

Project 1

Chapter13:

1. A _____ index is based on 0 and 1 bits to represent a given condition.
 - a.logical
 - b.multidimensional
 - c.normal
 - d.bitmapped

2. A _____ is a dynamic table that not only contains the SQL query command to generate the rows, but also stores the actual rows.
 - a.SQL view
 - b.materialized view
 - c.star schema
 - d.data cube

3. A _____ is optimized for decision support and is generally represented by a data warehouse or a data mart.
 - a.data store
 - b.ETL tool
 - c.data visualization
 - d.data analysis tool

4. A _____ schema is a type of star schema in which dimension tables can have their own dimension tables.
 - a.snowflake
 - b.starflake
 - c.dimension
 - d.matrix

5. An multidimensional database management systems (MDBMS) uses proprietary techniques to store data in _____ n-dimensional arrays.
 - a.table-like
 - b.matrix-like
 - c.network-like
 - d.cube-like

6. Bill Inmon and Chuck Kelley created a set of 12 rules to define a(n) _____.
 - a.data warehouse
 - b.multidimensional cube
 - c.OLAP tool
 - d.star schema

7. Computed or derived facts, at run time, are sometimes called _____ to differentiate them from stored facts.
 - a.schemas
 - b.attributes
 - c.metrics
 - d.dimensions

8. Conceptually, MDBMS end users visualize the stored data as a three-dimensional cube known as a _____.
 - a.multi-cube
 - b.database cube
 - c.data cube
 - d.hyper cube

9. Data _____ implies that all business entities, data elements, data characteristics, and business metrics are described in the same way throughout the enterprise.

- a.visualization
- b.analytics
- c.mining
- d.integration

10. Decision support data tend to be non-normalized, _____, and pre-aggregated.

- a.unique
- b.duplicated
- c.optimized
- d.sorted

11. Fact and dimension tables are related by _____ keys.

- a.shared
- b.secondary
- c.foreign
- d.linked

12. From a data analyst's point of view, decision support data differ from operational data in three main areas: time span, granularity, and _____.

- a.usability
- b.dimensionality
- c.transaction processing
- d.sparsity

13. In a star schema, attributes are often used to search, filter, or classify _____.

- a.tables
- b.sales
- c.facts
- d.dimensions

14. In a typical star schema, each dimension record is related to thousands of _____ records.

- a.attribute
- b.fact
- c.key
- d.primary

15. In business intelligence framework, data are captured from a production system and placed in _____ on a near real-time basis.

- a.decision support system
- b.portal
- c.data warehouse
- d.dashboard

16. In star schema representation, a fact table is related to each dimension table in a _____ relationship.

- a.many-to-one (M:1)
- b.many-to-many (M:M)
- c.one-to many (1:M)
- d.one-to-one (1:1)

17. Operational data are commonly stored in many tables, and the stored data represent information about a given _____ only.

- a.transaction
- b.database
- c.table
- d.concept

18. The attribute hierarchy provides a top-down data organization that is used for two main purposes: _____ and drill-down/roll-up data analysis.

- a.decomposition
- b.de-normalization
- c.normalization
- d.aggregation

19. The basic star schema has four components: facts, _____, attributes, and attribute hierarchies.

- a.keys
- b.relationships
- c.cubes
- d.dimensions

20. The reliance on _____ as the design methodology for relational databases is seen as a stumbling block to its use in OLAP systems.

- a.normalization
- b.denormalization
- c.star schema
- d.multidimensional schema

21. _____ can serve as a test vehicle for companies exploring the potential benefits of data warehouses.

- a.Data networks
- b.Data marts
- c.Data cubes
- d.OLAPs

22. _____ extends SQL so that it can differentiate between access requirements for data warehouse data and operational data.

- a.ROLAP
- b.OLAP
- c.DBMS
- d.BI

23. _____ splits a table into subsets of rows or columns and places the subsets close to the client computer to improve data access time.

- a.Normalization
- b.Meta modeling
- c.Replication
- d.Partitioning

24. _____ tools focus on the strategic and tactical use of information.

- a.Business
- b.Relational database management
- c.Business intelligence
- d.Networking

Deliverables

MySQL Server

Business Rules:

As the lead DBA for a local municipality, you are contacted by the city council to design a database in order to track and document the city's court case data. Some report examples:

Which attorney is assigned to what case(s)?

How many unique clients have cases (be sure to add a client to more than one case)?

How many cases has each attorney been assigned, and names of their clients (return number and names)?

How many cases does each client have with the firm (return a name and number value)?

Which types of cases does/did each client have/had and their start and end dates?

Which attorney is associated to which client(s), and to which case(s)?

Names of three judges with the most number of years in practice, include number of years.

