Review Answers

1. Write the command to create a database called **products**.

    CREATE DATABASE products
    ON (
      NAME = products_data,
      FILENAME = 'C:\Program Files\Microsoft SQL Server\MSSQL\Data\
    products_data.mdf'
)
    LOG ON (
       NAME = products_log,
       FILENAME = 'C:\Program Files\Microsoft SQL Server\MSSQL\Data\
    products_log.ldf'
)

2. What is a user-defined data type? Why is it used?

    A user-defined data type is created by the user for custom data storage. It can enforce data integrity.

    It is used when several tables must store the same type of data in a column and you must ensure that these columns have exactly the same data type, length, and null ability.

3. When creating or altering a table, what are constraints? List and explain two constraints.

    Used to enforce data integrity on a table and individual columns in a table, and limits the possible values a user can enter into a table or a column

    NOT NULL   - specify the column that does not accept NULL values
    CHECK   - specifies what data can be entered in a column
    UNIQUE   - enforces uniqueness of the values in a set of columns
    PRIMARY KEY- identifies the column(s) which values uniquely identity a
                                row in the table
    FOREIGN KEY- identifies the relationships between tables
    DEFAULT   - defines a value the system will automatically insert into a
                                column when a value is not given

4. What is the objective of SQL?

    The objective of SQL is to allow users to create the database and relation structures, perform data management tasks, and perform simple and complex queries to transform inputs into required outputs.

5.   What is a database.

A database contains the objects used to represent, manage, and access data. It is a collection of tables with data, and other objects, such as views, indexes, stored procedures, and triggers.

6.   Within the context of a database, what is a table?

A table is used to store and manipulate data. Each row in the table representing a unique record, and each column a field within the record.

7.   What are the 4 items you must define when you create a table?

The 4 items are: column name, data type, required data, and default value.

8.   Explain the data type timestamp.

Timestamp is a database wide automatically generated unique binary number which has precision down to the nanosecond.

9.   Explain the two major components of SQL.

DDL used for defining database structure and controlling access to the data
DML used for retrieving and updating data

10.   Create a table called **BOOKINFO** with the following attributes:

Column Name	Data Type	Length	NULL
TITLE_ID	INT		No
TITLE	VARCHAR	80	No
BOOK_TYPE	CHAR	12	No
PUBLISHER_ID	CHAR	4	Yes
PRICE	MONEY		Yes
PUBLISH_DATE	DATETIME		No

```
CREATE TABLE BOOKINFO
(
  TITLE_ID          INT NOT NULL,
  TITLE             VARCHAR(80) NOT NULL,
  BOOK_TYPE         CHAR(12) NOT NULL,
  PUBLISHER_ID      CHAR(4),
  PRICE             MONEY,
  PUBLISH_DATE      DATETIME NOT NULL
)
```

11. Write the command to add a default value of 'unclassified' to the BOOK_TYPE column in the BOOKINFO table created in question 10.

    ALTER TABLE BOOKINFO
    ADD DEFAULT ('unclassified') FOR BOOK_TYPE

12. Write the command to add a default value of **today's date** to the PUBLISHER_DATE column in the BOOKINFO table created in question 10.

    ALTER TABLE BOOKINFO
    ADD DEFAULT (GETDATE()) FOR PUBLISHER_DATE

13. Write the command to add the column NOTES to the BOOKINFO table created in question 10 with a data type of VARCHAR, length of 200, and allow null values.

    ALTER TABLE BOOKINFO
    ADD NOTES VARCHAR(200)

14. Write the command to add the constraint to the TITLE_ID column only allowing values of a number from 0 to 10000.

    ALTER TABLE BOOKINFO
    ADD CHECK (TITLE_ID >= 0 AND TITLE_ID <= 10000)

15. Write the command to drop the table BOOKINFO. Write the command to drop the database 'products'.

    DROP TABLE BOOKINFO
    DROP DATABASE products

Review Answers

1. List the five basic search conditions to restrict retrieved rows with the WHERE clause, and explain how they are used.

   The five basic search conditions and how they are used are:

   1. Comparison – compares the values of one expression to the value of another expression
   2. Range – tests whether the value of an expression falls within a specified range of values
   3. Set membership – tests whether the value of an expression equals one of a set of values
   4. Pattern match – tests whether a string matches a specified pattern
   5. Null – test whether a column has a NULL value

2. Explain the DATEADD and DATEDIFF commands.

   The DATEADD command returns a new datetime value based on adding an interval to the specified date. The result is a datetime value equal to the date plus the number of date parts. The DATEDIFF command returns the number of date and time boundaries crossed between two specified dates.

