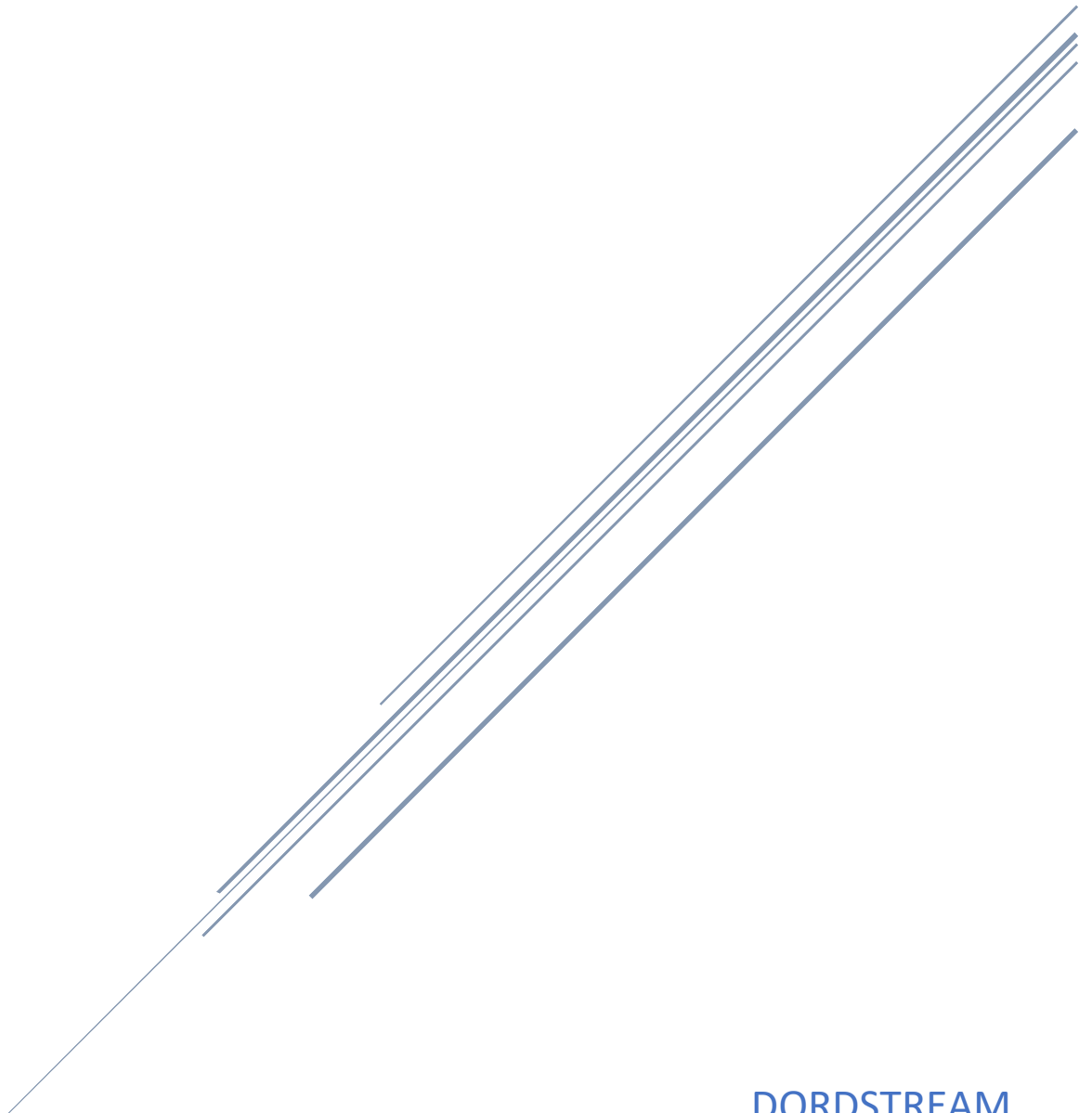


GETTING STARTED WITH FRONT END DEVELOPER (FED) IN DORDSTREAM



DORDSTREAM
FRONT END DEVELOPERS

Getting Started with Front End Developer (FED) in DordStream

This tutorial teaches the basics of building a DordStream code library (Themes). We recommend you to host DordStream CMS software on your local server (Computer) (internet information services (IIS)). before starting this tutorial.

Prerequisites

Install the following:

- [DordStream Studio.](#)

Languages required:

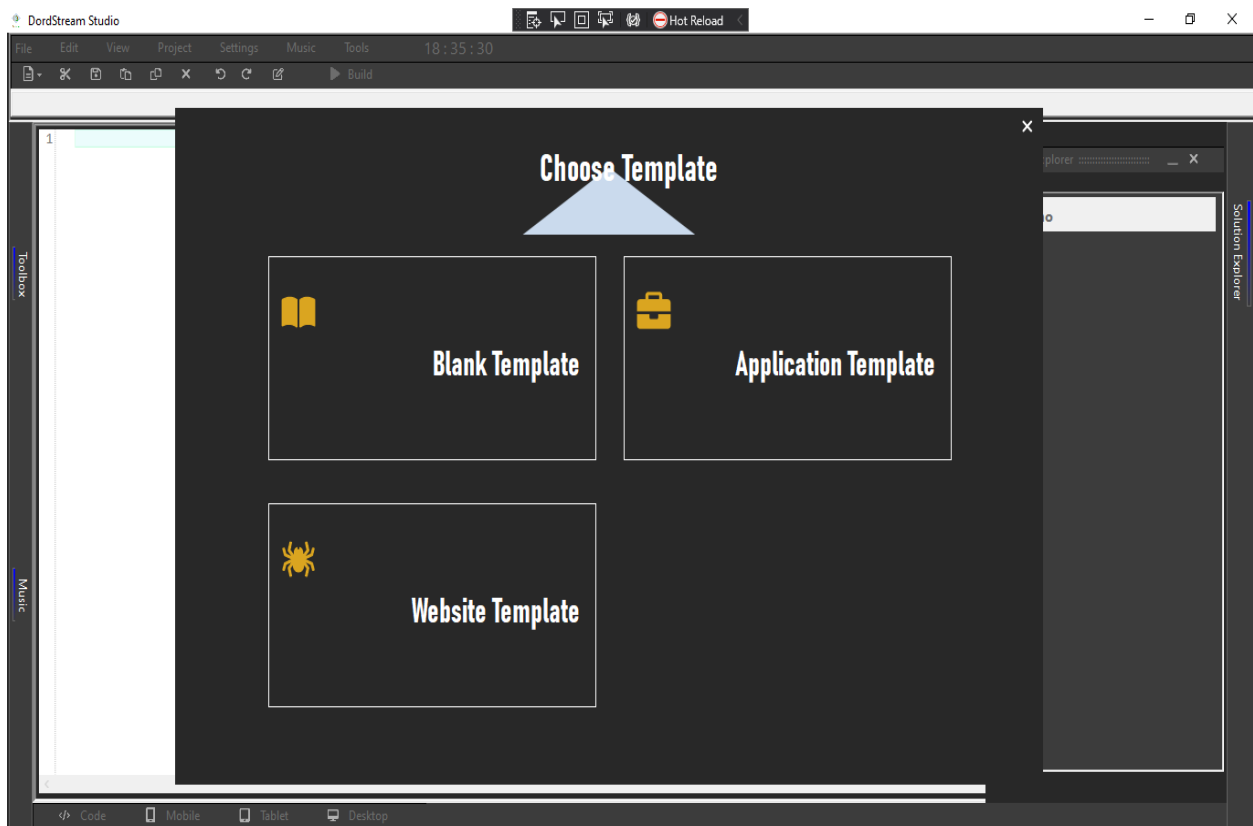
- [Html](#)
- [Css](#)
- [DsHtml](#)

