

-- DAMG 7275 Lab 3

-- Question 1 (2 points)

```
/*  
Write a SQL query using "FOR JSON PATH" and AdventureWorks2017  
to get the total sales a salesperson has made. Use TotalDue in  
SalesOrderHeader to calculate the total sales. Return the data  
in the format described below.
```

Use the provided format only for formatting purposes. Don't use
the data contained in the format for validation purposes.

Submit the code.

```
*/
```

```
/*  
{"SalesPersonID":280,"TotalSales":3748246},  
{"SalesPersonID":281,"TotalSales":7259568},  
{"SalesPersonID":282,"TotalSales":6683537},  
{"SalesPersonID":283,"TotalSales":4207895},  
{"SalesPersonID":284,"TotalSales":2608116}  
*/
```

```
/*  
Import the generated data into the Cosmos DB SQL API database.  
Submit a screenshot of importing results.  
*/
```

```
/*  
Write a SQL query for the Cosmos DB SQL API to get  
the sum, average, maximum and minimum of all of  
the total sales made by all salespersons.
```

Submit the code and a screenshot of the executing results.

```
*/
```

-- Question 2 (3 points)

```
/*  
Write a SQL query using "FOR JSON PATH" and AdventureWorks2017  
to get the products an order contains. Return the data  
in the format described below. Return data only for orders  
in the sales order id range between 43660 and 43680.
```

Use the provided format only for formatting purposes. Don't use
the data contained in the format for validation purposes.

Submit the code.

```
*/
```

```

/*
[{"SalesOrderID":43660,
  "Products":[{"ProductID":762,"OrderQty":1},
              {"ProductID":758,"OrderQty":1}]},
{"SalesOrderID":43661,
  "Products":[{"ProductID":745,"OrderQty":1},
              {"ProductID":743,"OrderQty":1},
              {"ProductID":747,"OrderQty":2},
              {"ProductID":712,"OrderQty":4},
              {"ProductID":715,"OrderQty":4},
              {"ProductID":742,"OrderQty":2},
              {"ProductID":775,"OrderQty":3},
              {"ProductID":778,"OrderQty":2},
              {"ProductID":711,"OrderQty":2},
              {"ProductID":741,"OrderQty":2},
              {"ProductID":776,"OrderQty":4},
              {"ProductID":773,"OrderQty":2},
              {"ProductID":716,"OrderQty":2},
              {"ProductID":777,"OrderQty":2},
              {"ProductID":708,"OrderQty":5}]}

*/

/*
  Import the generated data into the Cosmos DB SQL API database.
*/

/*
  Write a SQL query for the Cosmos DB SQL API to get
  the number of unique products contained in each order.
*/

-- Question 3 (3 points)

/*
  Write a SQL query using "FOR JSON PATH" and AdventureWorks2017
  to get the top 3 products in a sales territory and a product's
  total sold quantity. Return the data in the format described
  below. TotalProductQuantity is the total sold quantity of a
  product.

  The top 3 products have the 3 highest total sold quantities
  in a territory. If there is a tie, the tie needs to be retrieved.
  The product quantity of an order is stored in SalesOrderDetail.

  Use the provided format only for formatting purposes. Don't use
  the data contained in the format for validation purposes.

  Submit the code.
*/

```

```
/*
[{"TerritoryID":1,
  "Top3Products":[{"ProductID":870,"TotalProductQuantity":947},
                  {"ProductID":712,"TotalProductQuantity":887},
                  {"ProductID":711,"TotalProductQuantity":794}]},
{"TerritoryID":2,
  "Top3Products":[{"ProductID":715,"TotalProductQuantity":549},
                  {"ProductID":712,"TotalProductQuantity":514},
                  {"ProductID":711,"TotalProductQuantity":423}]}
*/
```

```
/*
  Import the generated data into a Cosmos DB SQL API database.
  Submit a screenshot of importing results.
*/
```

```
/*
  Write a SQL query for the Cosmos DB SQL API to get
  the totals of TotalProductQuantity regardless of
  the sales territory for each product.
  Submit the code and a screenshot of the executing results.
*/
```