SQL assignment

Part 1

```sql
SELECT TOP 10 *
FROM Production.Product

SELECT TOP 10 *
FROM Production.WorkOrder

SELECT TOP 10 *
FROM Production.ScrapReason

SELECT pw.ProductID, pp.Name AS ProductName, ps.Name AS ScrapReason
INTO #RawData
FROM Production.Product AS pp
INNER JOIN Production.WorkOrder AS pw
ON pp.ProductID = pw.ProductID
INNER JOIN Production.ScrapReason AS ps
ON pw.ScrapReasonID = ps.ScrapReasonID

SELECT *
FROM #RawData

SELECT ProductID,ProductName,COUNT(ProductID)AS WorkOrderCount, ScrapReason
FROM #RawData
GROUP BY ProductID,ProductName, ScrapReason
```

Part 2

```sql
/* In this part, we join the tables and connect ProductID with catagory.*/
SELECT Product.ProductID,ProductCategory.Name
INTO #ProductCata
FROM Production.Product
INNER JOIN Production.ProductSubcategory
ON ProductSubcategory.ProductSubcategoryID = Product.ProductSubcategoryID
INNER JOIN Production.ProductCategory
ON ProductCategory.ProductCategoryID = ProductSubcategory.ProductCategoryID

SELECT * FROM #ProductCata

SELECT  *
FROM Sales.SalesOrderHeader

/* In this part ,we join the tables together and form a new table that contain
information about time, salesOrderID(as the key), onlineflag,
and product catagory*/
SELECT *
INTO #RAWDATA
FROM (
SELECT YEAR(SalesOrderHeader.OrderDate) AS FY,
MONTH(SalesOrderHeader.OrderDate) AS M,
DAY(SalesOrderHeader.OrderDate) AS D,
SalesOrderHeader.SalesOrderID,
SalesOrderHeader.OnlineOrderFlag,
#ProductCata.Name,
SalesOrderDetail.OrderQty
FROM Sales.SalesOrderHeader
INNER JOIN Sales.SalesOrderDetail
ON SalesOrderHeader.SalesOrderID = SalesOrderDetail.SalesOrderID
INNER JOIN #ProductCata
ON SalesOrderDetail.ProductID = #ProductCata.ProductID
) AS T

SELECT * FROM #RAWDATA

SELECT FY, M,D, SalesOrderID,OnlineOrderFlag, Name AS Catagory, SUM(OrderQty)
FROM #RAWDATA
GROUP BY FY,M,D, SalesOrderID, OnlineOrderFlag,Name
ORDER BY SalesOrderID

/*In this part, we use pivot to get information about sales of different product
catagories.*/
SELECT *
INTO #RAWDATA2
FROM(
SELECT FY, M,D,SalesOrderID,OnlineOrderFlag, Name AS Catagory, SUM(OrderQty) AS Qty
FROM #RAWDATA
GROUP BY FY,M,D, SalesOrderID, OnlineOrderFlag,Name
) AS TBD
PIVOT (SUM(Qty) FOR Catagory IN (Accessories,Bikes,Clothing,Components)) AS UN

SELECT * FROM #RAWDATA2

/*In this part, as quantity of products been sold is not considered important, we get
rid of null and turn the quantity of product sold into 1 and 0. */
SELECT FY,
M,
D,
SalesOrderID,
ISNULL(Accessories,0) AS Accessories,
ISNULL(Bikes,0)AS Bikes,
ISNULL(Clothing,0)AS Clothing,
ISNULL(Components,0)AS Components,
OnlineOrderFlag
INTO #RAWDATA3
FROM #RAWDATA2

SELECT * FROM #RAWDATA3

SELECT FY,
M,
D,
SalesOrderID,
Bikes,
Clothing,
Components,
OnlineOrderFlag,
CASE WHEN Accessories > 0 THEN 1
ELSE 0
END Accessories
INTO #RAWDATA4
FROM #RAWDATA3

SELECT FY,
M,
D,
SalesOrderID,
Accessories,
Clothing,
Components,
OnlineOrderFlag,
CASE WHEN Bikes > 0 THEN 1
ELSE 0