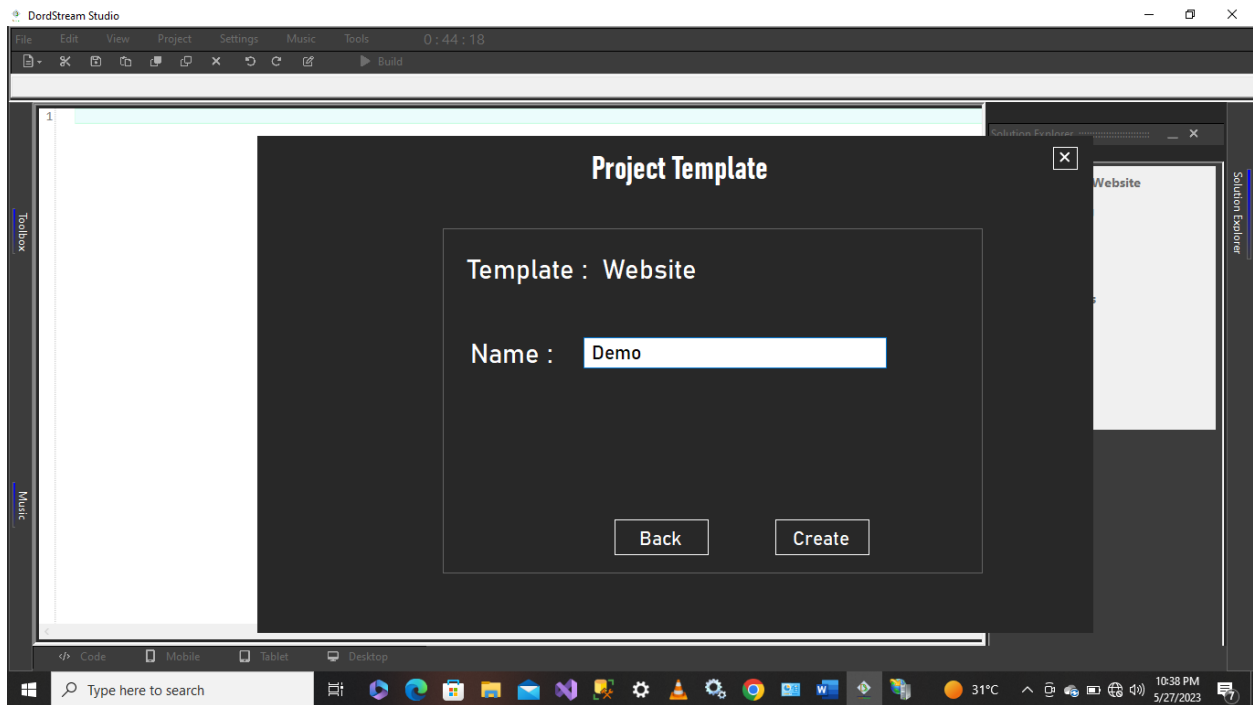
Create a DordStream Project

From the DordStream Studio **File** menu, select **New > Project**.

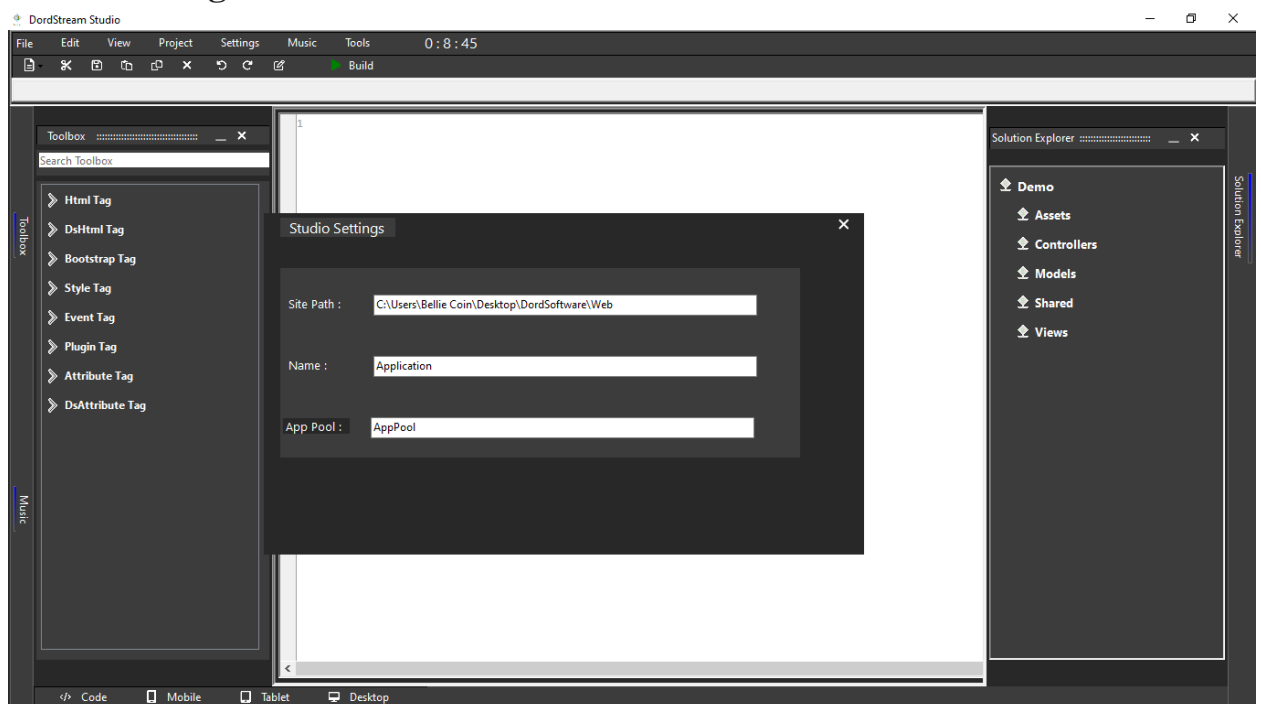
Create a new project and select project template. Name the project **Demo**. It's important to name the project *Demo* so the namespaces will match when publishing.

- Select the template, and name your project.

