



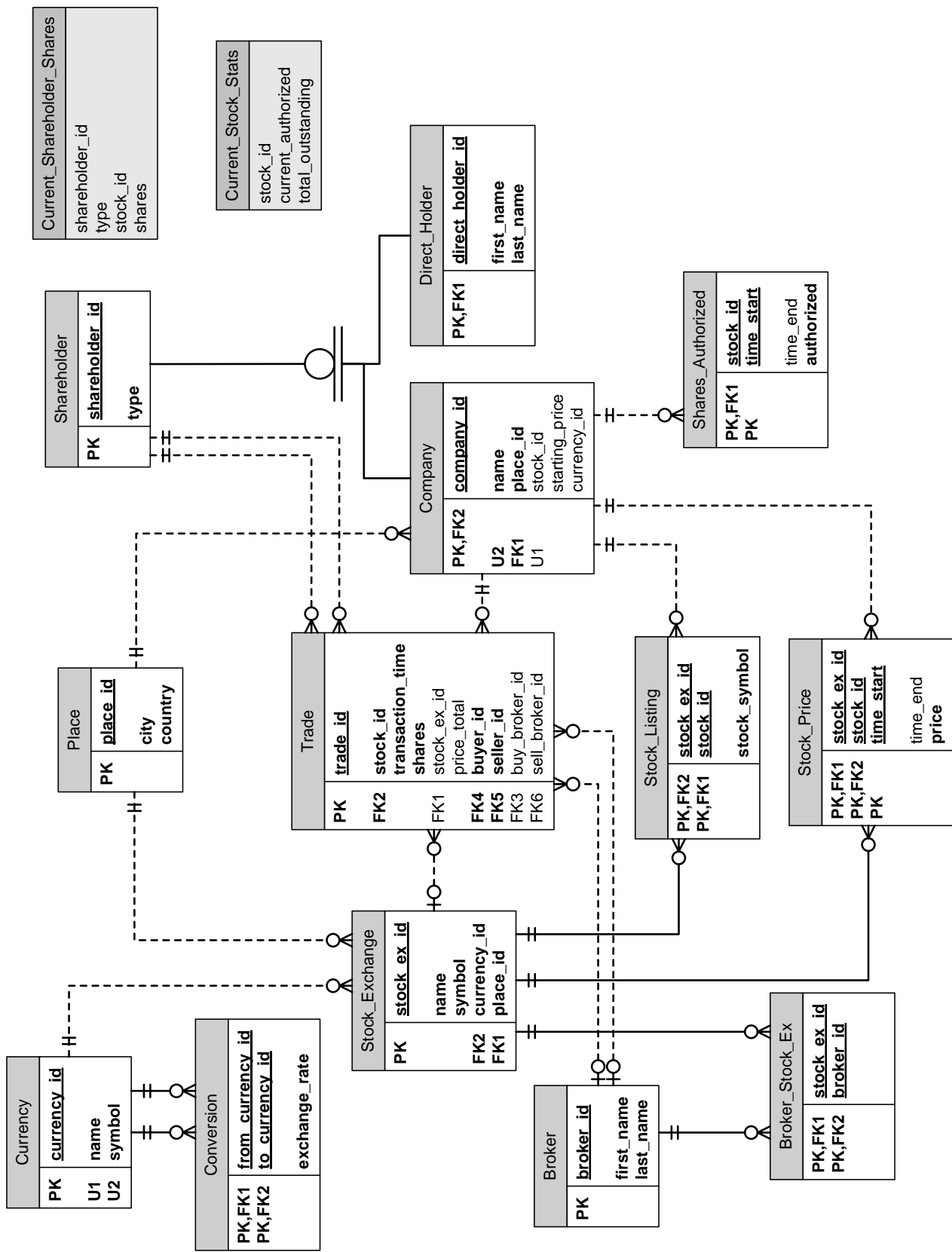
Core SQL

Part 2

Final Developer Project

Database Functionality and Reports

© FDM Group 2020. All Rights Reserved.
Any unauthorized reproduction or distribution in part
or in whole will constitute an infringement of copyright.