3. Explain the aggregate operators COUNT, SUM, and AVG.

   The aggregate operator COUNT returns the number of rows which contain non-null values, SUM returns the sum of the values in a specified column, and AVG returns the average of the values in a specified column.

   EMP_1 Table

EMP_NUM	EMP_LNAME	EMP_FNAME	EMP_INITIAL	EMP_HIREDATE	JOB_CODE
101	News	John	G	08-Nov-00	502
102	Senior	David	H	12-Jul-89	501
103	Arbough	June	E	01-Dec-96	500
104	Ramoras	Anne	K	15-Nov-87	501
105	Johnson	Alice	K	01-Feb-93	502
106	Smithfield	William		22-Jun-04	500
107	Alonzo	Maria	D	10-Oct-93	500
108	Washington	Ralph	B	22-Aug-91	501
109	Smith	Larry	W	18-Jul-97	501

   Use the EMP_1 table shown above, to answer questions 4 through 7.

4. Write the SQL code that will list all the attributes for a job code of 502.

   ```
   SELECT    *
   FROM      EMP_1
   WHERE     JOB_CODE = '502'
   ```

5. Write the SQL code required to list all employees whose last names start with Smith.

   ```
   SELECT *
   FROM      EMP_1
   WHERE     EMP_LNAME LIKE 'Smith%'
   ```

6. Write the SQL code that will produce a listing for the data in descending order by the last name.

   SELECT   *
   FROM   EMP_1
   ORDER BY   EMP_LNAME DESC

7. Write the SQL code that will list only the different job codes found in the table.

   SELECT   DISTINCT JOB_CODE
   FROM   EMP_1

Table name: CUSTOMER

	CUS_CODE	CUS_LNAME	CUS_FNAME	CUS_INITIAL	CUS_AREACODE	CUS_PHONE	CUS_BALANCE
+	10010	Ramas	Alfred	A	615	844-2573	0.00
+	10011	Dunne	Leona	K	713	894-1238	0.00
+	10012	Smith	Kathy	W	615	894-2285	896.54
+	10013	Olowski	Paul	F	615	894-2180	1285.19
+	10014	Orlando	Myron		615	222-1672	673.21
+	10015	O'Brian	Amy	B	713	442-3381	1014.56
+	10016	Brown	James	G	615	297-1228	0.00
+	10017	Williams	George		615	290-2556	0.00
+	10018	Farriss	Anne	G	713	382-7185	0.00
+	10019	Smith	Olette	K	615	297-3809	453.98

Use the CUSTOMER table shown above, to answer questions 8 through 10.

8. Create the SQL query that will produce a list of customers who have an unpaid balance, listing the balances in descending order by customer balances. The query should produce the result set listing below.

CUS_LNAME	CUS_FNAME	CUS_INITIAL	CUS_BALANCE
Olowski	Paul	F	1285.19
O'Brian	Amy	B	1014.56
Smith	Kathy	W	896.54
Orlando	Myron		673.21
Smith	Olette	K	453.98

   SELECT   CUS_LNAME, CUS_FNAME, CUS_INITIAL, CUS_BALANCE
   FROM   CUSTOMER
   WHERE   CUS_BALANCE > 0
   ORDER BY   CUS_BALANCE DESC

9. Create the SQL query that will find the average unpaid customer balance, the minimum balance, the maximum balance, and the total of the unpaid balances. The query should produce the result set listing below.

Average Balance	Minimum Balance	Maximum Balance	Total Unpaid Bills
432.35	0.00	1285.19	4323.48

```
SELECT      AVG( CUS_BALANCE ) AS AverageBalance,
            MIN( CUS_BALANCE ) AS MinimumBalance,
            MAX( CUS_BALANCE ) AS MaximumBalance,
            SUM( CUS_BALANCE ) AS TotalUnpaidBills
FROM        CUSTOMER
```

10. Write the query to count the number of customers with a customer balance over $500.

```
SELECT   COUNT(*)
FROM     CUSTOMER
WHERE    CUS_BALANCE > 500
```