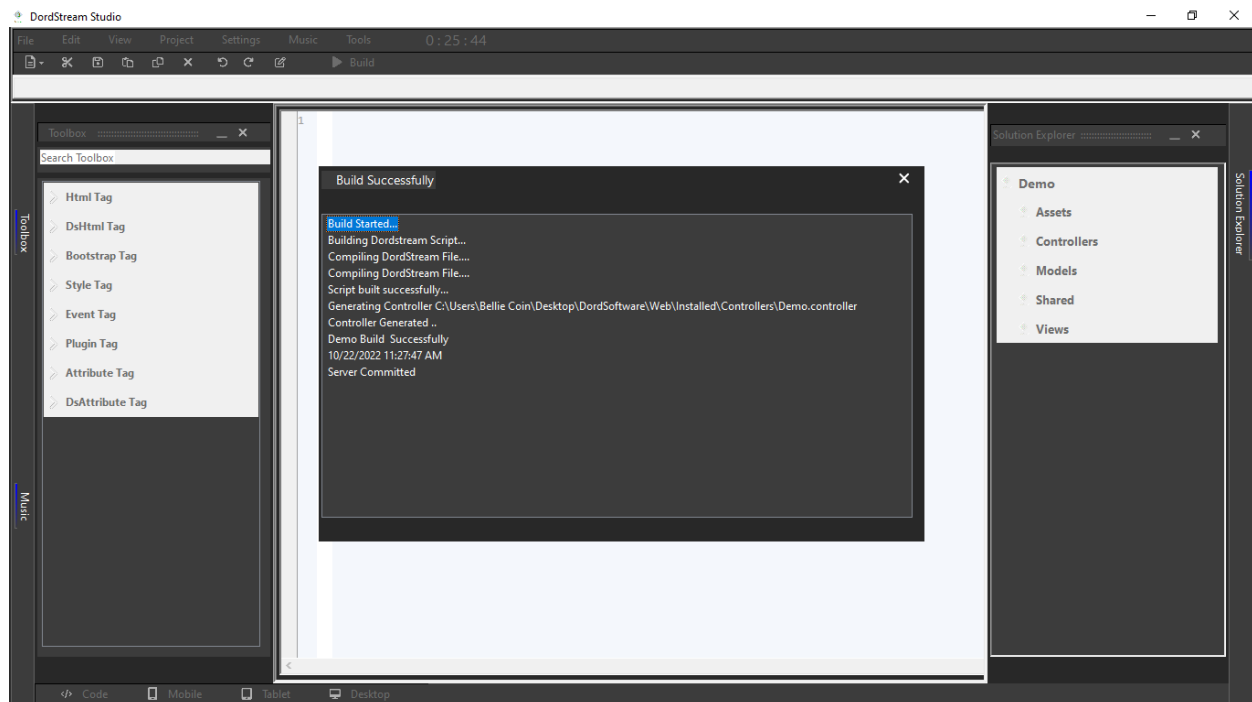


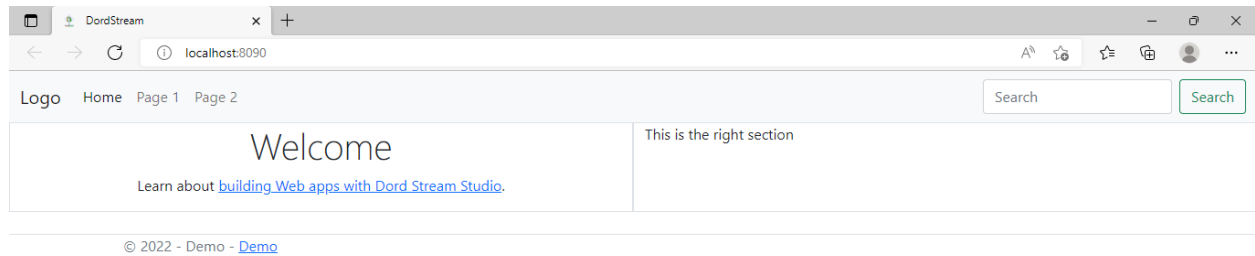


- Connect **DordStream Studio** with the **DordStream CMS** hosted on Internet Information Service (IIS). From the DordStream Studio **Settings** menu



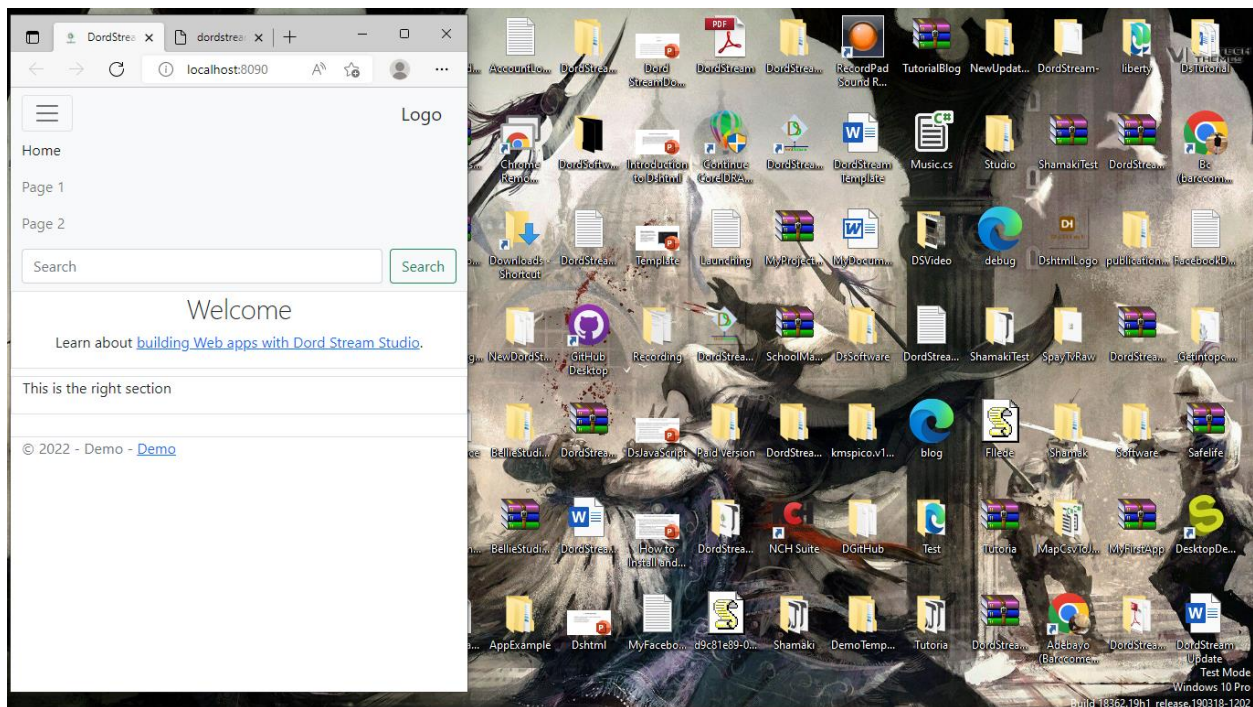
- **Site Path** is the path to the software (**DordStream CMS**). double click on the textbox to select the path to the software then press **Enter Key** to save the path.
- **Name** is the site name given to the hosted software (**DordStream CMS**) on Internet information services (IIS). After inputting the name then press **Enter Key** to save the site name.
- **App Pool** is the application pool name of the hosted software (**DordStream CMS**) on Internet information services (IIS). After inputting the application pool name then press **Enter Key** to save the application pool.
- Click on **Build** to run the app in debug mode.





- Run web browser and access the localhost host on IIS. The address bar shows localhost:port# and not something like example.com. That's because localhost is the standard hostname for your local computer. Localhost only serves web requests from the local computer. When Visual Studio creates a web project, a random port is used for the web server. In the preceding image, the port number is 8090. When you run the app, you'll see a different port number.

The default template creates **Demo**, **Home**, **Page 1** and **Page 2** links and pages. Depending on the size of your browser window, you might need to click the navigation icon to show the links.



Project files and folders

The following table lists the files and folders in the project. For this tutorial

| Folder | Purpose |
|--------|--|
| Assets | This is a folder that contains set of files generally text content, graphics, image, audio, video, Javascript, Css, etc. which server does not change when sending to users. |

| Folder | Purpose |
|--------|---|
| Shared | <p data-bbox="386 327 1430 407">This is a folder that contains set of files that will be shared throughout the application.</p> <p data-bbox="386 449 1458 573">Shared folder has a required (compulsory) files and optional files. compulsory files are layout.html, landing.html, readme.txt, banner.png banner.jpg and error.html</p> <p data-bbox="386 615 1354 739">optional files depend on usage. they are login.html, register.html, post.html, category.html and page.html depending on the project template</p> <ul data-bbox="444 781 1446 1843" style="list-style-type: none"> <li data-bbox="444 781 1446 861">• Layout (layout.html). This is a file that contain the hypertext markup language which defines the project layout. <li data-bbox="444 940 1446 1064">• Landing (landing.html). This is a file that contain the hypertext markup language which defines the landing page interface (first page to be loaded when running the application). <li data-bbox="444 1144 1446 1224">• ReadMe (readme.txt). This is a file that contain project description. <li data-bbox="444 1304 1446 1386">• Error (error.html). This is a file that contain hypertext markup language which defines the error page interface. <li data-bbox="444 1465 1446 1545">• Login (login.html). This is a file that contain hypertext markup language which defines the login page interface. <li data-bbox="444 1625 1446 1705">• Register (register.html). This is a file that contain hypertext markup language which defines the register page interface. <li data-bbox="444 1785 1446 1843">• Banner (banner.png banner.jpg). This is the project banner. (image file). |

| Folder | Purpose |
|--------|--|
| | <ul style="list-style-type: none"> • Category (category.html). This is a file that contain hypertext markup language which defines the blog post category page interface. • Post (post.html). This is a file that contain hypertext markup language which defines the post page interface (display post). • Page (page.html). This is a file that contain hypertext markup language which defines the blog page interface (display blog page). |
| Views | <p>This is a folder that contains html file that is visible to the user. this is the visible part which user can access through web browser.</p> <p>The files serve as route on server.</p> <p>The file can be access by everyone on browser through domain/ route/ filename without extension. E.g. www.example.com/route/index.</p> <p>View is being replace with route on sever.</p> <p>View folder consist the following folders</p> <ul style="list-style-type: none"> • Admin. This is the folder that contain the files which can be access only by users assigned the role of admin on server. It can be access on server by domain/route/admin/filename without extension. e.g. www.example.com/route/admin/index. • Author. This is the folder that contain the files which can be access only by users assigned the role of author on server. It can be access on server by domain/route/author/filename without extension. |

| Folder | Purpose |
|--------|--|
| | <p>e.g. www.example.com/route/author/index.</p> <ul style="list-style-type: none"> • User. This is the folder that contain the files which can be access only by users assigned the role of user on server. It can be access on server by domain/route/user/filename without extension. e.g. www.example.com/route/user/index. • Auth. This is the folder that contain the files which can be access only by authorized users on server. It can be access on server by domain/route/author/filename without extension. e.g. www.example.com/route/auth/index. <p>Note: accessing folder on browser without specifying the filename will check for index.html file, if the index file not will redirect to the error page. E.g. accessing www.example.com/route or www.example.com/route/admin or www.example.com/route/author or www.example.com/route/user or www.example.com/route/auth on browser will check for index.html and render on browser if exist or else redirect to error page.</p> |

