

ISYS2421 Exam Revision Questions

Week 1

Why data / information is needed

- For operational and transactional functions of the organisation
- Help in decision making
- "Data management" is needed to (store, collect and retrieve data)

What is a database

- A collection of data stored in a standardised format designed to be shared by multiple users
- An organised collection of logical related data

What is a DBMS?

- A software that defines a database, stores the data, supports a query language, produces reports and creates data entry screens. (DML)

Why are Databases Important

- Integral component of all information systems
- Integral part of most websites
- Almost all databases use SQL

Basic DBMS components

- tables/relation
- records/tuples/rows
- fields/attributes/columns
- Data dictionary/repository
- DDL, DML & DCL

Data dictionary or repository

- Meta - data
- Data about data
- Helps us describe what the raw data is, which is stored in our database
- Describe the semantics of the data stored: (naming, heading/caption, validation, foundation, scale, etc...)

Computability of Data

- To calculate new devices of data and information from the raw data. computing on the data.
- No need to store computable data
- We also need to present information in readable forms and formats

Advantages of database

- Minimal data redundancy
- Sharing of data - Uniform security, privacy and integrity
- Data consistency - Enforcement of standards
- Data independence

- Integration of data - Ease of application

Examples of database systems

- Oracle
- MySql
- Access, SQL server (Microsoft)
- DB2, SQL/DS (IBM)
- ingres, informix (Unix)

People associated with DBMS

- Database administrator (DBA)
- Data administrator (DA)
- End User
- System analysts and application runner
- Operators and maintenance personnel

Risks and costs of DBMS (main drawbacks)

- Organisational conflicts
- Development project failure
- System failure
- Overhead costs
- Need for sophisticated personnel

Week 2

SQL -> (Structured Query Language)

- A complete interface language for a database system
- Data definition : (ie. creating & defining files , fields etc)
- Data manipulation : (ie. inserting updating & deleting records) \
- Data Extraction: (ie. getting data from the database (queries)) \

Week 3

Scalar functions

- Functions that are a singular value
- Usable where you would normally use a value
- Normally used within the column definitions or within where clauses etc

Statistical functions

- There are 5 basic statistical functions (count(*) MAX MIN AVG SUM)
- They are also known as "grouping" functions
- All statistical functions return one value only, no matter how many rows they operate on.
- When they are used , value only ,no matter how many rows they operate on.
- When they are used , values of individual rows cannot be displayed
- "Can be used with usual where clauses."

Count(*)

- Counts numbers of rows

Sum(...)

- Adds up values in a specified column for all selected rows.

AVG

- Averages

Max

- Highest value found for a specific coulomb of selected rows

Min

- Lowest value for a specific coulomb of selected rows

Grouping Data - Group by

- Also known as “break” reports
- A grouping field is selected to group the rows
- The rows are sorted by the grouping field
- Rows with the same value for the grouping field are treated as a “group”
- Usually a statistical function is also used and applied to each group(eg: SUM)

Group By - Extra fields

- When group by used , the only fields that can be displayed are the grouping field and statistical functions
- Values for individual rows cannot be displayed.
- If you want to display other fields , a trick is to include the extra field as a secondary grouping field in the group by clause

Grouping Data ‘HAVING’

- The having clause operates like a where clause , but it’s applied to the grouping value
- Where is applied to each row before the grouping operation is done
- HAVING is applied after the grouping is done
- HAVING is applied after the grouping is performed and operates on the calculated grouping value (before it is displayed).

Week 4

Joining table

- Any operation involving more than one table , such as Querying on two tables involves a join Operation
- SQK specifies links between tables in the query its. MS Access allows Pre Defined relationships
- To us fields from two tables in one query, specify both tables in the from clause, separated by a comma.

Unbounded join

- An unbounded join is the combination of every row in one table with every row in the other table, regardless of their values.
- Every join will be an unbounded join unless it is constrained in some way (eg: by specifying common fields)

Natural join

- A natural join is made by specifying two fields (one in each table) that contain common values (eg keys), using the join/on or where clause.
- Assuming an unbounded join occurs, only the combinations where the two fields are the same are selected.
- In reality, the query processor does not need to produce every unbounded combination , only those that match linking criteria

Linking Criteria

- We name each common field pair defined in the where or from/on clause as a linking criteria

- The minimum number of linking criteria required per query is one less than the number of tables otherwise an unbounded join will occur (eg. if joining 4 tables there should be at least 3 linking criteria involving all 4 tables).
- Linking fields often have the same name ,which would cause ambiguous field without specifying table name.

Linking criteria : where

- FROM clause (link criteria can be specified within the FROM clause when the tables are specified using Join and ON clauses ; Allows LEFT and RIGHT joins to be performed)
- WHERE clause (link criteria can be specified in the WHERE clause. Limitations here is that naturally joins can be specified. Criteria (link & selection) are combined using AND clauses in the WHERE clause.there is only one WHERE clause in select statement.)

Qualifying fields and using aliases

- Any field name common to both tables must be qualified by the table name (or an abbreviation) within the query.
- A shorthand alias can be used by declaring it just after the table name in the FROM clause

Join other selection criteria

- There is only one WHERE clause , even when there are linking criteria
- Other selectio/search criteria are added on using AND or OR
- There is a WHERE clause for selection criteria when using the JOIN ON join syntac
- Multiple search criteria are added on using AND or OR

Self join

- A self join of a table to itself. This is a UNARY relationship
- The table name appears twice in the FROM clause
- Acts as if there are two copies of the same table.
- Aliases are used to identify each tables fields

Outer join

- An outer join specifies all rows from one table should be displayed even if they have no match in the other table.
- To write a query that performs an outer join of tables A and B and returns all rows from A apply the outer join operator to all columns of B in the join condition.
- This is done on the FROM clause using JOIN ON clause (Left join , Right join)

Views

- Setting up phantom tables made up from other table(s)
- Useful for often used joins and calculations helps get around problem of having to provide table join criteria within the queries themselves.
- Also useful for presenting tables in different ways to different users
- Used as form of security to give users access to a table but only curtain columb in the table.
- The view is active to remove a view do the following ('Drop View...'")
- If data is changed in the base tables, also changed in the view
- However high processing cost

Combine select statement

- Column results of two select statements are combined into one result set (eg : union

(union all) Intersect and minus)

- Column of each select must match by type & order placed in SELECT clause.
- ORDER BY cannot be Specified on whole result set.
- Column headings are based on the first query