SBS Dev Pilot – .NET and SQL

Hello 😊. You'll have already had a quick phone chat and hopefully got an understanding of what SBS do, but just in case here's some info on why this test is as it is.

SBS are a claims management provider for Insurers.

The bulk of our work is on our claims management system – this is a repository of events. Claims come in, some actions are done, people get new stuff, and we tell the insurer what's happened. This requires a lot of database work. As such this test is very geared towards that – we're not yet big enough to support total specialisation.

If you are struggling with the stored procedure or some other aspect – **Please focus on your area of strength**. We want to see which parts you enjoy and how you approach the problem, not to put barriers in the way.

While we're open to different submissions, where possible please try to adhere to the following:

Considerations

We're a .Net shop – It's going to be a core part of your lived experience one way or another. Not showing use of .Net is going to be a harder sell than using it. C# is preferred.

Full frontend frameworks (react, angular, etc.) are to be avoided but smaller libraries you wish to use (e.g. chart.js) are welcomed.

Aim for easy to read & separated SQL. Try to avoid using nested statements & temporary tables. We recommend looking at using CTEs and CROSS APPLYs.

Submission

Please submit a GitHub repository link with the project set to either invite or public.

Within the repository please also include your stored procedure – we may not be running the same version of SQLServer.

Pointers to any required config changes are very welcome.

Please feel free to provide us with feedback during the follow-up interview if it didn't meet your expectations!

Scenario

Using the NBA Database provided (SQL Server Database) create a webpage summary for all teams.

Creation of the displayed data should be carried out by a stored procedure.

Columns required are as follows:

- Name of Team
- Team Stadium Name
- Team Logo
- Season MVP on team
- Number of Games Played
- Number of Games Won
- Number of Games Lost
- Number of Games Played at Home
- Number of Games Played Away
- Points in Biggest Win
- Points in Biggest Lost
- Last Game Stadium Name
- Last Game Date

The database table does not include the team logos. You'll need to find the team logo on Google Images and then store them in the project folder in the directory Images/Logos.

The location and name of the relevant logo should be stored in the database next to the corresponding team enabling you to show the logo next to the team name in the summary.

Logos should be clickable and display the team's homepage in a new browser tab.

Under the table create a bar chart that displays the won column results by default. The lost column header should be clickable to change the bar chart to display lost results.

The table below gives you an example:

Name	Stadium	Logo	MVP	Played	Won	Lost	Played Home	Played Away	Biggest Win	Biggest Loss	Last Game Stadium	Last Game Date
Denver Nuggets	Pepsi Center		Ty Lawson	16	11	5	8	8	112-83	92-112	Bankers Life Fieldhouse	2013-12-31
Miami Heat	American Airlines Arena		Labron James	16	8	8	8	8	115-83	81-112	Chesapeake Energy Arena	2014-01-01
San Antonio Spurs	AT&T Center		Tim Duncan	16	9	7	8	8	116-68	78-101	United Center	2014-01-03
Oklahoma City Thunder	Chesapeake Energy Arena		Kevin Durant	16	5	11	8	8	115-92	81-112	Chesapeake Energy Arena	2014-01-01
Houston Rockets	Toyota Center		James Harden	16	5	11	8	8	101-83	86-112	Toyota Center	2014-01-02