Working with Dordstream Html

In this section, you will learn how to write your first Dshtml document

Dshtml is a set of custom tag which extend the function of html by communicating with the CMS software

In this tutorial we will retrieve and display a recent post posted on the CMS software

This can be achieved in two ways, either using recentpost tag or setting ds-parent attribute to recent

```
<recentposts>
```

```
<div class="media" >
```

```

```

```
<div class="media-body">
```

```
<h5 class="mt-0" ds-title="1"> media Content 1</h5>
```

```
<p ds-content="1">Media Content 2</p>
```

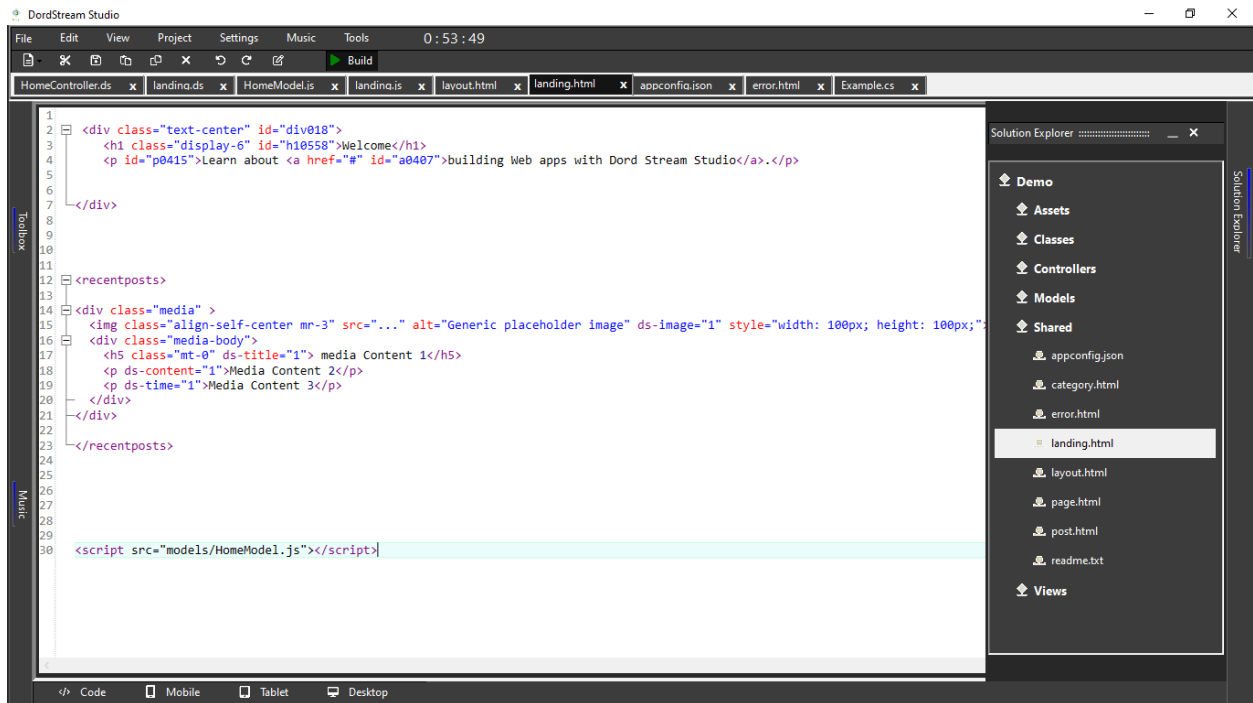
```
<p ds-time="1">Media Content 3</p>
```

```
</div>
```

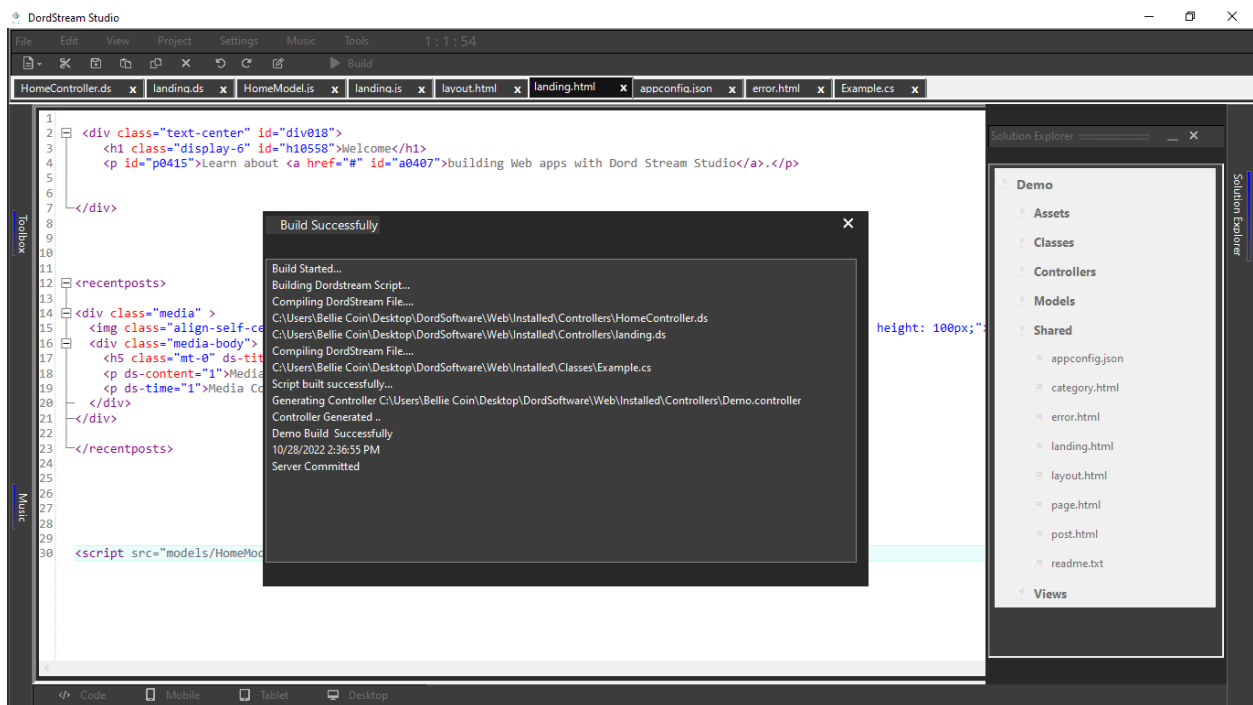
```
</div>
```

```
</recentposts>
```

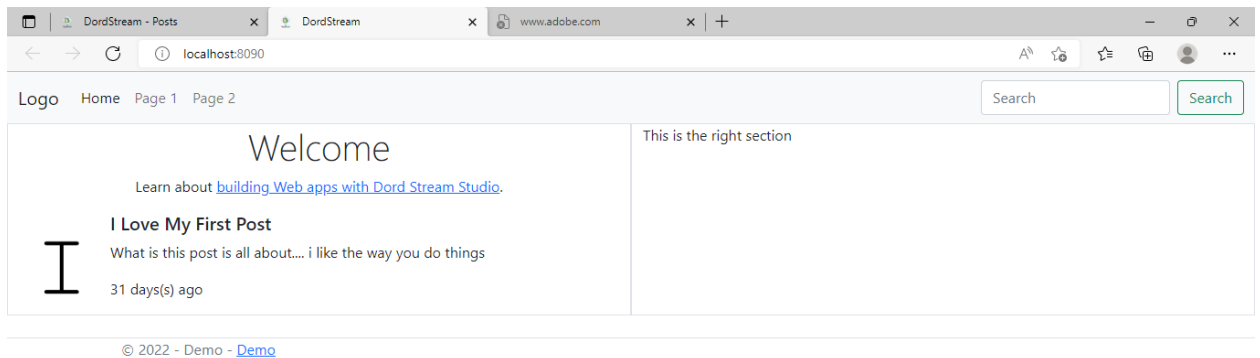
Open any html file and copy and paste the code above to the editor then, click on build and run it on browser.