Table name: CHARTER

		CHAR_TRIP	CHAR_DATE	AC_NUMBER	CHAR_DESTINATION	CHAR_DISTANCE	CHAR_HOURS_FLOWN	CHAR_HOURS_WAIT	CHAR_FUEL_GALLONS
▶	+	10001	05-Feb-06	2289L	ATL	936.0	5.1	2.2	354.1
	+	10002	05-Feb-06	2778V	BNA	320.0	1.6	0	72.6
	+	10003	05-Feb-06	4278Y	GNV	1,574.0	7.8	0	339.8
	+	10004	06-Feb-06	1484P	STL	472.0	2.9	4.9	97.2
	+	10005	06-Feb-06	2289L	ATL	1,023.0	5.7	3.5	397.7
	+	10006	06-Feb-06	4278Y	STL	472.0	2.6	5.2	117.1
	+	10007	06-Feb-06	2778V	GNV	1,574.0	7.9	0	348.4
	+	10008	07-Feb-06	1484P	TYS	644.0	4.1	0	140.6
	+	10009	07-Feb-06	2289L	GNV	1,574.0	6.6	23.4	459.9
	+	10010	07-Feb-06	4278Y	ATL	998.0	6.2	3.2	279.7
	+	10011	07-Feb-06	1484P	BNA	352.0	1.9	5.3	66.4
	+	10012	08-Feb-06	2778V	MOB	884.0	4.8	4.2	215.1
	+	10013	08-Feb-06	4278Y	TYS	644.0	3.9	4.5	174.3
	+	10014	09-Feb-06	4278Y	ATL	936.0	6.1	2.1	302.6
	+	10015	09-Feb-06	2289L	GNV	1,645.0	6.7	0	459.5
	+	10016	09-Feb-06	2778V	MQY	312.0	1.5	0	67.2
	+	10017	10-Feb-06	1484P	STL	508.0	3.1	0	105.5
	+	10018	10-Feb-06	4278Y	TYS	644.0	3.8	4.5	167.4

11. Using the CHARTER table shown above, write the SQL query that will list the date, aircraft number, destination, distance, and hours flown for aircraft number 2778V. The query should produce the result set listing below.

CHAR_DATE	AC_NUMBER	CHAR_DESTINATION	CHAR_DISTANCE	CHAR_HOURS_FLOWN
05-Feb-06	2778V	BNA	320	1.6
06-Feb-06	2778V	GNV	1574	7.9
08-Feb-06	2778V	MOB	884	4.8
09-Feb-06	2778V	MQY	312	1.5

```
SELECT   DISTINCT  CHAR_DATE,
         AC_NUMBER,
         CHAR_DESTINATION,
         CHAR_DISTANCE,
         CHAR_HOURS_FLOWN
FROM     CHARTER
WHERE  AC_NUMBER = '2778V'
```

12. List the two mandatory clauses in a SELECT command.

SELECT and FROM

13. Explain the logical operator AND, OR, and NOT.

The logical operator AND combines two search conditions that must both be true, OR combines two search conditions when one or the other must be true, and NOT selects rows where a search condition is false.

14. List the four data manipulation commands, and explain their purpose.

    The four data manipulation commands and their purpose are as follows:

    1. SELECT command is used to retrieve and display data from one or more database tables.
    2. INSERT command is used to add new rows of data in a table.
    3. UPDATE command is used to modify existing data in a table.
    4. DELETE command is used to remove rows of data from a table.

15. In SQL Server, what are literals?

    Literal are constants used in SQL statements. Non-numeric data vales are enclosed in single quotes, and numeric values are not enclosed in single quotes.

Review Answers

1. What is the difference between UNION and UNION ALL?

    UNION operator will yield unique rows eliminating duplicates rows. UNION ALL
    operator will yield all rows of both relations, including duplicates.

2. Suppose that you have two tables, EMPLOYEE and EMPLOYEE_1. The
    EMPLOYEE table contains the records for three employees: Alice Cordoza, John
    Cretchakov, and Anne McDonald. The EMPLOYEE_1 table contains the records for
    two employees: John Cretchakov and Mary Chen. Given that information, what is
    the query output for the UNION query? List the query output.

    The query output will be:

    Alice Cordoza
    John Cretchakov
    Anne McDonald
    Mary Chen

3. Given the employee information in question 2, what is the query output for the
    UNION ALL query? List the query output.

