Step 1: Data Extraction

In order to carry out the analysis, the following data elements are needed. All data required for answering the research question are included in the database associated with this textbook.

- A list of all nursing homes in the United States
- An indicator as to whether the nursing home is for-profit, nonprofit, or government owned
- An indicator as to whether the nursing home is located in a hospital
- A measure of adjusted nurse staffing hours per day per nursing home resident
- A measure of adjusted RN staffing hours per day per nursing home resident

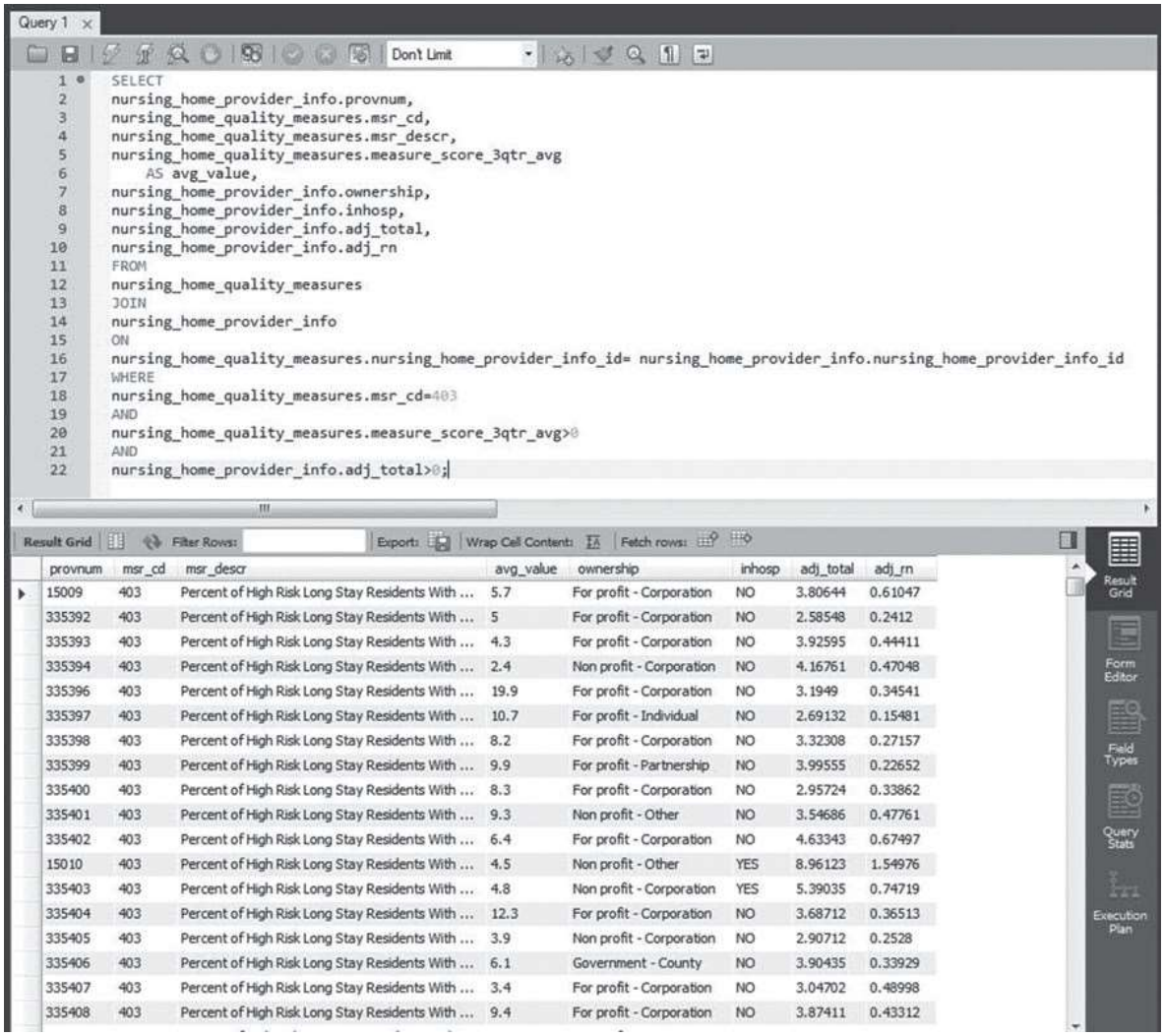
Two data sets need to be combined to obtain the needed data to answer the proposed research question. The first data set is named nursing_home_provider_info, and the second data set is named nursing_home_quality_measures.

The two data sets will be combined using MySQL scripts (see **figure 13.1**). The scripts will be constructed so that only the following data elements are extracted: nursing home NPI, quality measure code and description, detailed nursing home ownership and ownership group, whether the nursing home is located within a hospital, adjusted nurse staffing hours per resident per day, adjusted RN staffing hours per resident per day (**figure 13.1**).

