Questionnaire for Dundas Dashboard Associate

Objective:

- To assess the candidate's the level of technical competency in the areas of databases and business intelligence
- To better understand how an applicant processes and solves problems, such as design, implementation, attention to detail, etc.,

***Note: This test must be written by the candidate that is applying for a position. How we judge this test is based on the job being applied for. All candidates that meet or exceed our criteria for this test are then invited for interview to get more detailed understanding of the candidate's skill sets and personalities.

Contents:

This questionnaire consists of two parts:

Part I: Basic SQL Exercise
Part II: Dashboard design Exercise

Deadline:

Please submit your answer at your earliest time. For Part II, you can either submit a scanned copy with your other answers or mail to the address provided below (be sure to include your name and email address so we can confirm we received your answers).

Dundas Data Visualization Inc. C/O Emma Yang
#500 – 250 Ferrand Dr.
Toronto, ON
M3C 3G8

Thank you for your time and effort. Good luck!

Part I: The questions below are designed to test your knowledge of SQL database.

Below are some data tables:

Products table

ProductID	Name	UnitPrice
1	Chart	999
2	Gauge	599
3	Map	599

SalesPersons table

PersonID	Name	City
101	Andy	Toronto
102	Bob	Montreal
103	Mathew	Vancouver

Customers table

CustomerID	Company	City
10089	IBM	Toronto
24535	Johnsons	Toronto
33555	Goodies	Montreal

Orders table

OrderID	CustomerID	PersonID	ProductID	Quantity	Discount
1	75353	103	2	5	
2	24535	102	1	20	5%
3	10089	101	3	3	
4	10089	101	1	55	20%
5	33555	101	1	2	

QI-1: Identify the relationships between each table if you think it is applicable. You can modify the table if you think it's necessary.

There are two views towards this question: a business database view and a theoretical view. Below, I will cover both views briefly.

- The business database view is fairly simpler, since it is only concerned with the tables and links in terms of primary and foreign keys. Having this view, the tables above form three different links, wince the *Orders* table is the core one, holding all the foreign keys, and the other three tables hold the primary keys necessary to create the links. These links are as follows:
    - Orders -- Products: using ProductID as the key

- Orders -- Customers: using CustomerID as the key
- Orders -- SalesPersons: using PersonID as the key

- The theoretical view might seem a bit more complicated, yet it is basically the infrastructure for the preceding view. According to the relationship table *Orders*, each order (with OrderID as the primary key) corresponds to the information of a customer (CustomerID as the foreign key), ordering a product (ProductID as the foreign key) by contacting a salesperson (PersonID as the foreign key) who then processes the customer's order for a product.

  In this regard, there exists an m-n *contacts* relationship between the Customers table and the SalesPersons table, through which a customer can order products by contacting a salesperson. The salesperson then checks the product availability (which is neglected in the question) and then prepares a deal. Accordingly, the SalesPerson table has an m-n *processes* relationships with the Products table. This is represented as a simplified ER diagram in the Figure 1 below:

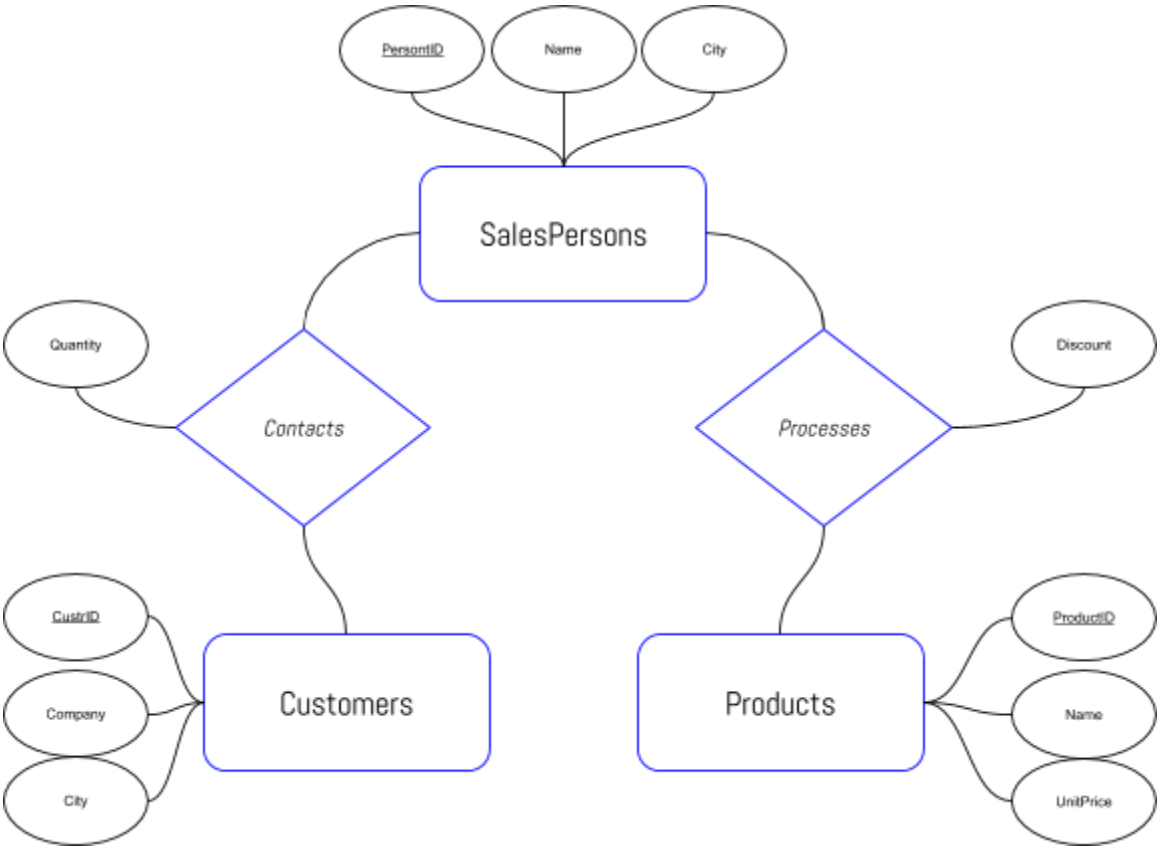


Figure 1. Relationships between different entities of concern

Of course, there is another way around this problem, in which the Products table is focal, and orders are not based on customers contacting a specific salesperson, but based on calls for a product which will later on be assigned to a salesperson. While both approaches are logically correct, the former is closer to real-world scenarios, and thus has been described here in details.

QI-2: Based on the relationships defined in QI-1, write a SQL query that shows the total sales amount by each sales person for each product.

```
SELECT  SalesPersons.Name AS SalesPersonName, Products.Name AS ProductName, SUM ((Orders.Quantity * Products.UnitPrice)*( 1.00 - (ISNULL(Orders.Discount, 0)))) AS SalesAmount FROM ((Orders INNER JOIN Products ON Orders.ProductID = Products.ProductID) INNER JOIN SalesPersons ON Orders.PersonID = SalesPersons.PersonID) GROUP BY SalesPersons.Name, Products.Name
```

QI-3: Based on the relationships defined in QI-1, write a SQL query that shows all the products ordered by IBM, including number of units and total sales amount.

```
SELECT  Products.Name AS ProductName, SUM(Orders.Quantity) TotalOrderQuantity, SUM ((Orders.Quantity * Products.UnitPrice)*( 1.00 - (ISNULL(Orders.Discount, 0)))) AS TotalSales FROM ((Orders INNER JOIN Customers ON Orders.CustomerID = Customers.CustomerID) INNER JOIN Products ON Orders.ProductID = Products.ProductID) WHERE Customers.Company = 'IBM' GROUP BY Products.Name
```

Part II: Now imagine you are hired as a business analyst.

QII-1: In an organization of disparate data sources, what would you do to deliver a successful dashboard initiative? If you need to make any assumptions, please state them. Make sure to explain why you would take each action in your plan.

To effectively and concisely answer this question, and for better comprehension, I decided to provide my considerations in the form of a list:

1. Dashboards are tools. Hence, the simpler a dashboard design is, the better. Poor layout choices negatively affects their users' performances. The components within the dashboard should be designed in a clear, light, and clean way. Also, misuse of color has a terrible effect of the dashboard design. Colors should be properly incorporated for alerting and notification, and for signifying importance or status. Clutter is a killer and enough white space is an important factor, so that maximum contrast is achieved. One needs to avoid to the best of one's' power the use of multiple pages and scrolling for a single dashboard.
2. Dashboards are cognitive tools. Hence, their points of foci and direction should be properly defined upfront. This helps with the breaking of tasks into simpler logical units and their flow, as well as the transitions between them, and thus minimizes the cognitive load of the user. To better facilitate this aspect, designers need to keep the dashboard as visual as possible and avoid textual overload, and also follow the Gestalt principles. Many of the best dashboards tell a story; they start with the bigger picture and allow the user to drill into the different aspects using simple patterns, so that the brain could determine values more quickly. The point is to avoid forcing the user to do the math and provide visual cognitive facilities for doing so.
3. Dashboards are (mainly) visual tools. Hence, the choices of visualizations are of utmost importance, and thus should be determined wisely and in accord with the focus and direction

of the cognitive utility, so that they can communicate the information most effectively, and not only by the best look.

4. Dashboards are interactive tools. Hence, their design should guarantee responsiveness, along with allowing the users to sort, filter, and organize the data. Using keyboard shortcuts is often a good suggestion. Also, feedback and help should be designed to aid the users in their interpretation. Since most of the modern dashboards are needed to be web-based, abilities such as linking to web-services such as research or update the data should be provided.
5. Dashboards are tools for the human users. Hence, their goal and audience should be determined properly, and the design should follow the end goal in mind. F possible, the audience should be polled and the design should follow the average users' needs. The key performance indicators (KPIs) should be clear to the designer ahead of time. Also their brand should be integrated to give the users a sense of connectivity.

Obviously there are many more considerations, but I tried to collectively provide those that are of more importance in this short space. I welcome the chances of discussing this topic in more detail if need be.

QII-2: How would you design the following dashboard with these requirements? A sketch on a piece of paper is acceptable, but be sure to be clear on what you are showing (it's open for interpretation for a reason, so do your best).

Metrics with sample data:

Sales by Month

SalesMonth	Sales
Dec. 1, 2008	$100,000.00
Jan. 1, 2009	$150,000.00
Feb. 1, 2009	$180,000.00

Target Sales by Month

SalesMonth	Sales Target
Dec. 1, 2008	$90,000.00
Jan. 1, 2009	$170,000.00
Feb. 1, 2009	$150,000.00

Rep Sales Performance

Rep	Revenue
John	$30,000.00
Harry	$50,000.00
Anna	$90,000.00

Sales by US region (NorthEast, NorthWest, SouthEast, SouthWest)

Region	Revenue
NorthEast	$130,000.00
NorthWest	$190,000.00
SouthEast	$220,000.00
SouthWest	$320,000.00

Sales by US region On a map, break down based on company, customer, etc

What other metrics do you think would be relevant with this dashboard?

In what follows, I will first discuss the possible visual modules. Then I will discuss the dashboard design. To lay down my analysis, I will assume that the company only works with customers in the four regions within US. Later on, I will revoke this assumption when answering the last question. To better represent my thoughts, I have used a free online tool, Datawrapper to create very simple and naive visualizations for the purpose of this analysis. Since the possibilities that this tool provides is limited, and also due to a lack of time, I have sacrificed the possibility of more complex examples. However, I would always welcome future discussions on the choice of visual structures of the data.

"Sales by Month" is an aggregate metric, i.e., it resembles the company sales aggregated on different salespersons and for different customers located in different regions. "Target Sales by Month" however is a baseline metric, which introduces a comparison threshold for "Sales by Month". Therefore, These two metrics can be visualized separately, and also in a single visualization in accord with one another. For example, the "Target Sales by Month" metric can be visualized as a line graph, similar to that in Figure 2 below, to demonstrate the changes in the minimum sales needed through time. This, in accord with the visualized data of some other origin, e.g., product availability, might shed some light on one's analyses.

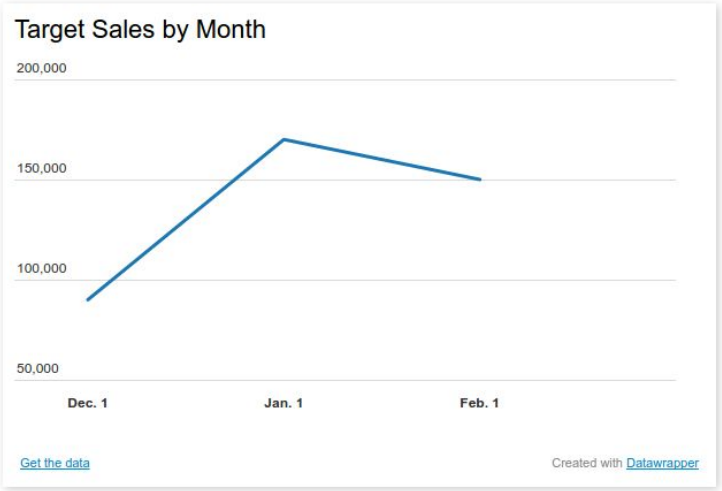


Figure 2. Variations of target sales by month

On the other hand, “Sales by Month” can be visualized as an aggregated metric, simply into a bar chart such as that in Figure 3.