    The query output will be:

    Alice Cordoza
    John Cretchakov
    Anne McDonald
    John Cretchakov
    Mary Chen

4. What are the three join types included in the OUTER JOIN classification? Describe
    each of the types.

    An OUTER JOIN is a type of JOIN operation that yields all rows with matching
    values in the join columns as well as all unmatched rows. The three join types are:

        LEFT OUTER JOIN
        RIGHT OUTER JOIN
        FULL OUTER JOIN

    The LEFT OUTER JOIN will yield all rows with matching values in the join columns,
    plus all of the unmatched rows from the left table. The RIGHT OUTER JOIN will
    yield all rows with matching values in the join columns, plus all of the unmatched
    rows from the right table. The FULL OUTER JOIN will yield all rows with matching
    values in the join columns, plus all the unmatched rows from both tables named in
    the FROM clause.

Contents of the EMPLOYEE Table

EMP_NUM	EMP_LNAME	EMP_FNAME	EMP_INITIAL	EMP_HIREDATE	JOB_CODE
101	News	John	G	08-Nov-00	502
102	Senior	David	H	12-Jul-89	501
103	Arbough	June	E	01-Dec-96	500
104	Ramoras	Anne	K	15-Nov-87	501
105	Johnson	Alice	K	01-Feb-93	502
106	Smithfield	William		22-Jun-04	500
107	Alonzo	Maria	D	10-Oct-93	500
108	Washington	Ralph	B	22-Aug-91	501
109	Smith	Larry	W	18-Jul-97	501

5. Using the EMPLOYEE table shown above, write the SQL code to enter the first two rows for the table.

   INSERT INTO EMPLOYEE
   VALUES ('101', 'News', 'John', 'G', '08-Nov-00', '502')

   INSERT INTO EMPLOYEE
   VALUES ('102', 'Senior', 'David', 'H', '12-Jul-89', '501')

6. Using the EMPLOYEE table shown above, write the SQL code to change the job code to 501 for the person whose employee number is 107.

   UPDATE EMPLOYEE
   SET   JOB_CODE = '501'
   WHERE   EMP_NUM = '107'

7. Using the EMPLOYEE table shown above, write the SQL code to delete the row for the person named William Smithfield, who was hired on June 22, 2004 and whose job code classification is 500.

   DELETE   FROM EMPLOYEE
   WHERE   EMP_NUM = '107'
   AND   EMP_LNAME = 'Smithfield'
   AND   EMP_FNAME = 'William'
   AND   EMP_HIREDATE = '22-June-04'
   AND   JOB_CODE = '500'

8. Using the EMPLOYEE table shown above, write the SQL code to create a copy of EMPLOYEE table, naming the copy EMPLOYEE_2. Then write the SQL code that will add the attribute PROJ_NUM which is character with a length of 3 to its structure.

   SELECT *
   INTO EMPLOYEE_2
   FROM EMPLOYEE

   ALTER TABLE EMPLOYEE_2 ADD PROJ_NUM   CHAR(3)

Table name: CHARTER

		CHAR_TRIP	CHAR_DATE	AC_NUMBER	CHAR_DESTINATION	CHAR_DISTANCE	CHAR_HOURS_FLOWN	CHAR_HOURS_WAIT	CHAR_FUEL_GALLONS
►	+	10001	05-Feb-06	2289L	ATL	936.0	5.1	2.2	354.1
	+	10002	05-Feb-06	2778V	BNA	320.0	1.6	0	72.6
	+	10003	05-Feb-06	4278Y	GNV	1,574.0	7.8	0	339.8
	+	10004	06-Feb-06	1484P	STL	472.0	2.9	4.9	97.2
	+	10005	06-Feb-06	2289L	ATL	1,023.0	5.7	3.5	397.7
	+	10006	06-Feb-06	4278Y	STL	472.0	2.6	5.2	117.1
	+	10007	06-Feb-06	2778V	GNV	1,574.0	7.9	0	348.4
	+	10008	07-Feb-06	1484P	TYS	644.0	4.1	0	140.6
	+	10009	07-Feb-06	2289L	GNV	1,574.0	6.6	23.4	459.9
	+	10010	07-Feb-06	4278Y	ATL	998.0	6.2	3.2	279.7
	+	10011	07-Feb-06	1484P	BNA	352.0	1.9	5.3	66.4
	+	10012	08-Feb-06	2778V	MOB	884.0	4.8	4.2	215.1
	+	10013	08-Feb-06	4278Y	TYS	644.0	3.9	4.5	174.3
	+	10014	09-Feb-06	4278Y	ATL	936.0	6.1	2.1	302.6
	+	10015	09-Feb-06	2289L	GNV	1,645.0	6.7	0	459.5
	+	10016	09-Feb-06	2778V	MQY	312.0	1.5	0	67.2
	+	10017	10-Feb-06	1484P	STL	508.0	3.1	0	105.5
	+	10018	10-Feb-06	4278Y	TYS	644.0	3.8	4.5	167.4

