

## Chapter 14

### Understanding Basic SQL

#### Making Sense of Joins

##### Inner Joins

###### Bad SQL Code:

```
SELECT Region, Market, AccountManagerID, FullName  
FROM Dim_AccountManagers INNER JOIN Dim_Territory  
ON Dim_AccountManagers.AccountManagerID = Dim_Territory.AccountManagerID
```

###### Correction:

```
SELECT Region, Market, Dim_AccountManagers.AccountManagerID, FullName  
FROM Dim_AccountManagers INNER JOIN Dim_Territory  
ON Dim_AccountManagers.AccountManagerID = Dim_Territory.AccountManagerID
```

## Chapter 14

### Understanding Basic SQL

#### Making Sense of Joins

##### Outer Joins

###### Bad SQL Code:

```
SELECT Region, Market, .AccountManagerID, FullName  
FROM Dim_AccountManagers LEFT JOIN Dim_Territory  
ON Dim_AccountManagers.AccountManagerID = Dim_Territory.AccountManagerID
```

###### Correction:

```
SELECT Region, Market, Dim_AccountManagers.AccountManagerID, FullName  
FROM Dim_AccountManagers LEFT JOIN Dim_Territory  
ON Dim_AccountManagers.AccountManagerID = Dim_Territory.AccountManagerID
```

###### Bad SQL Code:

```
SELECT Region, Market, AccountManagerID, FullName  
FROM Dim_AccountManagers RIGHT JOIN Dim_Territory  
ON Dim_AccountManagers.AccountManagerID = Dim_Territory.AccountManagerID
```

###### Correction:

```
SELECT Region, Market, Dim_AccountManagers.AccountManagerID, FullName  
FROM Dim_AccountManagers RIGHT JOIN Dim_Territory  
ON Dim_AccountManagers.AccountManagerID = Dim_Territory.AccountManagerID
```

## Chapter 14

### Getting Fancy with Advanced SQL Statements

#### Grouping and Aggregating with the GROUP BY Clause

##### Bad Text:

For instance, this SQL statement will group the records where the value in the Market field is Dallas, then only return those records where the grouped average LineItem is less than 100. Again, the grouping is done before checking if the average LineItem is less than 100.

```
SELECT Market, Count(State)
FROM Dim_Territory
WHERE Market = "Dallas"
GROUP BY Market
HAVING Avg(LineItem) < 100
```

##### Correction:

For instance, this SQL statement will group the records where the value in the Market field is Dallas, then only return those customer records where the sum of Revenue is less than 100. Again, the grouping is done before checking if the sum of Revenue is less than 100.

```
SELECT Customer_Name, Sum(Revenue) AS Sales
FROM PvTblFeed
Where Market = "Dallas"
GROUP BY Customer_Name
HAVING (Sum(Revenue)<100)
```