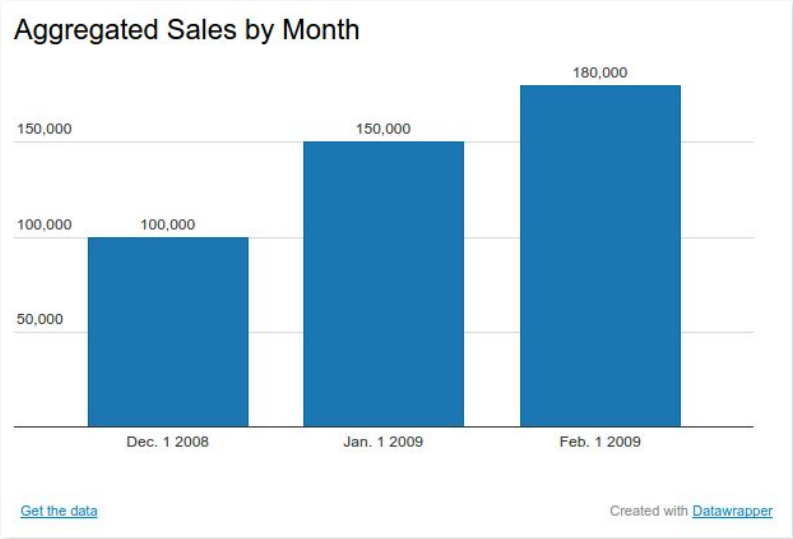


Figure 3. Aggregated sales by month

One can also design these two into one single visualization to provide better feedback. Although a bar chart/line chart duo is pretty standard, I am a proponent of other ways of demonstrating this baseline. In Figure 4 below, one way is demonstrated, as green lines above or red lines beneath the bars represent whether the target sales has been met or not.

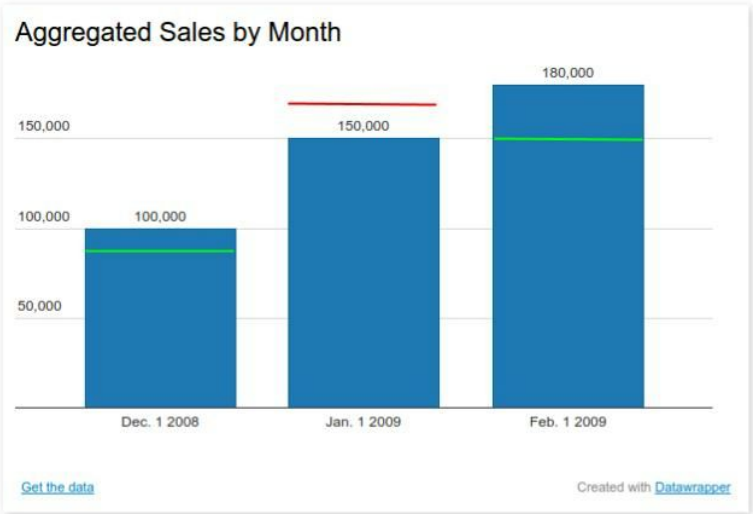


Figure 4. Baseline values are revealed upon interaction

The the baseline values can be shown on demand, i.e., upon interaction with the visualization. Interactive visualizations introduce many advantages into the design of dashboards, if they are properly operationalized. It is also worth mentioning that each one of the aforementioned

visualizations can interactively translate into other visual structures, if the data lays itself onto the new structure. As an example, if we have the specific data for the sales by month for the four regions, or for the three different salespersons, one can design the bar chart to either a stacked bar chart, or a grouped bar chart, depending on the user's desire[1]. In the case of the former, each stacked part of each bar can represent a region, or each bar within a group can represent a salesperson. This can be seen in Figure 5.