Table name: CUSTOMER

		CUS_CODE	CUS_LNAME	CUS_FNAME	CUS_INITIAL	CUS_AREACODE	CUS_PHONE	CUS_BALANCE
►	+	10010	Ramas	Alfred	A	615	844-2573	0.00
	+	10011	Dunne	Leona	K	713	894-1238	0.00
	+	10012	Smith	Kathy	W	615	894-2285	896.54
	+	10013	Olowski	Paul	F	615	894-2180	1285.19
	+	10014	Orlando	Myron		615	222-1672	673.21
	+	10015	O'Brian	Amy	B	713	442-3381	1014.56
	+	10016	Brown	James	G	615	297-1228	0.00
	+	10017	Williams	George		615	290-2556	0.00
	+	10018	Farriss	Anne	G	713	382-7185	0.00
	+	10019	Smith	Olette	K	615	297-3809	453.98

Table name: AIRCRAFT

		AC_NUMBER	MOD_CODE	AC_TTAF	AC_TTEL	AC_TTER
►	+	1484P	PA23-250	1833.1	1833.1	101.8
	+	2289L	C-90A	4243.8	768.9	1123.4
	+	2778V	PA31-350	7992.9	1513.1	789.5
	+	4278Y	PA31-350	2147.3	622.1	243.2

Table name: MODEL

		MOD_CODE	MOD_MANUFACTURER	MOD_NAME	MOD_SEATS	MOD_CHG_MILE
►	+	C-90A	Beechcraft	KingAir	8	$2.67
	+	PA23-250	Piper	Aztec	6	$1.93
	+	PA31-350	Piper	Navajo Chieftain	10	$2.35

9. Using the tables shown above, produce the output shown below. The output, derived from the CHARTER and MODEL TABLES, is limited to February 6, 2004. (Hint: You will need to join through another table.)

	CHAR_DATE	CHAR_DESTINATION	AC_NUMBER	MOD_NAME	MOD_CHG_MILE
►	06-Feb-06	STL	1484P	Aztec	1.93
	06-Feb-06	ATL	2289L	KingAir	2.67
	06-Feb-06	STL	4278Y	Navajo Chieftain	2.35
	06-Feb-06	GNV	2778V	Navajo Chieftain	2.35

```
SELECT    CHARTER.CHAR_DATE,
          CHARTER.CHAR_DESTINATION,
          CHARTER.AC_NUMBER,
          MODEL.MOD_NAME,
          MODEL.MOD_CHG_MILE
FROM      MODEL
JOIN      AIRCRAFT  ON MODEL.MOD_CODE      = AIRCRAFT.MOD_CODE
JOIN      CHARTER   ON AIRCRAFT.AC_NUMBER = CHARTER.AC_NUMBER
WHERE     CHARTER.CHAR_DATE = 'Feb 6 2004'
```

10. Explain the GROUP BY clause.

The GROUP BY clause specifies a summary query. Instead of producing one rows of query results for each rows of data in the database, a summary query groups together similar rows and then produces one summary rows of query results for each group.

11. Explain the HAVING clause.

The HAVING clause tells SQL to include only certain groups produced by the GROUP BY clause in the query results. Like the WHERE clause, it uses a search condition to specify the desired groups.

12. Explain the ROLLUP clause.

The ROLLUP clause is used with row aggregates to produce subtotals of groups and final totals. These summary values appear as additional rows in the query results.

13. Explain the inner join type.

The inner join produces a result set that includes only the rows of the joining tables that meet the restriction using a comparison operator. Rows that do not meet the join restriction are not included in the joined table.

Review Answers

1. Describe a view.

    A view is a virtual table in the database whose contents are defined by a query. It does not exist as a table in the database. The result set is produced when the view is run.