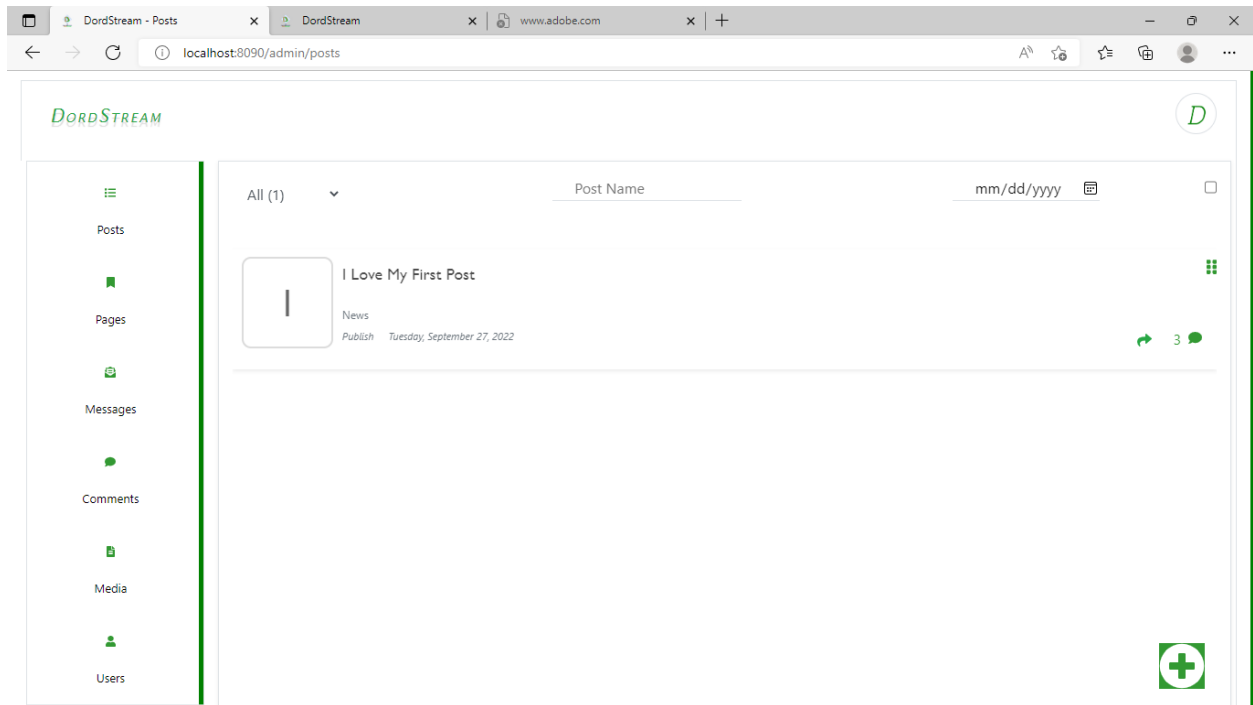
- Click on **Build** to compile the code



- Run it on browser (access it on browser through the url)



- `<recentposts>` is used to retrieved and displayed a recent posts from the posts section of the CMS (admin section). [Learn DordStream Html \(Dshtml\)](#)

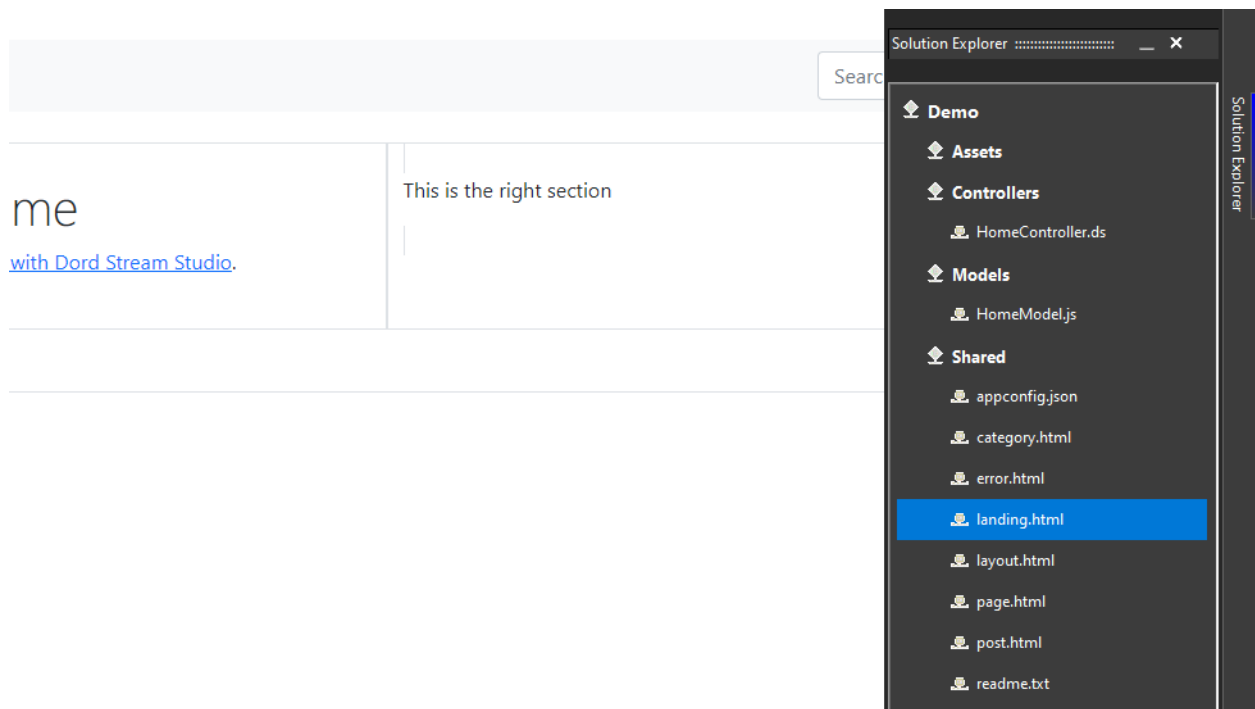


Drag and Drop

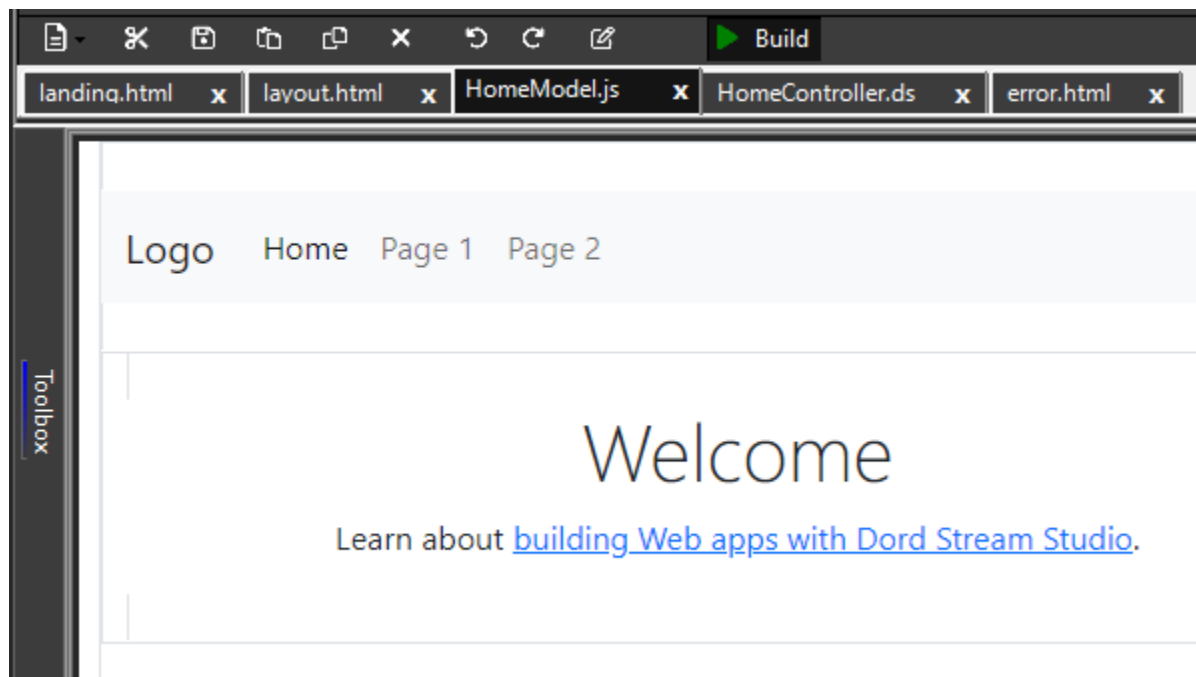
In this section, you will learn how to drag and drop tags on browser