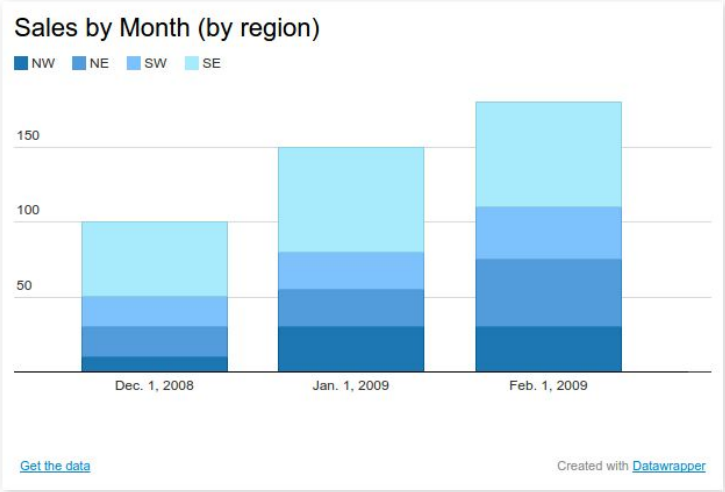
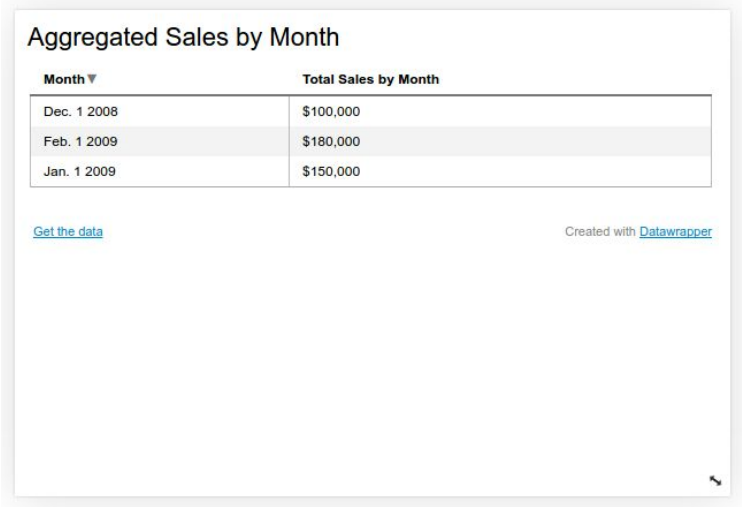


Figure 5. Interaction with the bar chart results in a breakdown of sales in different US regions

It is also possible for the user to see a tabular view of the data upon calling a procedure, e.g., on click on a table icon in the bottom right corner of the visualization, to turn the visualization 180 degrees and show a table as demonstrated in Figure 6.



[1] It is important to mention that even though this is not the best visual representation for this data, it is simply presented here for its simplicity and to follow the same basic visual structure.

Figure 6. Tabular view of the bar chart in Figure 4

A very similar logic may apply to “Rep Sales Performance”, where the aggregate revenue for each salesperson can be visualized over a fiscal year, and for customers within different regions, using a specific type of visual representation such as a sunburst or a treemap[2]. Then, interaction techniques can provide the ability to drill into subsets of the data, or to change the view. For example, one might be interested in seeing if Harry has most of his sales in the first quarter of the fiscal year or not, or whether he has sold more to customers from NW or NE. If we have the location of the salespersons as well, we can pursue more complicated hypotheses such as this: “How was Anna’s revenue prior to moving from NW to NE? Is she still selling to NW customers due to her connections with the customers in that region?” Another example can be the salesperson with highest revenue in each state, as in Figure 7.

Figure7. The map shows that Harry is the most valuable salesperson in New York.

To visually represent “Sales by US region”, and since this is an important aspect of the sales data, a map visualization can be very effective. The map can encode the aggregated sales using color. In addition, an interactive map can help perform many interesting analyses. For instance, each region

[2] Since these examples are not available through Datawrapper, I am not providing example figures of them in this documen.

can be drilled inside, so that the data for major cities be represented as well. In addition, one might be interested in the top 3 or last 5 customers, or the most consistent salesperson in each region.

