Solution: Reading SQL

Statement 1

```sql
SELECT [SalesOrderID]
      ,[OrderDate]
      ,[ShipDate]
      ,[DueDate]
      ,[SalesOrderNumber]
      ,[PurchaseOrderNumber]
      ,[AccountNumber]
      ,[CustomerID]
FROM [AdventureWorks2017].[Sales].[SalesOrderHeader]
ORDER BY [OrderDate], [DueDate]
```

Statement 1 Explanation

Lists the fields in the order of the select statement from a table named SalesOrderHeader. It then sorts the information starting with OrderDate and then DueDate.

Statement 2

```sql
SELECT [SalesOrderID]
      ,[OrderDate]
      ,[ShipDate]
      ,[DueDate]
      ,[SalesOrderNumber]
      ,[PurchaseOrderNumber]
      ,[AccountNumber]
      ,[CustomerID]
FROM [AdventureWorks2017].[Sales].[SalesOrderHeader]
ORDER BY [OrderDate], [DueDate]
WHERE [CustomerID] in ('545','689','755')
```

Statement 2 Corrected

```sql
SELECT [SalesOrderID]
      ,[OrderDate]
      ,[ShipDate]
      ,[DueDate]
      ,[SalesOrderNumber]
      ,[PurchaseOrderNumber]
      ,[AccountNumber]
      ,[CustomerID]
FROM [AdventureWorks2017].[Sales].[SalesOrderHeader]
WHERE [CustomerID] in ('545','689','755')
ORDER BY [OrderDate], [DueDate]
```

Statement 2 Explanation

Correct the order of the ORDER BY and WHERE statements. You must filter before you sort. Once corrected, this statement lists the fields in this order from a table named SalesOrderHeader, only shows customers with IDs that are 545, 689 and 755, and then sorts by OrderDate and DueDate.