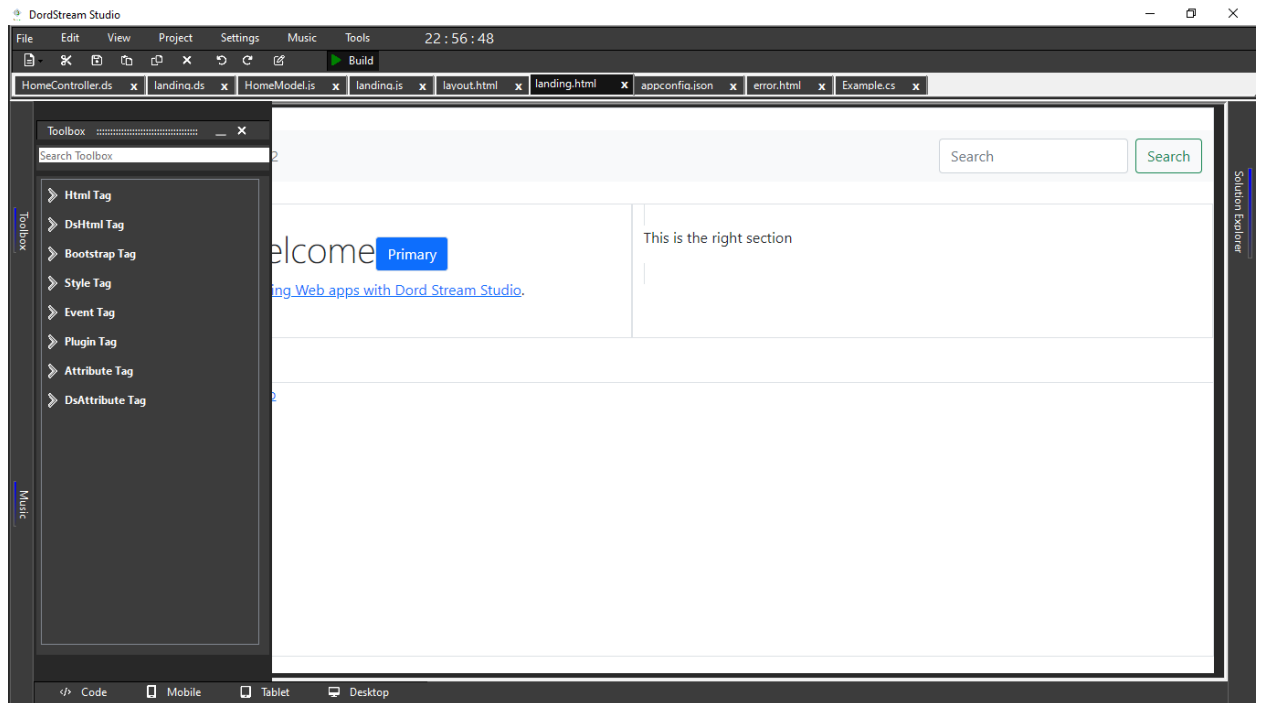
- At the bottom section of the application click on **Desktop**
- In this tutorial we are using landing.html page.



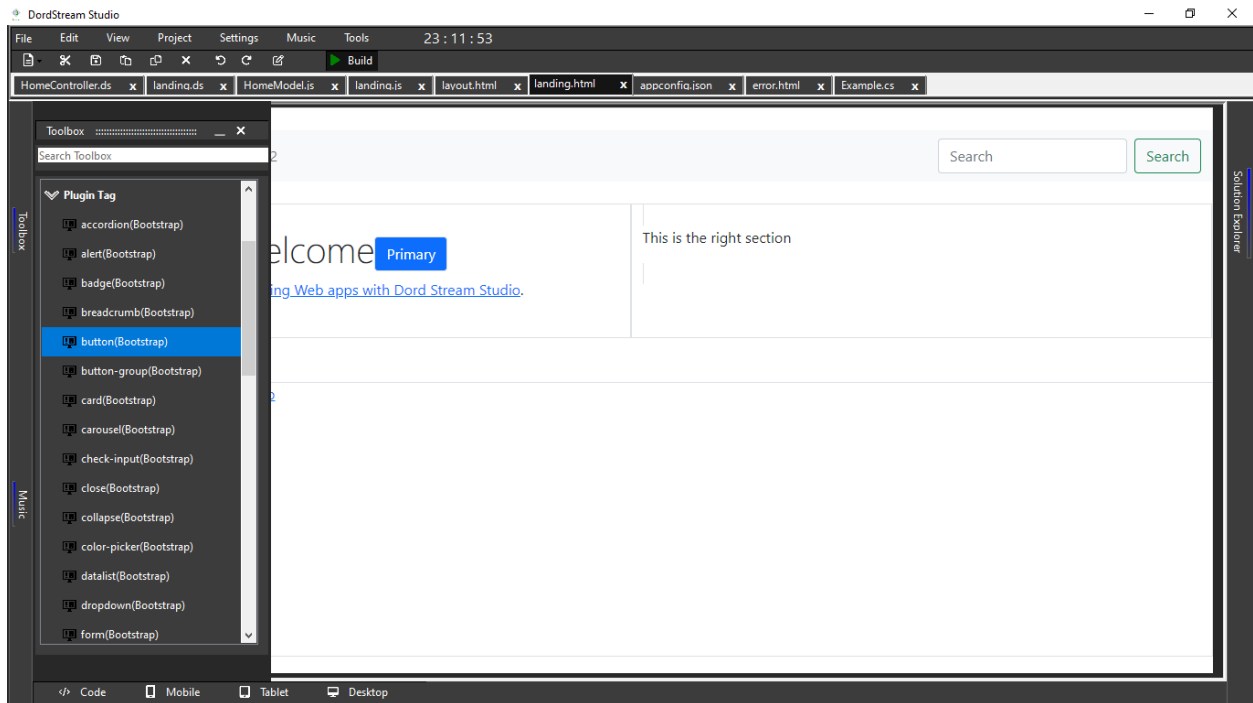
- Click on **landing.html**



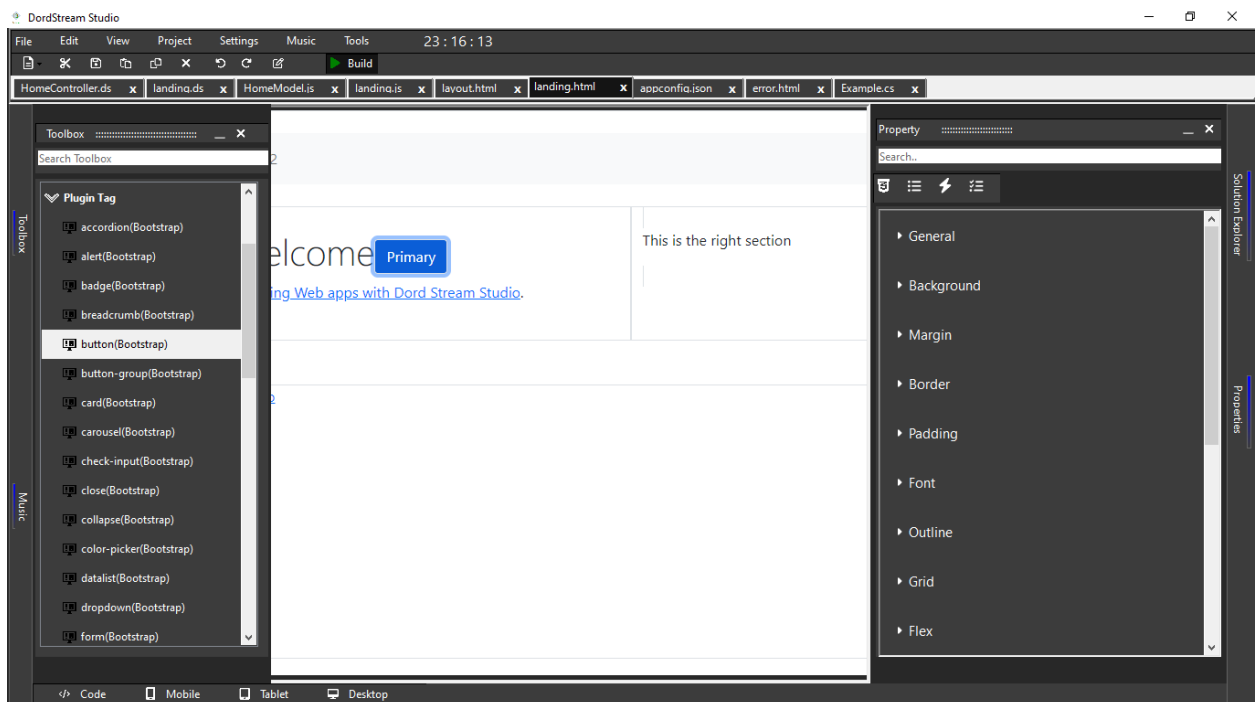
- At the left section of the application click on **Toolbox**



