

Hands on Labs – SQL Self-JOINS

These ungraded hands-on-lab will help you build your skills working with self-joins.

Your task is to take the highlighted information in the image below, and create a single SQL table that models the relationships between the highlighted business entities. You should populate the table with the highlighted rows, then write a SELECT statement that displays a result set that shows each business entity and its immediate parent.

List of assets owned by General Electric

From Wikipedia, the free encyclopedia

List of assets owned by **General Electric**: ✓

Primary business units [edit]

- **GE Capital** ✓
 - GE Capital Aviation Services ✓
 - GE Energy Financial Services ✓
 - GE Real Estate
 - GE Americas
 - GE Asia
 - GE Europe, Middle East & Africa
- **GE Energy Management** ✓
 - Industrial Solutions
 - GE Power Electronics ✓
 - GE Power Components ✓
 - GE Critical Power
 - GE Intelligent Platforms
 - Power Conversion
 - Digital Energy
- **GE Oil & Gas** ✓
 - GE Power & Water
 - GE Home & Business Solutions

Source: http://en.wikipedia.org/wiki/List_of_assets_owned_by_General_Electric

Here is one possible resultset (other answers could be reasonable, or even better, depending on intended purpose):

	parent character varying	subsidiary character varying
1	GE Capital	GE Capital Aviation Services
2	GE Capital	GE Energy Financial Services
3	GE Energy Management	GE Power Components
4	GE Energy Management	GE Power Electronics
5	General Electric	GE Capital
6	General Electric	GE Energy Management
7	General Electric	GE Oil and Gas

To consider:

- How many levels deep is it possible to “nest” data?
- Can you think of other “use cases” for organizing data hierarchically?
- Are there other choices besides SQL Server for modeling hierarchical data?