Figure 13.1 shows the SQL script that combines the two required data sets and extracts only the relevant data elements. For the purpose of this study, two tables of data will be combined: nursing_home_quality_measures and nursing_home_provider_info. In this query, we will be selecting four columns of data. From the nursing_home_provider_info table, provnum, ownership, inhosp, adj_total, and adj_rn are selected. From the nursing_home_quality_measures table, msr_cd, msr_descr, and measure_score_3qtr_avg are selected. In our query, we can change the name of the measure_score_3qtr_avg column by including the AS after selecting the column. After the AS function, you are required to specify the new name of the column. A JOIN is also required in order to combine the two tables of data. The ON function specifies which two fields of data in our two tables match. This used to JOIN the two tables by finding matches between the matching fields. In this case, the

patching field is nursing_home_provider_info_id. Finally, the WHERE statement is used to specify that we only want to query data when msr_cd is equal to 403, measure_score_3qtr_avg is greater than zero, and adj_total is greater than zero. The top section of **figure 13.1** depicts the MySQL script that was used to obtain the data. The bottom section of the figure depicts the first few rows of the output that was derived from the query.

```
SELECT
nursing_home_provider_info.provnum,
nursing_home_quality_measures.msr_cd,
nursing_home_quality_measures.msr_descr,
nursing_home_quality_measures.measure_score_3qtr_avg
    AS avg_value,
nursing_home_provider_info.ownership,
nursing_home_provider_info.inhosp,
nursing_home_provider_info.adj_total,
nursing_home_provider_info.adj_rn
FROM
nursing_home_quality_measures
JOIN
nursing_home_provider_info
ON
nursing_home_quality_measures.nursing_home_provider_info_id= nursing_home_
provider_info.nursing_home_provider_info_id
WHERE
nursing_home_quality_measures.msr_cd=403
AND
nursing_home_quality_measures.measure_score_3qtr_avg>0
AND
nursing_home_provider_info.adj_total>0;
```



```
SELECT
nursing_home_provider_info.provnum,
nursing_home_quality_measures.msr_cd,
nursing_home_quality_measures.msr_descr,
nursing_home_quality_measures.measure_score_3qtr_avg
    AS avg_value,
nursing_home_provider_info.ownership,
nursing_home_provider_info.inhosp,
nursing_home_provider_info.adj_total,
nursing_home_provider_info.adj_rn
FROM
nursing_home_quality_measures
JOIN
nursing_home_provider_info
ON
nursing_home_quality_measures.nursing_home_provider_info_id= nursing_home_provider_info.nursing_home_provider_info_id
WHERE
nursing_home_quality_measures.msr_cd=403
AND
nursing_home_quality_measures.measure_score_3qtr_avg>0
AND
nursing_home_provider_info.adj_total>0;
```

provnum	msr_cd	msr_descr	avg_value	ownership	inhosp	adj_total	adj_rn
15009	403	Percent of High Risk Long Stay Residents With ...	5.7	For profit - Corporation	NO	3.80644	0.61047
335392	403	Percent of High Risk Long Stay Residents With ...	5	For profit - Corporation	NO	2.58548	0.2412
335393	403	Percent of High Risk Long Stay Residents With ...	4.3	For profit - Corporation	NO	3.92595	0.44411
335394	403	Percent of High Risk Long Stay Residents With ...	2.4	Non profit - Corporation	NO	4.16761	0.47048
335396	403	Percent of High Risk Long Stay Residents With ...	19.9	For profit - Corporation	NO	3.1949	0.34541
335397	403	Percent of High Risk Long Stay Residents With ...	10.7	For profit - Individual	NO	2.69132	0.15481
335398	403	Percent of High Risk Long Stay Residents With ...	8.2	For profit - Corporation	NO	3.32308	0.27157
335399	403	Percent of High Risk Long Stay Residents With ...	9.9	For profit - Partnership	NO	3.99555	0.22652
335400	403	Percent of High Risk Long Stay Residents With ...	8.3	For profit - Corporation	NO	2.95724	0.33862
335401	403	Percent of High Risk Long Stay Residents With ...	9.3	Non profit - Other	NO	3.54686	0.47761
335402	403	Percent of High Risk Long Stay Residents With ...	6.4	For profit - Corporation	NO	4.63343	0.67497
15010	403	Percent of High Risk Long Stay Residents With ...	4.5	Non profit - Other	YES	8.96123	1.54976
335403	403	Percent of High Risk Long Stay Residents With ...	4.8	Non profit - Corporation	YES	5.39035	0.74719
335404	403	Percent of High Risk Long Stay Residents With ...	12.3	For profit - Corporation	NO	3.68712	0.36513
335405	403	Percent of High Risk Long Stay Residents With ...	3.9	Non profit - Corporation	NO	2.90712	0.2528
335406	403	Percent of High Risk Long Stay Residents With ...	6.1	Government - County	NO	3.90435	0.33929
335407	403	Percent of High Risk Long Stay Residents With ...	3.4	For profit - Corporation	NO	3.04702	0.48998
335408	403	Percent of High Risk Long Stay Residents With ...	9.4	For profit - Corporation	NO	3.87411	0.43312

Figure 13.1. **SQL script and returned data for combining two data sets**

Note that you must alter a setting in MySQL so that the output shows all of the results rather than just showing the first 1,000 rows (see chapter 4). After executing the query, export the data to a CSV file and save the file as nurse_staff. For detailed instructions regarding exporting data from MySQL Workbench to CSV format, see chapter 4.

Step 2: Data Preparation

Data preparation can be the most labor-intensive and, arguably, the most important step in the analytics process. The purpose of data preparation is to ensure the correct data, in the correct format, is used for the analysis. Open the nurse_staff.csv file that was exported from MySQL Workbench with Excel. You should see the following columns of data:

- provnum: The unique provider id
- msr_cd: A code for the performance measure
- msr_descr: A description of the performance measure
- avg_value: The percent of high-risk long-stay residents with pressure ulcers

206

207