2. What is a subquery, and what are its basic characteristics?

    A subquery is a query (expressed as a SELECT statement) that is located inside another query. The first SQL statement is known as the outer query, the second is known as the inner query or subquery. The inner query or subquery is normally executed first. The output of the inner query is used as the input for the outer query. A subquery is normally expressed inside parenthesis and can return zero, one, or more rows and each row can have one or more columns.

3. What is a correlated subquery?

    A correlated subquery is subquery that executes once for each row in the outer query. This process is similar to the typical nested loop in a programming language. Contrast this type of subquery to the typical subquery that will execute the innermost subquery first, and then the next outer query … until the last outer query is executed. That is, the typical subquery will execute in serial order, one after another, starting with the innermost subquery. In contrast, a correlated subquery will run the outer query first, and then it will run the inner subquery once for each row returned in the outer subquery.

4. What are some general guidelines for creating and using indexes?

    Create indexes for each single attribute used in a WHERE, HAVING, ORDER BY, or GROUP BY clause. If you create indexes in all single attributes used in search conditions, the DBMS will access the table using an index scan, instead of a full table scan.

    Do not use indexes in small tables. Do not create indexes for tables with few rows and few attributes - unless you must ensure the existence of unique values in a column.

    Declare primary and foreign keys so the optimizer can use the indexes in join operations. All natural joins and old-style joins will benefit if you declare primary keys and foreign keys because the optimizer will use the available indexes at join time.

    Declare indexes in join columns other than primary and foreign keys. If you do join operations on columns other than the primary and foreign key, you may be better off declaring indexes in such columns.

5. List the four types of indexes. Describe each of these indexes.

   Clustered
   – An index where the logical order of the indexed values is the same as the physical
     order in which the corresponding rows exist in the table
   – Can be only one clustered index per tables

   Nonclustered
   – An index where the logical order of the index does not match the physical order of
     the rows on the disk

   Composite index
   – Used when two or more columns are best searched as a unit because of their
     logical relationship.

   Unique
   – Used to prevent duplicate rows in a column or columns

6. Given the following view definition, does updating the database through this view
   create any problems? If so, what are they? If not, why not?

   CREATE VIEW ORDTOT
   (ORDER_NUMBER, ORDER_TOTAL)
   AS
   SELECT ORDER_NUMBER, SUM(NUMBER_ORDERED * QUOTED_PRICE)
   FROM ORDER_LINE
   GROUP BY ORDER_NUMBER

   You cannot update data in the ORDER_LINE table using the ORDTOT view because
   the view does not contain the primary key of the base table, and the view includes a
   computed column.

7. Describe the CASE expression?

   A CASE expression evaluates a list of conditions and returns one of multiple possible
   result expressions. CASE can be used in any statement or clause that allows a valid
   expression. The CASE expression has two formats:
   • Simple CASE expression compares an expression to a set of simple
     expressions to determine the result
   • Searched CASE expression evaluates a set of Boolean expressions to
     determine the result

8. Explain an advantage of using subqueries in your query, and an advantage of using
   joins in your query.

   Subqueries allow you to calculate aggregate values adhoc and to feed it back to the
   outer query for comparison. Joins allow you to display information form both tables.

9. A table contains a column defined as: SIN VARCHAR(9) NOT NULL   Which type of index would you create if you want the data to be unique and physically stored in SIN sequence?

   UNIQUE CLUSTERED index

10. Compare and contrast clustered and nonclustered indexes.

   Clustered index
        - Table rows are physically sorted into index order
        - Provides physical order of data storage
        - Table can have only one clustered index
        - Faster sequential access to data in index order

   Nonclustered index
        - Index order and physical order are not related
        - Provides ordered list with pointers to physical location of the data
        - Larger and slower than clustered indexes

Review Answers

1. What is a trigger, and what is its purpose?

   A trigger is a block of SQL code that is automatically invoked by the DBMS upon the occurrence of a data manipulation event (INSERT, UPDATE or DELETE). Triggers are always associated with a table and are invoked before or after a data row is inserted, updated, or deleted. Any table can have one or more triggers.

   Triggers provide a method of enforcing business rules. Triggers are also excellent for enforcing data constraints that cannot be directly enforced by the data model.

