Step-by-Step: Setting up Bootstrap & Font Awesome with LibMan in VS Code

---

Step 1: Open the Project in VS Code

1. Start **Visual Studio Code**.
2. Go to **File → Open Folder**.
3. Select your **PriceQuotationApp** project folder.
4. You should see folders like `Controllers`, `Models`, `Views`, and `wwwroot`.

Step 2: Make Sure LibMan is Installed

LibMan (Library Manager) comes with the .NET SDK, but confirm it works.

In the VS Code terminal (`Ctrl + ``), type:

```
libman --version
```

- If you see a version number → you're good.
- If not, install it via the .NET SDK or check LibMan CLI documentation.

Reference: *Chapter 3 slides – Adding Bootstrap with LibMan*

Step 3: Install Bootstrap with LibMan

Run this in the terminal:

```
libman install twitter-bootstrap@5.1.0 -d wwwroot/lib/bootstrap
```

- This downloads Bootstrap into `wwwroot/lib/bootstrap`.
- It also creates a `libman.json` file to track installed libraries.

Step 4: Install Popper.js

Bootstrap's JavaScript requires **Popper**.

```
libman install popper.js@2.11.5 -d wwwroot/lib/popper.js
```

Now you'll see a `wwwroot/lib/popper.js` folder.

Step 5: Install Font Awesome

Add Font Awesome for icons:

```
libman install @fortawesome/fontawesome-free@6.4.0 -d wwwroot/lib/fontawesome
```

Now you'll see `wwwroot/lib/fontawesome`.

Reference: Font Awesome Icons

Step 6: Update `_Layout.cshtml`

File: `Views/Shared/_Layout.cshtml`

Inside `<head>`, add:

```
<link rel="stylesheet" href="~/lib/bootstrap/css/bootstrap.min.css" />
<link rel="stylesheet" href="~/lib/fontawesome/css/all.min.css" />
<link rel="stylesheet" href="~/css/site.css" />
```

At the bottom of `<body>`, add:

```
<script src="~/lib/bootstrap/js/bootstrap.bundle.min.js"></script>
<script src="~/js/site.js"></script>
```

References:

- Microsoft Docs: Bootstrap in ASP.NET Core
- Slides: *Razor layout enabling client-side libraries*

Step 7: Add Header to Index.cshtml

File: `Views/Home/Index.cshtml`

At the top, before the form:

```
<header class="bg-primary text-white text-center p-4 mb-4">
    <h1><i class="fas fa-money-bill-wave"></i> Price Quotation App</h1>
</header>
```

References:

- Microsoft Docs: Bootstrap utilities – colors
- Slides: *Context classes example*

Step 8: Style the Form with Bootstrap Grid

Still in `Index.cshtml`, replace the form with a responsive grid layout. Example for **Subtotal**:

```
<div class="row mb-3">
    <label asp-for="Subtotal" class="col-md-2 col-form-
label">Subtotal:</label>
    <div class="col-md-5">
        <input asp-for="Subtotal" class="form-control" />
    </div>
    <div class="col-md-5 text-danger">
        <span asp-validation-for="Subtotal"></span>
    </div>
</div>
```

Repeat for **DiscountPercent**, add rows for **Discount** and **Total**, and add **Calculate/Clear** buttons.

References:

- Microsoft Docs: Bootstrap grid system
- Slides: *Horizontal form layout*

Step 9: Run and Test

1. Open the terminal in VS Code.
2. Run the project:

```
dotnet run
```

3. Open the browser at the provided URL (e.g., `https://localhost:5001`).
4. Test responsiveness:
    - Wide screen → form is horizontal.
    - Small screen → form stacks vertically.

References:

- Slides: *App on large vs small screens*

Step 10: Capture Screenshots

Take screenshots:

- One on a **large screen** (horizontal layout).
- One on a **small screen** (vertical layout).

Save them into:

```
PriceQuotationApp/
└── screenshots/
    ├── large-screen.png
    └── small-screen.png
```

Student Challenges

1. Change the header color from `bg-primary` to `bg-success`.
2. Add a **Tax %** input and update the calculation logic.
3. Replace the icon with `<i class="fas fa-coins"></i>`.