END Bikes
INTO #RAWDATA5
FROM #RAWDATA4

SELECT FY,
M,
D,
SalesOrderID,
Accessories,
Bikes,
Components,
OnlineOrderFlag,
CASE WHEN Clothing > 0 THEN 1
ELSE 0
END Clothing
INTO #RAWDATA6
FROM #RAWDATA5

SELECT FY,
M,
D,
SalesOrderID,
Accessories,
Bikes,
Clothing,
OnlineOrderFlag,
CASE WHEN Components> 0 THEN 1
ELSE 0
END Components
INTO #RAWDATA7
FROM #RAWDATA6

/*In this part, as American standard of fiscal year should be used, we turn natural
date into fiscal date.*/
SELECT CAST(FY AS varchar)+’-’+CAST(M AS varchar)+’-’+CAST(D AS varchar) AS OrderDate,
SalesOrderID,
Accessories,
Bikes,
Clothing,
Components,
OnlineOrderFlag
INTO #RAWDATA8
FROM #RAWDATA7

SELECT
OrderDate,
SalesOrderID,
Accessories,
Bikes,
Clothing,
Components,
OnlineOrderFlag,
CASE
WHEN MONTH(OrderDate) BETWEEN 7 AND 12 THEN YEAR(OrderDate)+1
```

```sql
WHEN MONTH(OrderDate) BETWEEN 1 AND 6 THEN YEAR(OrderDate)
END FY
INTO #RAWDATA9
FROM #RAWDATA8

SELECT FY,
SalesOrderID,
Accessories,
Bikes,
Clothing,
Components,
OnlineOrderFlag,
CASE
WHEN MONTH(OrderDate) BETWEEN 7 AND 9 THEN 1
WHEN MONTH(OrderDate) BETWEEN 10 AND 12 THEN 2
WHEN MONTH(OrderDate) BETWEEN 1 AND 3 THEN 3
WHEN MONTH(OrderDate) BETWEEN 4 AND 6 THEN 4
END FQ
INTO #RAWDATA10
FROM #RAWDATA9

/*In this part, we want to consider online sales and offline sales separatetly, so we
use where to separate the table.*/
SELECT *
INTO #RAWDATA_OFF
FROM #RAWDATA10
WHERE OnlineOrderFlag = 0
 SELECT * FROM #RAWDATA_OFF
SELECT *
INTO #RAWDATA_ON
FROM #RAWDATA10
WHERE OnlineOrderFlag = 1

/*In this part, we count the orders of online and offline.*/
SELECT DISTINCT
FY,
FQ,
Accessories,
Bikes,
Clothing,
Components,
COUNT(SalesOrderID) AS OfflineOrders
INTO #DATA_OFF
FROM #RAWDATA_OFF
```

```
GROUP BY FY,FQ,Accessories,Bikes,Clothing,Components

SELECT DISTINCT
FY,
FQ,
Accessories,
Bikes,
Clothing,
Components,
COUNT(SalesOrderID) AS OnlineOrders
INTO #DATA_ON
FROM #RAWDATA_ON
GROUP BY FY,FQ,Accessories,Bikes,Clothing,Components

/*In this part, we join the online table and offline table together and handle the NULL
parts, finally and hopefully, we get the right result.*/
SELECT * FROM #DATA_OFF
FULL OUTER JOIN #DATA_ON
ON #DATA_OFF.FY = #DATA_ON.FY
AND #DATA_OFF.FQ = #DATA_ON.FQ
AND #DATA_OFF.Accessories = #DATA_ON.Accessories
AND #DATA_OFF.Bikes = #DATA_ON.Bikes
AND #DATA_OFF.Clothing = #DATA_ON.Clothing
AND #DATA_OFF.Components = #DATA_ON.Components

SELECT
    ISNULL(a.FY,b.FY) AS FY,
    ISNULL(a.FQ,b.FQ) AS FQ,
    ISNULL(a.Accessories,b.Accessories) AS Accessories,
    ISNULL(a.Bikes,b.Bikes) AS Bikes,
    ISNULL(a.Clothing,b.Clothing) AS Clothing,
    ISNULL(a.Components,b.Components)AS Components,
    ISNULL(a.OfflineOrders,0) AS OfflineOrders,
    ISNULL(b.OnlineOrders,0) AS OnlineOrders
FROM #DATA_OFF AS a
FULL OUTER JOIN #DATA_ON AS b
ON a.FY = b.FY
AND a.FQ = b.FQ
AND a.Accessories = b.Accessories
AND a.Bikes = b.Bikes
AND a.Clothing = b.Clothing
AND a.Components = b.Components
```