Availity Fullstack .NET Homework Assignment

We highly recommend that you use one of the free source code management platforms (GitHub, GitLab or BitBucket) when storing your code. Once you are ready for us to look at your answers, **send us the link** to your code. If you have any questions about the homework, please do not hesitate to ask.

**Note**: If you have experience writing unit tests, please incorporate unit tests into the coding exercises below.

1. Tell me about your proudest professional achievement. It can also be a personal or school project.

I worked for Advanced Micro Devices (AMD), where I had the opportunity to develop a solution helped automate an important process in the supply chain department.

There was a team dedicated to coordinate the distribution of AMD semiconductor products (chips, graphic cards, mainboards, etc) that relied with the use of a MS Excel spreadsheet template. These team members used this template and manually copy and paste data containing the current warehouse inventory from different distribution sites in the US and Asia, as well as product requests provided by external clients (Apple, HP, XBOX, PlayStation). These product requests were provided through different sources (database connection, web service, CSV files, Sharepoint lists) depending on the client. Once the data was entered, the user would make some calculations with Excel formulas and commit the products AMD would be able to send to the respective clients.

This process was time consuming, as manually copying and pasting took a lot of time, not to mention the writing of formulas for each product listed. My team came up with the idea to automate the process, specifically the consolidation of data in the spreadsheet. I was assigned the task of creating a proof of concept, by creating a demo for a system, that executed a scheduled task to read the different databases and services, as well as read CSV files, and consolidate the data in a database. Then create a Desktop application that created a spreadsheet with that same information and generate formulas automatically through the C# code.

Once this demo was approved, I proceeded to work with the real project, in two separate phases; one to develop the extraction, transformation and loading of data (ETL) from our clients; the second was to develop the desktop application, written in C# to automate the template and the insertion of formulas to optimize the decision making process in the commitment process.

This product was not only used by our team in the US; it was also used by the different warehouses located in both China and India. As the owner of this project, I was in charge of providing support for it, so it would be common for me to meet with these users on the other side of the world to listen to their requests for enhancements and suggestions for new features.

It makes me proud to know that my work made a difference with all these users, to make things more efficient.

1. Tell me about something you have read recently that you would recommend and why. (Can be a Github Repo, Article, Blog, Book, etc)

I am currently reviewing the book “Professional Enterprise .NET” by Jon Arking and Scott Millett. It covers important concepts for the creation of large, maintainable, and scalable applications, including but not limited to separation of concerns, dependency inversion, dependency injection and more. I think it is a valuable resource to practice these principles, as they allow to write code in a more organized, structured manner.

1. How would you explain to your grandmother what Availity does?

I would phrase it like this: “This is the company that creates those programs run in computers and used by the receptionists and front desk personnel at your doctor’s office. Instead of the nurse calling the insurance company to verify your coverage and copay, this program allows them to read your insurance card and get that information instantly, with a few key strokes. This makes it faster to check you in…”

1. Coding exercise (using C#): You are tasked to write a checker that validates the parentheses of a LISP code. Write a program which takes in a string as an input and returns true if all the parentheses in the string are properly closed and nested.
2. Coding exercise (React is preferred, but you may also use Angular, or a JavaScript framework of your choice): Healthcare providers request to be part of the Availity system. Create a registration user interface so healthcare providers can electronically join Availity. The following data points should be collected:

* First and Last Name
* NPI number
* Business Address
* Telephone Number
* Email address

The registration UI should display the data in a table and provide a button that becomes enabled upon data entry of a complete, and valid form. This button can then be clicked to add that user data from the form to your table. This table should also allow for users to be cleared from it.

1. Coding exercise (using C#): Availity receives enrollment files from various benefits management and enrollment solutions (I.e. HR platforms, payroll platforms).  Most of these files are typically in EDI format.  However, there are some files in CSV format.  For the files in CSV format, write a program (not using a 3rd party CSV library) that will read the content of the file and separate enrollees by insurance company in its own file. Additionally, sort the contents of each file by last and first name (ascending).  Lastly, if there are duplicate User Ids for the same Insurance Company, then only the record with the highest version should be included. The following data points are included in the file:

* User Id (string)
* First and Last Name (string)
* Version (integer)
* Insurance Company (string)

1. This database diagram is to be used for the questions that follow: 
   1. Write a SQL query that will produce a reverse-sorted list (alphabetically by name) of customers (first and last names) whose last name begins with the letter ‘S.’

SELECT FirstName, LastName FROM Customer

WHERE SUBSTRING(LastName,1,1) = 'S'

ORDER BY LastName DESC

* 1. Write a SQL query that will show the total value of all orders each customer has placed in the past six months. Any customer without any orders should show a $0 value.

SELECT

FirstName,

LastName,

CASE WHEN TotalValue IS NULL THEN '0'

ELSE TotalValue

END AS TotalOrderValue

FROM

(

SELECT C.FirstName, C.LastName, SUM(Cost\*Quantity) AS TotalValue FROM

CUSTOMER C LEFT JOIN ORDERS O ON C.CustID = O.CustomerID

LEFT JOIN ORDERLINE L ON O.OrderID = L.OrdID

where O.OrderDate >= Dateadd(Month, Datediff(Month, 0, DATEADD(m, -6, current\_timestamp)), 0)

GROUP BY C.FirstName, C.LastName

) T

* 1. Amend the query from the previous question to only show those customers who have a total order value of more than $100 and less than $500 in the past six months.

SELECT

FirstName,

LastName,

CASE WHEN TotalValue IS NULL THEN '0'

ELSE TotalValue

END AS TotalOrderValue

FROM

(

SELECT C.FirstName, C.LastName, SUM(Cost\*Quantity) AS TotalValue FROM

CUSTOMER C LEFT JOIN ORDERS O ON C.CustID = O.CustomerID

LEFT JOIN ORDERLINE L ON O.OrderID = L.OrdID

where O.OrderDate >= Dateadd(Month, Datediff(Month, 0, DATEADD(m, -6, current\_timestamp)), 0)

GROUP BY C.FirstName, C.LastName

) T

WHERE TotalValue BETWEEN 100 AND 500

Again, please let us know if you have any questions. Thanks!

-Availity Team