Revision on list of extensions for Visual studio code:

Box 1: List of Visual Studio Code Extensions
Reference: web dev simplifier

1. vscode-icons: https://marketplace.visualstudio.com/...
2. Color Highlight: https://marketplace.visualstudio.com/...
3. Git Lens: https://marketplace.visualstudio.com/...
4. Auto Rename Tag: https://marketplace.visualstudio.com/...
5. Live Server: https://marketplace.visualstudio.com/...
6. Code Spell Checker: https://marketplace.visualstudio.com/...
7. Debugger for Chrome: https://marketplace.visualstudio.com/...
8. Rainbow Brackets: https://marketplace.visualstudio.com/...
9. IntelliSense for CSS class names in HTML: https://marketplace.visualstudio.com/...
10. Any Linter
4. Settings Sync: http://bit.ly/2O6eGdU
5. SVG Viewer: http://bit.ly/2PptZmY
6. REST Client: http://bit.ly/2Pn1kii
7. Paste JSON as Code: http://bit.ly/2PmQsAO
8. ES7 React/Redux/GraphQL/React-Native Snippets: http://bit.ly/2PneMm0
9. Import Cost: http://bit.ly/2Pkerkh
10. Indent Rainbow: http://bit.ly/2PgI1XX
11. TODO Highlight: http://bit.ly/2Pmr1zn
12. Prettier: http://bit.ly/2Pmr

*End of lecture note!*