Also, include the following business rules:

- An attorney is retained by (or assigned to) one or more clients, for each case.
- A client has (or is assigned to) one or more attorneys for each case.
- An attorney has one or more cases.
- A client has one or more cases.
- Each court has one or more judges adjudicating.
- Each judge adjudicates upon exactly one court.
- Each judge may preside over more than one case.
- Each case that goes to court is presided over by exactly one judge.
- **A person can have more than one phone number.**

Notes:

- Attorney data must include social security number, name, address, office phone, home phone, e-mail, start/end dates, dob, hourly rate, years in practice, bar (may be more than one - multivalued), specialty (may be more than one - multivalued).
- Client data must include social security number, name, address, phone, e-mail, dob.
- Case data must include type, description, start/end dates.
- Court data must include name, address, phone, e-mail, url.
- Judge data must include **same** information as attorneys (except bar, specialty and hourly rate; instead, use salary).
- Must track judge historical data—tenure at each court (i.e., start/end dates), and salaries.
- Also, history will track which courts judges presided over, if they took a leave of absence, or retired.
- **All** tables must have notes.

NB: In some designs, the common attributes would be inherited from a "person" table.

Additional Notes:

- Social security numbers, should be unique, and must use **SHA2 hashing with salt**.
- Entities must be included in logical layers (colored appropriately).
- ERD **MUST** include relationships **and** cardinalities.

Deliverables

1. ERD (in .mwb format, .png files will ***not*** be accepted)
2. SQL Statement Questions

Note: must include at least 15 records in the person table, and at least 5 records in all other tables (though, do not have to be in ERD—can be imported from data files).

No Credit will be given if tables and data do not forward-engineer to the CCI server.

Note: per_id values (must match for SQL statements to work!):

person (super type): 1 - 15

- client (sub type): 1 - 5
- attorney (sub type): 6 - 10
- judge (sub type): 11 - 15

HELPER VIDEO: http://qcitr.com/vids/LIS3781_P1.mp4

Note comments in video:

1. *****Be sure***** to use the updated **person** table and **stored procedure** files uploaded as images to the P1 module in Canvas. Use the code in the updated files, in lieu of the similar code shown in the project videos: **per_ssn** should be salted and hashed, as per the image files. The files demonstrate how to auto-generate data for testing purposes. More importantly, they demonstrate how to obfuscate data using unique salt and hash values for each record.
2. In the videos, sometimes, I say "encrypted"--I meant "hashed" values. SHA functions are "hashing" functions--***not*** encrypting functions!
3. **Except for the person table (see images online), ***BE SURE*** TO USE THE DATA IN THE VIDEOS!!!** (Debugging will be ***much*** easier*!)

Note: the stored procedure could be easily modified to obscure real data (e.g., social security numbers, credit card numbers, drivers' license numbers, etc.). Lastly, ***always*** use a secure connection!

Note:

README.md file should include the following items:

1. Screenshot of ***your*** ERD;
3. Optional: SQL code for the required reports.
4. Bitbucket repo links: ***Your*** lis3781 Bitbucket repo link

Deliverables (see screenshots below):

1. Provide **Bitbucket** read-only access to **course** repo, using Markdown syntax, (**README.md** must also include screenshots per above.)
(**DO NOT create README in Bitbucket—ALWAYS** do it locally, then push it to Bitbucket.)
2. **FSU's Learning Management System:** include course **Bitbucket** repo link

SQL Statements for P1

Save as **lis3781_p1_solutions.sql**

NOTE: All dollar amounts must be formatted to two decimal places, including a dollar sign (\$). All phone numbers and zip codes must include proper hyphens (-).

- 1) Create a view that displays attorneys' *full* names, *full* addresses, ages, hourly rates, the bar names that they've passed, as well as their specialties, sort by attorneys' last names.
- 2) Create a stored procedure that displays how many judges were born in each month of the year, sorted by month.
- 3) Create a stored procedure that displays *all* case types and descriptions, as well as judges' *full* names, *full* addresses, phone numbers, years in practice, for cases that they presided over, with their start and end dates, sort by judges' last names.
- 4) Create a trigger that automatically adds a record to the judge history table for every record added to the judge table.
- 5) Create a trigger that automatically adds a record to the judge history table for every record modified in the judge table.
- 6) Create a one-time event that executes one hour following its creation, the event should add a judge record (one more than the required five records), have the event call a stored procedure that adds the record (name it **one_time_add_judge**).

Extra credit

Create a scheduled event that will run every two months, beginning in three weeks, and runs for the next four years, starting from the creation date. The event should not allow more than the first 100 judge histories to be stored, thereby removing all others (name it **remove_judge_history**).