1. Project specification

It is recommended that you read the entire specification before you begin.

Basic Reports

Views

Two versions of two queries are provided. See QueriesProvided.sql.

1. Current_Shareholder_Shares – Two queries are provided in QueriesProvided.sql. Both of these queries list shareholder id, shareholder type, stock id, and the total shares currently held by the shareholder. Create a view called CURRENT_SHAREHOLDER_SHARES using the more efficient query and include the view in your script. Write a *detailed* comment explaining why you chose the query. Cite the exact numbers of the appropriate run-time statistic that you are basing your choice on.
2. Current_Stock_Stats – These queries list each stock id, the number of shares currently authorized, and the total number of shares currently outstanding. Create a view called CURRENT_STOCK_STATS using the more efficient query and include it in your script. Write a *detailed* comment explaining why you chose the query. Cite the exact numbers of the appropriate run-time statistic that you are basing your choice on.

Queries

3. List the names of companies owned, as well as the number of shares authorized, the total shares outstanding, and outstanding shares as a % of authorized shares. Sort the output by company name.
4. For every direct holder, list the first and last name, the names of the companies this person owns, the number of shares currently held, % this holder has of the shares outstanding, and % this holder has of the total authorized shares. Sort the output by direct holder last name, first name, and company name. Display the percentages to two decimal places.
5. For every institutional holder - companies that own stock, list the name of the holder, the names of the companies invested in by this holder, number of shares currently held, % this holder has of the total shares outstanding, and % this holder has of the total authorized shares. Sort the output by holder name, and company owned name. Display the percentages to two decimal places.
6. Show any trade if more than 50000 shares were traded on the *secondary* markets. Display the trade id, stock symbol, name of the company being traded, stock exchange symbol, number of shares traded, price total and currency symbol. Sort the result by trade id.
7. For each stock listed at each stock exchange, display the exchange name, stock symbol and the transaction date *and time* when that stock was last traded. Sort the output by stock exchange name and stock symbol. When a stock has not been traded, nulls should appear in the date last traded.
8. Display the trade id, name of the company and number of shares for the single largest trade made on the secondary markets in terms of the number of shares traded. Unless there are multiple trades with the same number of shares traded, only one row should be returned.

Data Manipulation

Write the necessary INSERT, UPDATE and/or DELETE statements to complete the following data changes.

Add a Direct Holder

9. Add Jeff Adams as a new direct holder. You will have to insert a row into the shareholder table and make another insert into the direct_holder table.

Add an Institutional Holder

10. Add Makoto Investing as a new institutional holder that is located in Tokyo, Japan. Makoto does not currently have a stock id. A row must be inserted into the shareholder table and a corresponding row must be inserted into the company table.

Initial Public Offering (IPO)

11. Makoto Investing would like to declare stock. Today they will authorize 100,000 shares. The starting price will be 50 yen. You must update the company table, and insert a new row in the shares_authorized table.

Listing on an Exchange

12. Makoto Investing will list on the Tokyo Stock Exchange under the stock symbol TYO:8602. You need to insert into the stock_listing table and the stock_price table.

Commit all changes made thus far.

Stored Procedures

Write the necessary PL/SQL statements to create the following procedures. Include all statements that test your procedures.

Add a Direct Holder

13. Write a PL/SQL procedure called INSERT_DIRECT HOLDER which will be insert new direct holders. Create a sequence to automatically generate shareholder_ids. Use this sequence in your procedure.
-Input parameters: p_first_name, p_last_name

Add an Institutional Holder

14. Write a PL/SQL procedure called INSERT_COMPANY which will insert new companies. The stock_id for new companies will be null. Use the sequence that you created in problem 13 to insert shareholder_ids.
-Input parameters: p_company_name, p_city, p_country

Declare Stock (Initial Public Offering)

15. Write a PL/SQL procedure called DECLARE_STOCK which will be run when a company issues stock.
-Input parameters: company name, number of shares authorized, starting price in the company's currency, and currency name.
-Check to ensure the company has not already been given a stock id.
-If the company already has a stock id then *do not perform any data changes*. Use RAISE_APPLICATION_ERROR to send the user an error number, error message and stop processing.
-Otherwise, the company must be assigned a stock id and the date of issue - current system date, number of shares authorized, and the starting price. Create a sequence to generate the new stock_ids.

