Generating an Accumulating Column within Multiple Groups

The **sashelp.shoes** table contains sales information for various products in each region and subsidiary. Numbers for sales and returns are recorded for each row. Create a summary table that includes the sum of **Profit** for each region and product.

1. Create a new program.

   - Create a sorted copy of **sashelp.shoes** that is ordered by **Region** and **Product**.
   - Use the DATA step to read the sorted table and create a new table named **profitsummary**.
   - Create a column named **Profit** that is the difference between **Sales** and **Returns**.
   - Create an accumulating column named **TotalProfit** that is a running total of **Profit** within each value of **Region** and **Product**.
   - Reset **TotalProfit** for each new combination of **Region** and **Product**.
   - Submit the program and verify that **TotalProfit** is accurate.

```
proc sort data=sashelp.shoes out=sort_shoes;
    by Region Product;
run;

data profitsummary;
    set sort_shoes;
    by Region Product;
    Profit=Sales-Returns;
    if first.Product then TotalProfit=0;
    TotalProfit+Profit;
    format TotalProfit dollar12.;
run;
```

2. How many rows and columns are in the **profitsummary** table?

   The **profitsummary** table has 395 rows and nine columns.

3. Modify the DATA step.

   - Include only the last row for each **Region** and **Product** combination.
   - Keep **Region**, **Product**, and **TotalProfit**.
   - Format **TotalProfit** as a currency value.
   - Submit the program and examine the output data.

```
data profitsummary;
    set sort_shoes;
    by Region Product;
    Profit=Sales-Returns;
    if first.Product then TotalProfit=0;
    TotalProfit+Profit;
    if last.Product=1;
    keep Region Product TotalProfit;
    format TotalProfit dollar12.;
run;
```

4. How many rows and columns are in the **profitsummary** table?

The **profitsummary** table has 80 rows and three columns.