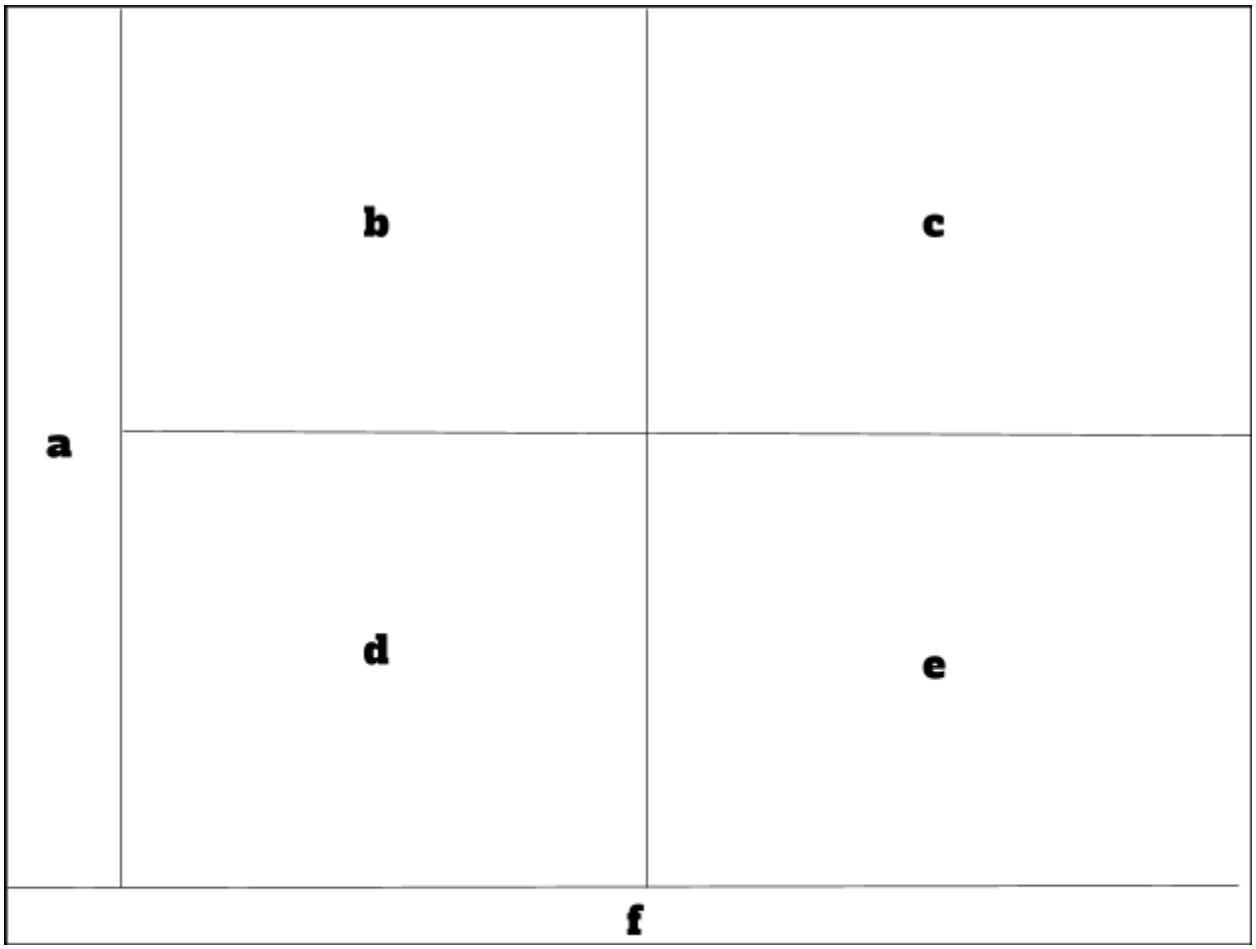


Figure 8. The abstract design structure of the dashboard

The dashboard will have a structure similar to that in Figure 8. Different sections of this dashboard are:

a: the filters dock, which can be easily accessible and then hides automatically to give more space to the dashboard area. Data binding processes and different choices of visualizations can also be found here.

b, c, d, e: each panel includes a visualization of data, as selected by the user. For example, one can initiate the stacked bar chart in panel (e).

f: the textual insight box. At each time and based on the current choice of visualization and the data bound to it, a summary information will also be provided in the textual form to provide quick insight. This can include examples such as top cities, salespersons or customers within a region, top products within a fiscal year, most important quarters, etc.

It is imperative to mention that this is the visual data part of the dashboard, and space will be allocated for necessary options such as company name and logo.