**Part I**

**Short answer questions. Answer the following questions with several sentences. Provide references if you are quote from published articles or books. You can also relate to your personal professional experiences while answering the questions. For detailed instructions, please refer to Page 1 of this document.**

1. Discuss data governance and what needs to be included in a data governance program.

**Data governance is a set of processes meant for managing the data within an organization. There is an emphasis on availability, integrity & compliance within regulations. Data governance drives data quality, initiatives, data architecture, integration, data management, data warehousing, & business intelligence. It encompasses all data matters. From when data is collected, stored, used, & eventually discarded or archived.**

1. Discuss when to use the GROUP BY clause.

**Let’s say you want to see the average, sum, count, distinct count, maximum, minimum, standard deviation, some kind of aggregate in a query. Simply calling upon those functions alone will only grant that singular output based on the entire query. It will not show those aggregates divided out based on columns. SQL needs GROUP BY to make this distinction.. This creates sections & sub-sections for those aggregate functions to breakdown the results. For someone going from Excel to SQL, this would resemble something similar to a pivot table.**

1. Explain what the star schema is and why it is required for data warehouses.

**Star schemas are used to organize data more clearly. It also makes it easier for querying, analysis & reporting. When data comes in through the ETL process, it’s not very pretty. The data is organized into FACT tables & DIMENSION tables. These comprise Star Schemas with relationships between tables established. FACT tables display information like transactions & orders. A lot of dynamic data with much traffic. DIMENSION tables that reveal attributes to data. This helps with querying data & lowering the amount of storage required to house data. A customer or employee most likely will be given an ID that will be in the FACT tables. Their corresponding DIMENSION tables like EMPLOYEE or CUSTOMER will have the attributes such as Name , Address etc. This information will not need to be repeated in the FACT tables as they’re mostly static. They usually don’t go through many changes. DATE tables are also a type of DIMENSION table that help fill out information during analysis such as weekdays, months & fiscal quarters that a FACT table DATE column could be expanded upon.**

1. Explain the three classes of SQL commands and when they would be used.

**There are Data Definition Languages (DDL), Data Manipulation Languages (DML), & Data Control Languages (DCL). DDL is used for CREATE (making tables, columns, constraints), ALTER (modifying column datatypes, adding or dropping columns) DROP, (deletes table) & TRUNCATE (emptying tables without deleting tables). DML is used for INSERT ( adding rows of data), UPDATE (modifying a row) , DELETE (removing rows) & querying data (SELECT, FROM, WHERE ORDER BY). Data Control Language is for administering various privileges amongst users such as GRANT, ADD & REVOKE. DCL is also used for saving (committing) data.**

1. What are integrity controls? Describe the different types.

**Integrity controls protect data from unauthorized use & update. There are domains, which include user-defined data types. For example, it can place a rule on a column that numerical values have to be in a certain range. Assertions are controls that place database conditions. If a certain condition is not meant, the data cannot be added or modified. For example, an Assertion would be that an employee’s spouse cannot be their supervisor. Triggers are more complex than assertions. They can be used for security purposes. Examples include prohibiting salaries outside of business hours, if customer invoice received after due date triggers a penalty, creates audit log rows when sensitive information is viewed/queried.**

1. How is the HAVING clause different from the WHERE clause?

**WHERE clauses are used for standard queries. Usually, to address a condition or filter on a column. HAVING clause is used for aggregates (SUM, AVG, COUNT) with GROUP BY. The HAVING clause applies to conditions on the aggregate. WHERE should be used for WHERE EMPLOYEE\_ID = 1. (Placed after FROM statement) HAVING would be used when HAVING SUM(COSTS) > 100. (Placed after GROUP BY)**

1. Explain the difference between operational and informational systems as well as the primary factors that contribute to the need for separation.

**Operational systems (system of record) are designed to process incoming data. They’re used to run business in real time on current data.**

**Informational Systems involve collection, compilation and derivation of attributes from data. This is used for reporting, mining, queries, & predictive models.**

1. What is a self-join and how is it used?

**Self-Joins are used when a table is joined to itself. It can be based on multiple columns in the same table. It can be based on a sub query of the same table that joins to the original table. It can be a duplicate table is used such as selecting FROM EXAMPLE A, EXAMPLE B WHERE conditions exist on one table but not in another. (Used for comparison or testing).**

**An example I can use, I had to track how much time each truck took on an order. I had to run sub queries grabbing the minimum date from the first trip on the order, another sub query grabbing the max date on the last order, then running a join on the two subqueries, finally creating the DATEDIF between these two dates. This all happened in the same table. That’s a circumstance to use a self-join.**

1. What threats to data security must be addressed in a comprehensive data security plan?

**Threats to data security include accidental losses (human error, software & hardware breaches), theft, fraud, loss of privacy, confidentiality, data integrity & availability. There can be many scamming/phishing attempts to gain access to company or personal data without anyone knowing there is a breach. Cyber criminals can use this information to reveal business secrets, steal financial information, retrieve personal info for identity theft, or levy for large ransoms. People can also breach data by sneaking into a building & downloading data through a thumb drive or a planted wireless device that transmits that data to somewhere outside. Scammers can also use this type of technology in hotels, coffee shops, places with public wifi to steal private data. That’s why it’s important to use a secure VPN when logging into work networks outside the company’s site internet.**

1. Discuss the ETL process.

**ETL means Extract, Transform, & Load. Extraction is when the data comes in. It’s very raw, unorganized without much structure. The next stage, Transformation solves these problems. This makes the data ready for transit to the final stage, Loading. The data is placed in the appropriate servers & databases.**