Listing on an Exchange

16. Write a PL/SQL procedure called LIST_STOCK which will be used when stock is listed on a stock exchange.
-Input parameters: p_stock_id, p_stock_ex_id, p_stock_symbol.
-The stock_id, stock_ex_id and stock_symbol must be recorded in the stock_listing table.
-The starting price from company must be copied to the stock price table for the stock exchange used. The current system date/time will be used for the time_start and the time_end will be null. The procedure must convert from company currency to the stock exchange currency.

Stock Split

17. Write a PL/SQL procedure called SPLIT_STOCK.
 - input parameters: p_stock id, p_split_factor
 - The split_factor must be *greater than 1* and can be fractional. The number of shares will be calculated using the split_factor. If the split factor is not valid, call RAISE_APPLICATION_ERROR to stop all processing.
 - The total shares outstanding cannot exceed the authorized amount. Your procedure must raise an application error if the split would cause the shares outstanding to exceed the shares authorized.
 - Every shareholder must receive, as a buyer, an additional trade equal to the additional shares to which they are entitled. For example, if the split_factor is 2 then each shareholder will be entitled to an additional trade that is equal to the number of shares that they owned before the split. Use the Current_Shareholder_Shares view to determine the number of shares owned. These trades will not take place at a stock exchange. The price total will be null, and there will be no brokers involved.

Reverse Split

18. Write a PL/SQL procedure called REVERSE_SPLIT.
 - input parameters: p_stock id, p_merge_factor
 - The merge_factor *must be greater than 0 and less than 1*. The number of shares will be calculated using the merge_factor. Call RAISE_APPLICATION_ERROR if the merge factor is invalid.
 - Every shareholder must sell some of the stock it currently owns. Use the Current_Shareholder_Shares view to determine the number of shares owned. If the merge_factor is .75 then adjustments must be made to indicate that one fourth of each shareholder's stock has been removed. The database can handle fractions of a share. These trades will not take place at a stock exchange. The price total will be null, and there will be no brokers involved.

Additional Queries

19. Display the trade id, the stock id and the total price in US dollars for the secondary market trade with the highest total price. Convert all prices to US dollars from the stock exchange currency.
20. Display the name of the company and trade volume for the company whose stock has the largest volume worldwide. The volume of a stock is the amount of shares of that stock that are traded on the secondary markets.
21. For each stock exchange, display the symbol of the stock with the highest trade volume. Show the stock exchange name, stock symbol and trade volume. Sort the output by the name of the stock exchange.
22. List all companies on the New York Stock Exchange. Display the company name, trade volume, the current price, and the percentage change from the previous price. Sort the output in descending order of trade volume. The sample data in the database contains information for only 3 companies but your query must attempt to list the top 5 companies.

2. Coding Standards

- Indent subordinate clauses as shown in Lesson 4a *Basic DML*
- Use a column alias with all functions and math expressions
- Multiply all percentages by 100 for readability
- Round math expressions two places past the decimal as needed

3. Project Submission

Your statements are to be contained in a single .sql file saved as *yourname.sql*. Each query for questions 1 to 12 and 19 to 22 is to be written as **one** statement. For items 13 to 18, include any sequence creation statements, and test exec statements.

Your code will be graded based on the output it creates and its conformance to the coding standards. Any code that does not execute will be awarded 0 points. Any code that cannot be explained or defended will also receive 0 points.

Although you may discuss concepts with fellow classmates, answers and code cannot be shared. Under no circumstances are you allowed to work at another's workstation while they are logged in.