- Click on Tag to Expand the control then drag and drop on browser.
- In this tutorial we will drag and drop button tag in Plugin Tag
- Click on Plugin Tag to Expand the control then drag and drop **button(Bootstrap)** on browser.

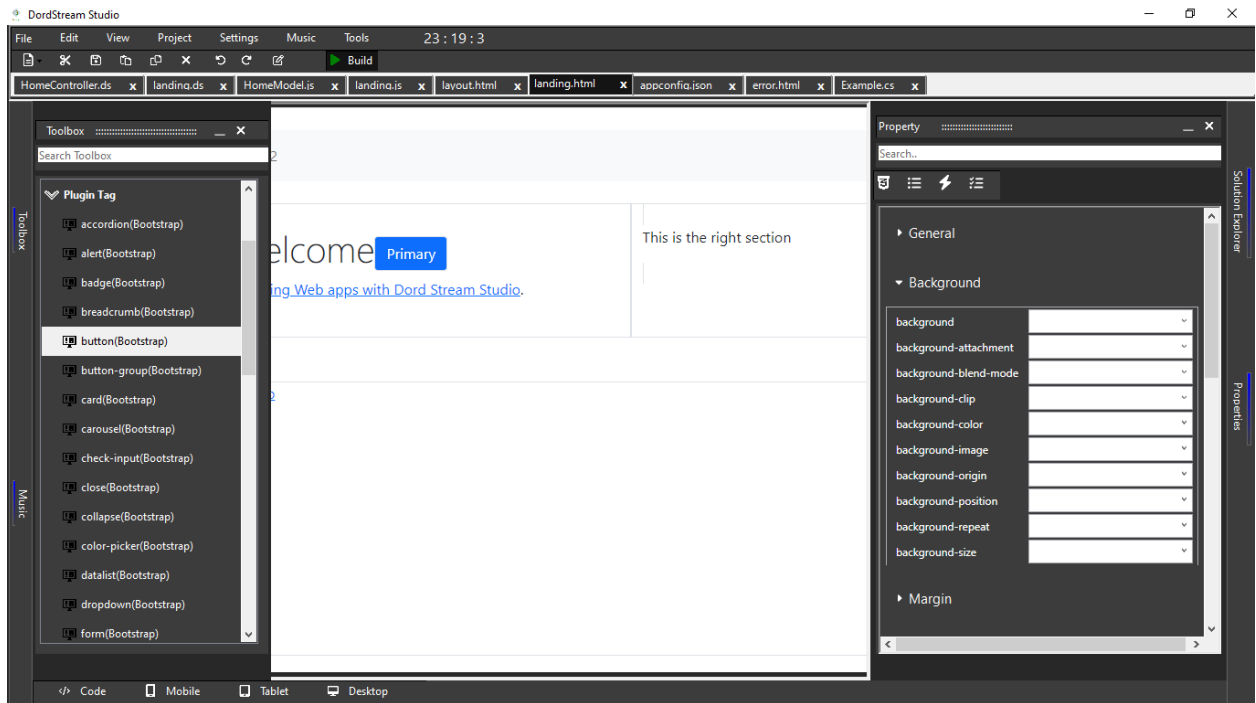


- Double click on the dragged **button(Primary)** then, at property box

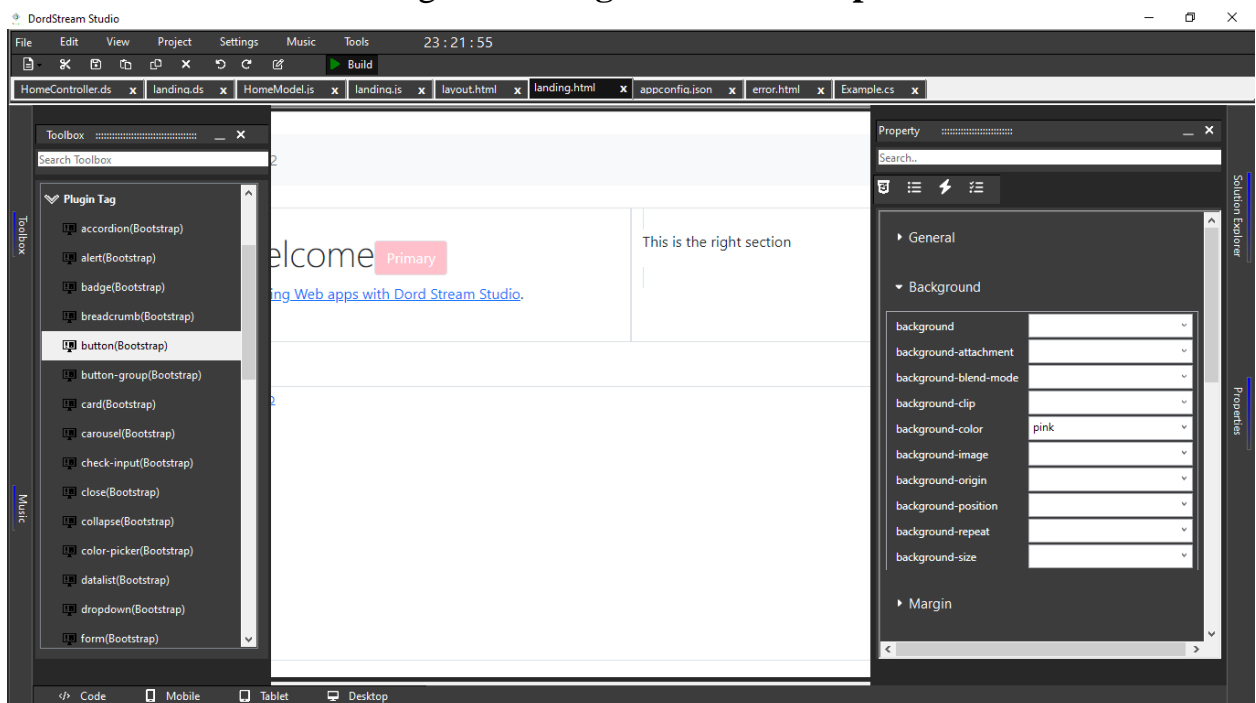


- Property Control consists of stylesheet, attribute, event and dordstream attribute which are used for styling the tag.
- To change the background color of the dragged tag (**button**)

- Double click on the dragged tag (**button**) to display property control.
- Click on **Background** on the property control



- Change the **background-color** by specifying the color name and press Enter key
- In this tutorial we will change the **background-color** to **pink**