**Part II**

**For each question in this part of the exam, you are required to submit both your SQL code and a screen shot of the results returned. Missing any one part will result in a 0 for the question. If I take your SQL code and run it in my APEX account, I should receive the exam same result as the screen shot you provided. Please note, your SQL code should not be a screen shot (I should be able to copy and paste your code directly from this document); only take screen shot of the results so I can view your result clearly. For detailed instructions on this part of the exam, please refer to Page 1 of this document. In this part, each question is 5 points.**

**Questions**

1. **Find the item classes and the number of parts under each class. Only show those classes with at least 3 types of parts.**

SELECT ITEM\_CLASS, SUM(UNITS\_ON\_HAND) AS Number\_of\_Parts

FROM PART

WHERE ITEM\_CLASS IN

(SELECT ITEM\_CLASS

FROM PART

GROUP BY ITEM\_CLASS

HAVING COUNT(PART\_DESCRIPTION) >= 3)

GROUP BY ITEM\_CLASS

Graphical user interface, application

Description automatically generated

1. **List the customer number, last name, first name, and complete address of every customer who lives on Pine; that is, whose address contains the letters "Pine".**

SELECT C\_NUMBER, CLAST, CFIRST, STREET||', '||CITY||', '||STATE||' '||ZIP\_CODE AS FULL\_ADDRESS

FROM CUSTOMER

WHERE STREET LIKE '%Pine%'

Graphical user interface, text, application

Description automatically generated

1. **List the customer number, last name, and first name of every customer who has the same credit limit as the largest credit limit awarded to any customer of sales rep 6.**

SELECT C\_NUMBER, CLAST, CFIRST

FROM CUSTOMER

WHERE CREDIT\_LIMIT = (SELECT MAX(CREDIT\_LIMIT) AS MAX\_CL

FROM CUSTOMER

WHERE SLSREP\_NUMBER = 6)

Graphical user interface, application

Description automatically generated

1. **List the warehouse number and total dollar amount of all items sold from that warehouse. Rename the calculated column TOTAL\_VALUE. TOTAL\_VALUE=SUM(NUMBER\_ORDERED\*QUOTED\_PRICE).**

SELECT P.WAREHOUSE\_NUMBER, SUM(OL.NUMBER\_ORDERED\*OL.QUOTED\_PRICE) AS CALCULATED\_TOTAL

FROM ORDER\_LINE OL

LEFT JOIN PART P ON OL.PART\_NUMBER = P.PART\_NUMBER

GROUP BY P.WAREHOUSE\_NUMBER

Graphical user interface, application

Description automatically generated

1. **List the order number and order date for every order that contains a part stored in warehouse number 3.**

SELECT O.ORDER\_NUMBER, O.ORDER\_DATE

-- P.PART\_NUMBER, P.WAREHOUSE\_NUMBER

FROM ORDERS O

LEFT JOIN ORDER\_LINE OL ON O.ORDER\_NUMBER = OL.ORDER\_NUMBER

LEFT JOIN PART P ON OL.PART\_NUMBER = P.PART\_NUMBER

WHERE P.WAREHOUSE\_NUMBER = 3

Graphical user interface, application

Description automatically generated

1. **Find the customer number, last name, first name, current balance, and sales rep number of every customer whose balance is larger than the balance of at least one customer of sales rep number 12.**

SELECT C\_NUMBER, CLAST, CFIRST, BALANCE, SLSREP\_NUMBER

FROM CUSTOMER

WHERE BALANCE > (SELECT MIN(BALANCE) AS MIN\_BALANCE

FROM CUSTOMER

WHERE SLSREP\_NUMBER = 12)

Graphical user interface, text, application

Description automatically generated

1. **List the customer number, last name, first name, and balance of each customer whose balance is greater than the average balance of all customers.**

SELECT C\_NUMBER, CLAST, CFIRST, BALANCE

FROM CUSTOMER

WHERE BALANCE > (SELECT AVG(BALANCE) AS AVG\_BALANCE

FROM CUSTOMER)

Graphical user interface, text, application

Description automatically generated

1. **Show the customer number for every customer who does not have a sales rep assigned yet.**

SELECT C\_NUMBER

FROM CUSTOMER

WHERE SLSREP\_NUMBER IS NULL

Graphical user interface, text, application

Description automatically generated

1. **List the warehouse number and the number of different parts stored in each warehouse, only include those warehouse with more than 2 different parts.**

SELECT DISTINCT WAREHOUSE\_NUMBER, COUNT(PART\_NUMBER) AS PART\_COUNT

FROM PART

GROUP BY WAREHOUSE\_NUMBER

HAVING COUNT(PART\_NUMBER) > 2

Graphical user interface, application

Description automatically generated

1. **List the full name and mailing address of customers who placed an order on 04-AUG-2013. If same** **customer has placed more than one order on this day, only show his/her information once.**

SELECT DISTINCT C.CFIRST||' '||C.CLAST AS FULL\_NAME, C.STREET||', '||C.CITY||', '||C.STATE||' '||C.ZIP\_CODE AS FULL\_ADDRESS, O.ORDER\_DATE

FROM CUSTOMER C

RIGHT JOIN ORDERS O ON C.C\_NUMBER = O.C\_NUMBER

WHERE ORDER\_DATE = '04-AUG-2013'

ORDER BY FULL\_NAME

Graphical user interface, application

Description automatically generated