

Project 6-1 Build the FAQs app

For this project, you will build a multi-page, data-driven app with multiple routes like the one shown below.

The Home page on initial load

A screenshot of a web browser window titled "FAQs". The address bar shows "localhost:7058". The main content area has a blue header with the text "Frequently Asked Questions". Below the header, there is a navigation bar with links for "All FAQs", "Bootstrap", "C#", and "JavaScript". On the left, there is a sidebar with two buttons: "General" and "History". The main content area displays five FAQ entries, each with a question and answer, topic, and category information.

Topic	Category
What is Bootstrap? A CSS framework for creating responsive web apps for multiple screen sizes.	General
What is C#? A general purpose object oriented language that uses a concise, Java-like syntax.	General
What is JavaScript? A general purpose scripting language that executes in a web browser.	General
When was Bootstrap first released? In 2011.	History
When was C# first released? In 2002.	History

The Home page with a topic selected

A screenshot of a web browser window titled "FAQs". The address bar shows "localhost:7058/home/index/topic/js/". The main content area has a blue header with the text "Frequently Asked Questions". Below the header, there are navigation links: "All FAQs", "Bootstrap", "C#", and "JavaScript". On the left, there is a sidebar with two buttons: "General" (which is highlighted) and "History". The main content area displays two questions in cards:

What is JavaScript? A general purpose scripting language that executes in a web browser.
Topic: JavaScript Category: General

When was JavaScript first released? In 1995.
Topic: JavaScript Category: History

The Home page with a category selected

A screenshot of a web browser window titled "FAQs". The address bar shows "localhost:7058/home/index/category/hist/". The main content area has a blue header bar with the text "Frequently Asked Questions". Below the header, there is a navigation bar with links: "All FAQs", "Bootstrap", "C#", and "JavaScript". To the left, there is a sidebar with two buttons: "General" (which is unselected) and "History" (which is selected). The main content area displays three questions in a list:

- When was Bootstrap first released?** In 2011.
Topic: Bootstrap Category: History
- When was C# first released?** In 2002.
Topic: C# Category: History
- When was JavaScript first released?** In 1995.
Topic: JavaScript Category: History

The Home page with a topic and category selected

A screenshot of a web browser window titled "FAQs". The address bar shows "localhost:7058/home/index/topic/js/category/hist/". The main content area has a blue header bar with the text "Frequently Asked Questions". Below the header, there is a navigation bar with links: "All FAQs", "Bootstrap", "C#", and "JavaScript". To the left, there is a sidebar with two buttons: "General" (which is unselected) and "History" (which is selected). The main content area displays one question in a list:

- When was JavaScript first released?** In 1995.
Topic: JavaScript Category: History

Specifications

- When the app starts, it should display a list of all frequently asked questions (FAQs).
- The Home page should have a Bootstrap navbar across the top with links that filter the FAQs by topic. The All FAQs link should identify the brand and should clear any filtering and display all the FAQs when clicked.

- The Home page should have a Bootstrap list group to the left of the list of FAQs with links that filter the FAQs by category.
- Here's how the filtering should work:
 - On initial load, or when the user clicks the All FAQs link, the app should display all the FAQs.
 - When the user clicks a category link, the app should filter FAQs by category. If the user subsequently clicks a topic link, the FAQs will be filtered by topic and by category.
 - When the user clicks a topic link, the app should filter FAQs by topic. If the user subsequently clicks a category link, the app should filter the FAQs by topic and by category.
- Use custom routes to create the links for filtering. Use static segments to distinguish the topic route from the category route.
- The FAQ model class should have a primary key that's generated by the database.
- The Topic and Category model classes should have primary keys that are strings. Use primary key values that will be user friendly in a URL.
- The FAQ class should have foreign key fields that relate it to the Topic and Category classes.
- Use EF Code First and migrations to create a database based on your model classes. Include seed data for the topics, categories, and FAQs. (You can copy the data shown above, or you can create your own topics, categories, and FAQs.)
- The URLs for the app should be lowercase with trailing slashes.