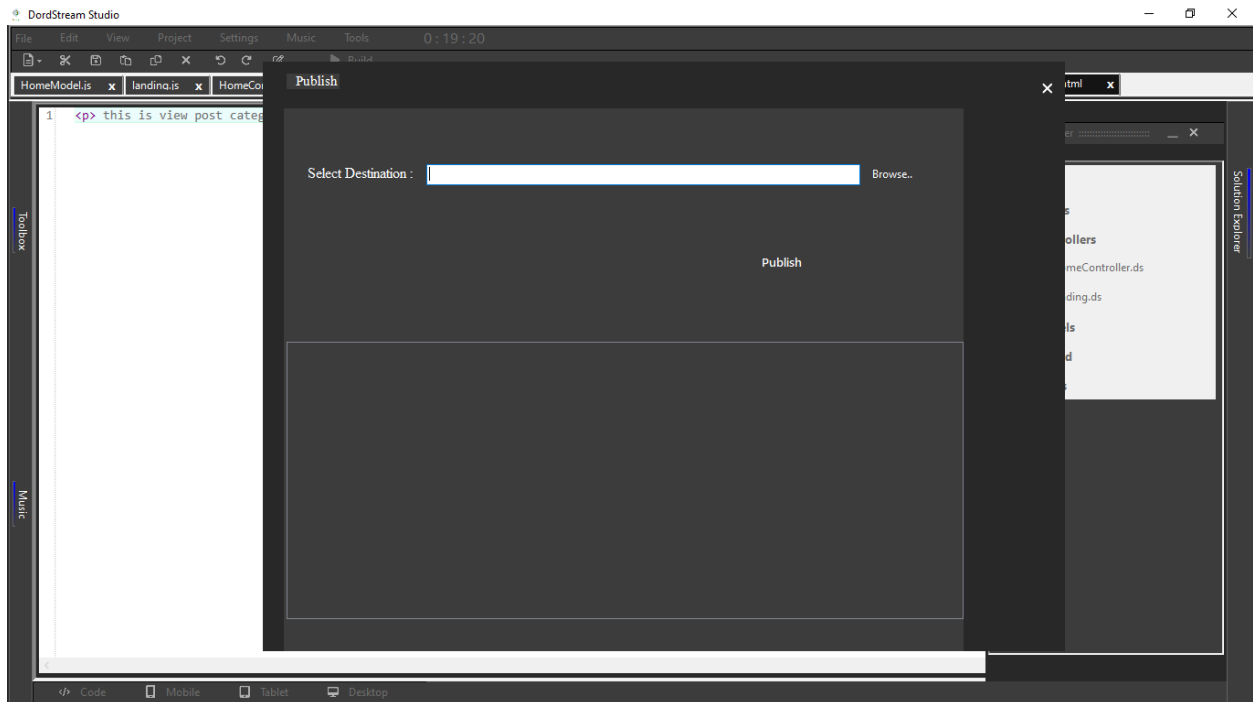
- Click on **Save (ICON)** button at menu bar to save the document.

Note: Always click on Build to build and compile the project before running on browser.

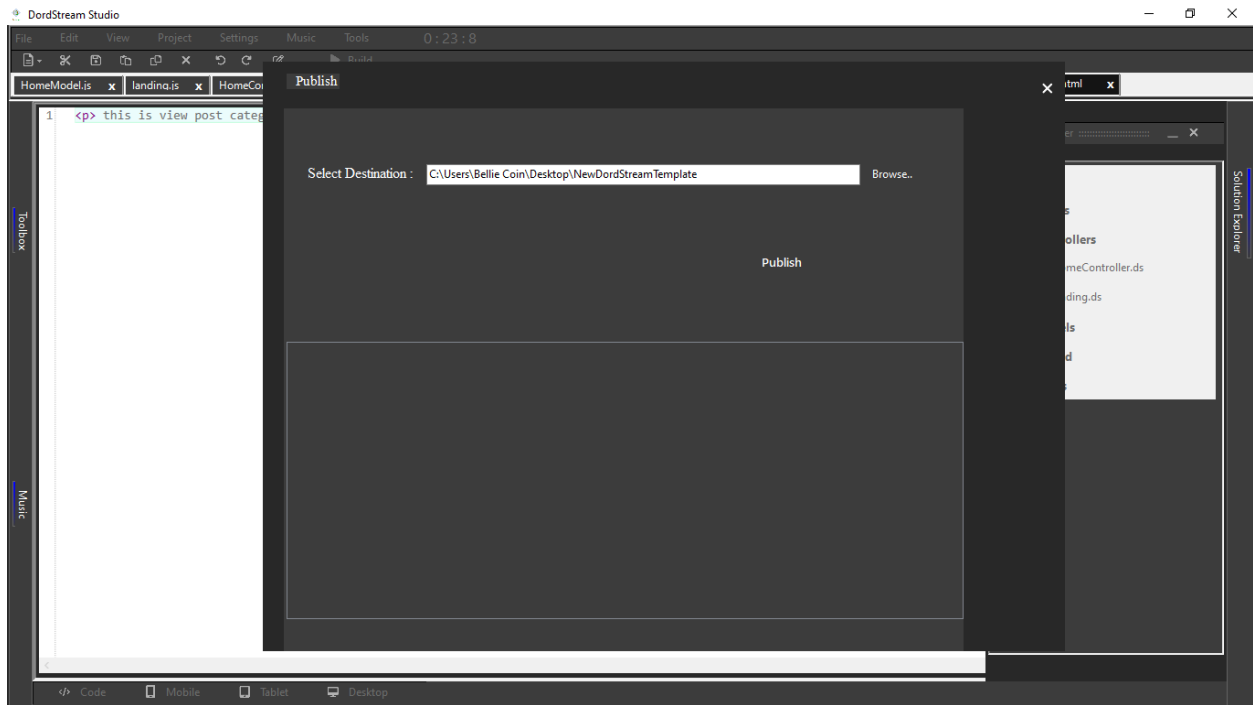
Publish Project

This is the final section where your project is ready for production.

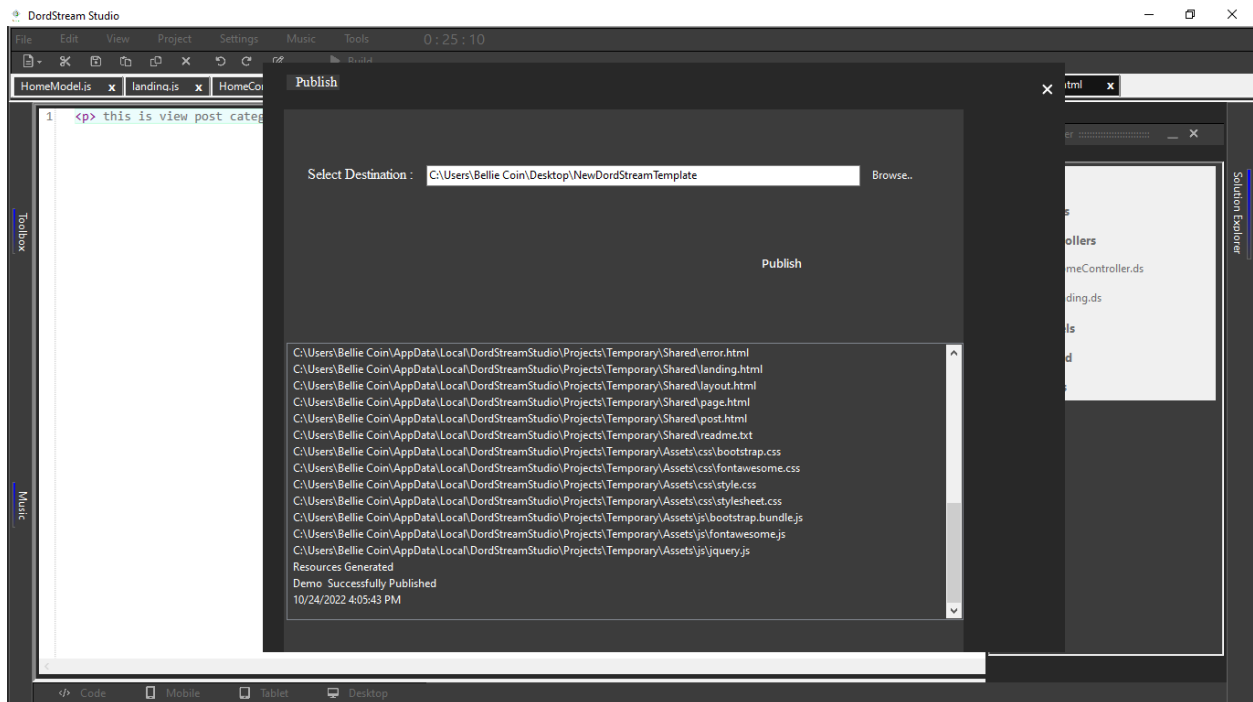
From the DordStream Studio click **Project** menu.



- Click on **Browse** to select the project destination i.e. the location to save the generated file.



- Click on **Publish** to compile and build the project i.e. ready for production.



- After publishing, then you can now upload it to the Themes section of the CMS software.