2. Explain local variables.

   Local variables are defined with the DECLARE statement. The names of local variables are preceded by the @ sign. They cannot be defined as text or image. Their values are assigned with the SET or SELECT statement. Local variables are often used in a batch or in a stored procedure.

3. Explain global variables.

   Global variables are defined and maintained by SQL Server. Global variables are predefined by SQL Server, and can be used without being declared. Their names are preceded by two @@ signs. Many global variables report on system activities from the last time SQL Server was started, or they report information about a connection.

4. What is a stored procedure?

   A stored procedure is a named block of SQL statements. One of the major advantages of stored procedures is that they can be used to encapsulate and represent business transactions. You can encapsulate SQL statements within a single stored procedure and execute them as a single transaction.

5. What are the advantages to using stored procedures?

   There are two advantages to the use of stored procedures:
   1. Stored procedures substantially reduce network traffic and increase performance. Because the stored procedure is stored at the server, there no individual SQL statements are transmitted over the network.
   2. Stored procedures help reduce code duplication through code isolation and code sharing (creating unique SQL modules that are called by application programs), thereby minimizing the chance of errors and the cost of application development and maintenance.

6. List the order in which triggers are executed.

    Triggers are executed in the following order:
    *   INSTEAD OF trigger
    *   Regular constraints
    *   AFTER trigger

7. Explain an AFTER trigger.

    An AFTER trigger:
    *   Most common type of triggers
    *   Fire after the data modification is carried out
    *   Use with INSERT, DELETE or UPDATE statements

8. Explain an INSTEAD OF trigger.

    An INSTEAD OF trigger:
    *   Execute something else other than the triggering action
    *   Use with INSERT, DELETE or UPDATE statements

9. Describe the two special virtual tables used with a trigger.

    The two special virtual tables are inserted and deleted. The inserted virtual table contains copies of the new or updated rows when an INSERT or UPDATE statement is executed. The deleted virtual table contains the deleted rows when a DELETE statement is executed, or the old values when and UPDATE statement is executed.

10. Given the following partial script of a stored procedure, write the EXEC statement which will return the new price into the @IncAmount variable using an item price of 28.98 and a percentage increase of 15.

    ```
    CREATE PROCEDURE CalcPriceInc
    (
            @ItemPrice   float,
            @PercInc     float,
            @IncAmount float OUTPUT
    ...

    EXEC CalcPriceInc 28.98, 15, @IncAmount OUTPUT
    ```

Review Answers

1. What is a data warehouse, and what are its main characteristics?

   A data warehouse is an integrated, subject-oriented, time-variant and non-volatile database that provides support for decision-making. Its main characteristics are:

   • Subject-oriented - a warehouse is organized around major subjects rather than major application areas
   • Integrated - since data is coming from several sources, it must be made consistent and integrated to present a unified view of the data to the users
   • Time-variant - data in the warehouse is only accurate and valid at some point in time or over some time period.
   • Non-volatile - data is not updated in real time but is refreshed from operational data on a regular basis

2. What is OLAP, and what are its main characteristics?

   OLAP stands for On-Line Analytical Processing and uses multidimensional data analysis techniques. OLAP yields an advanced data analysis environment that provides the framework for decision making, business modeling, and operations research activities. Its four main characteristics are multidimensional data analysis techniques, advanced database support, easy to use end user interfaces, and support for client/server architecture.

3. What are the most relevant differences between operational and decision support data?

   The most relevant differences between operational and decision support data are:
   • time span
   • granularity
   • dimensionality

4. What does RAID stand for? Why and how is it commonly used?

   RAID is the acronym for Redundant Array of Independent Disks. RAID is used to improve fault tolerance or reliability by mirroring and by error detection/correction using parity. It increases performance by data striping, the simultaneous writing and reading over several disks.

5. Define security and privacy.

   Security means protecting the data against accidental or intentional access by unauthorized users. Privacy deals with the right of individuals not to have personal information collected, stored, and disclosed, either willfully or indiscriminately.

6. Describe encryption.

    Encryption is the process of transforming data from clear text into unreadable form with an algorithm that uses an encryption key. The encrypted data can then be transmitted or stored. To transform the encrypted data back into readable form, an algorithm using a decryption key is used.

7. Define authorization and authentication.

    Authorization is the granting of a right or privilege that enables a user to have legitimate access to a system. Authentication is the mechanism that determines whether a user is who he or she claims to be.