

-- Lab 1

/\* Review of the Data Hierarchy, Data Aggregation and Horizontal Reporting \*/

/\* Use AdventureWorks2017 database \*/

/\* The following code creates a longlist report. \*/

```
SELECT DISTINCT h.SalesPersonID, p.FirstName, p.LastName, d.ProductID
FROM Sales.SalesOrderHeader h join Sales.SalesOrderDetail d on
d.SalesOrderID=h.SalesOrderID
JOIN Person.Person p
ON h.SalesPersonID = p.BusinessEntityID
ORDER BY h.SalesPersonID, ProductID;
```

/\* We want to create a shortlist report containing the same data without duplicating the salesperson info. The code below tried to do it, but the generated report had an issue. It still duplicated some salesperson info. \*/

```
select distinct sh.SalesPersonID,
pp.FirstName,pp.LastName,
stuff((select ', ' , rtrim (cast(ProductID as char))
      from Sales.SalesOrderDetail      where
SalesOrderID= sh.SalesOrderID          order by
sd.ProductID                          for xml path('')), 1,2, '' )
as Product
from Sales.SalesOrderHeader sh join
Sales.SalesOrderDetail sd on
sd.SalesOrderID=sh.SalesOrderID join
Person.Person pp
on pp.BusinessEntityID = sh.SalesPersonID orderby
sh.SalesPersonID;
```

/\* Please investigate the issue and find out what caused the issue. Give a detailed explanation of your findings. Writing some code to prove your explanation would be convincing. \*/

There is a duplication problem which is caused by the join of Sales.SalesOrderDetail to the Sales.SalesOrderHeader. It should have been Sale.SalesOderHeader joined to the Person.Person table. This can be resolve by changing the joining of the table

**Correct Code :-**

```
STUFF((SELECT ', ' +
CAST(sd.ProductID AS VARCHAR(10))
```

```
FROM Sales.SalesOrderHeader sh
JOIN Person.Person pp ON sh.SalesPersonID = pp.BusinessEntityID
```

Rucha Chotalia  
NUID:002711888

/\* Modify the incorrect solution provided above to create the correct report. \*/

```
SELECT DISTINCT sh.SalesPersonID, pp.FirstName, pp.LastName,  
    STUFF((SELECT ', ' + CAST(sd.ProductID AS VARCHAR(10))  
        FROM Sales.SalesOrderDetail sd  
        WHERE sd.SalesOrderID IN (SELECT sh1.SalesOrderID  
        FROM Sales.SalesOrderHeader sh1 WHERE sh1.SalesPersonID = sh.SalesPersonID)  
        ORDER BY sd.ProductID  
        FOR XML PATH(''), 1, 2, '' ) AS Product  
FROM Sales.SalesOrderHeader sh  
JOIN Person.Person pp ON sh.SalesPersonID = pp.BusinessEntityID  
ORDER BY sh.SalesPersonID;
```

/\* Use STRING\_AGG to create the same report \*/

```
select sh.SalesPersonID, pp.FirstName, pp.LastName,  
    STRING_AGG(cast(sd.ProductID as NVARCHAR(MAX)), ', ') as Products  
from Sales.SalesOrderHeader sh  
join Sales.SalesOrderDetail sd on sd.SalesOrderID = sh.SalesOrderID  
join Person.Person pp on pp.BusinessEntityID = sh.SalesPersonID  
group by sh.SalesPersonID, pp.FirstName, pp.LastName  
order by sh.SalesPersonID;
```