# DevExtreme Basics Documentation

## 1. Introduction to DevExtreme

- DevExtreme is a front-end UI framework for creating high-impact, responsive desktop and mobile web apps.- DevExtreme is a front-end UI framework for creating high-impact, responsive desktop and mobile web apps.

- Supports multiple frameworks: Angular, React, Vue, and jQuery.- Supports multiple frameworks: Angular, React, Vue, and jQuery.

- Provides both CDN and local files for integration.- Provides both CDN and local files for integration.

### 1.1 Template for Creating an App Using jQuery

<head>  
 <meta charset="utf-8" />  
 <title>DevExtreme Widgets Basics</title>  
   
 <script src="../Scripts/jquery-3.5.1.min.js"></script>  
 <link rel="stylesheet" href="../Content/dx.common.css" />  
 <link rel="stylesheet" href="../Content/dx.light.css" />  
 <script src="../Scripts/dx.all.js"></script>  
 <script src="../JS/WidgetBasics.js"></script>  
</head>  
<body>  
 <h1>SignUp Form</h1>  
 <div id="signUpForm"></div>  
 <div id="LoginButton"></div>  
 <div id="ResetButton"></div>  
</body>

## 2. Installation

### 2.1 NuGet Package Installation

1. Open the project’s NuGet package manager.1. Open the project’s NuGet package manager.

2. Search for `DevExtreme.Web` and install the required version.2. Search for `DevExtreme.Web` and install the required version.

### 2.2 Package Manager Console

Install-Package DevExtreme.Web -Version 21.1.3

## 3. Widget Basics

### 3.1 Create and Configure Widget

- All DevExtreme UI components must be placed in a container (e.g., `<div>`).

<div id="signUpForm"></div>

$(function () {  
 $("#signUpForm").dxForm({  
 formData: {  
 UserName: "Priyank",  
 Email: "priyank@gmail.com",  
 Education: "B.Tech - CE",  
 College: "DDU",  
 StudentId: 21,  
 }  
 });  
});

### 3.2 Get a Widget Instance

var formInstance = $("#signUpForm").dxForm("instance");  
console.log("Form instance: ", formInstance);

### 3.3 Get and Set Options

// Get Options  
let btnVal1 = buttonInst.option("text");  
console.log("With Instance ", btnVal1);  
  
let btnVal2 = $("#buttonContainer").dxButton("option", "text");  
console.log("Without instance", btnVal2);  
  
let buttonOptions = buttonInst.option();  
console.log("All properties", buttonOptions);

// Set Options  
buttonInst.option("text", "value set by instance.");  
$("#buttonContainer").dxButton("option", "text", "Without instance");

buttonInst.option({  
 text: "Setting all properties using instance.",  
 onClick: function () {  
 console.log("New updated event is clicked.");  
 },  
});

### 3.4 Call Methods

formInstance.beginUpdate(); // Suspend update  
  
formInstance.option("formData", {  
 UserName: "Priyank",  
 Email: "priyank@gmail.com",  
 Education: "BE - CE",  
 College: "DDIT",  
 StudentId: 11,  
});  
  
formInstance.endUpdate(); // Apply updates

### 3.5 Handle Events

$("#LoginButton").dxButton({  
 text: "Login",  
 onClick: function () {  
 alert("You have successfully signed up!");  
 }  
});

### 3.6 Destroy a Widget

// Reset to Default Div  
$("#ResetButton").dxButton("dispose");

// Remove from HTML  
$("#ResetButton").remove();