

Practical activity - Reporting

The main objective of this exercise is to introduce you to creating and exporting charts to obtain PDFs using data retrieved from a database.

1 Practical objectives

Participants are expected to possess the ability to execute the following tasks after completing the hands-on exercise:

- To retrieve data from a database using C#, passing it to a view, and then accessing the data using Javascript;
- To create a bar chart using Chartjs and display the result in a view;
- To convert a Chartjs chart to an image and place it in a PDF using pdfmake.

2 Database and tables

We will be using a mock-database for that contains information about Baseball Hall of Fame players.

The database consists of three tables (i.e., **halloffame**, **players** and **managers**). The **players** table stores the unique identifier for each player, their first name, their last name, other demographic information, and other game-related information (e.g., weight, batting hand, etc). The **managers** table stores each manager's unique identifier, the identifier of their team, and other match statistics. The **halloffame** table stores the identifier of the hall of fame player or manager, the year they were voted in, the number of ballots they received, etc.

You have been provided with a script (named **SmallBaseball.sql**) to create and populate the database.

3 Program functionality

When the program is launched, its main page must list ten Hall-of-Fame Inductees per page as shown in Figure 1. The logic for retrieving the inductees and listing them for each page has already been implemented.

For this exercise, you will be responsible for implementing the functionality associated with the Export button. The button must generate a PDF that compares the ballots received by all the inductees in that page. For instance, Figure 2 lists the PDF that must be automatically generated and opened when the Export button is clicked on the first page.

Hall of fame

About

The National Baseball Hall of Fame and Museum is a history museum and hall of fame in Cooperstown, New York, operated by private interests. It serves as the central point of the history of baseball in the United States and displays baseball-related artifacts and exhibits, honoring those who have excelled in playing, managing, and serving the sport. The Hall's motto is "Preserving History, Honoring Excellence, Connecting Generations". Cooperstown is often used as shorthand for the National Baseball Hall of Fame and Museum, similar to "Canton" for the Pro Football Hall of Fame in Canton, Ohio.

Inductees (Page 1)

First	Last	Year	Country
Cal	Ripken	2007	USA
Tony	Gwynn	2007	USA
Roberto	Alomar	2011	P.R.
Rickey	Henderson	2009	USA
Barry	Larkin	2012	USA
Nolan	Ryan	1999	USA
George	Brett	1999	USA
Wade	Boggs	2005	USA
Rich	Gossage	2008	USA
Bert	Blyleven	2011	Netherlands

[Next](#) [Export](#)

Figure 1: First page of the hall-of-fame inductees

4 Tasks

In order to implement the complete export functionality, you have to implement the smaller tasks we have identified. You are expected to complete the following tasks:

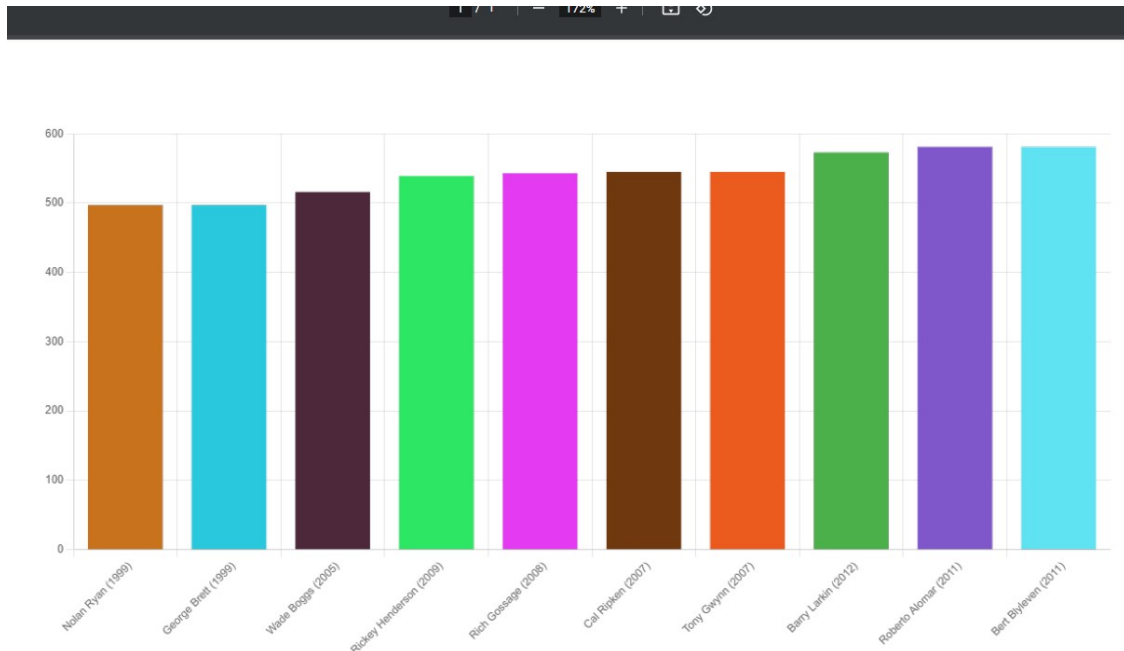


Figure 2: Bar chart of the number of ballots received by the bottom 10 inductees

1. Run the `SmallDatabase.sql` script and modify line 54 of `Web.config` so that you can access the database on your machine. **[5 Marks]**
2. Create a script at the bottom of the `Index.cshtml` file that will convert the list of `HallOfFamePlayerVM` objects passed to the view (see line 1 of `Index.cshtml`) to a javascript object. Log the created object to the console. **[5 marks]**.
3. Use the object created for Task 2 to render a bar chart below the table of Inductees (recall Figure 1). Please note that your x-axis labels need not be oriented in the same way as Figure 2. You can also use whatever colours you prefer for the bars. **[10 marks]**
4. Use the information listed in the Chartjs documentation¹ to disable animations on the bar chart created in Task 3. **[5 marks]**
5. Convert the bar chart² created in Task 3 to an image (base64 string encoding), create a pdf whose only content is the aforementioned image using pdfmake³, and open the pdf when Export is clicked. **[10 marks]**

Please note that the Chartjs and pdfmake dependencies have already been added to the project (see `Shared/_Layout.cshtml`); hence you do not need to add them again.

¹<https://www.chartjs.org/docs/latest/configuration/animations.html>

²<https://www.chartjs.org/docs/latest/developers/api.html>

³<http://pdfmake.org>