Week 2 Security

Q1. PgSQL Roles

* User
* SuperUser
* CreateDB
* CreateRole
* Group Roles

Q2. Role Inheritance

* Can be specified with INHERIT or NOINHERIT
* Can be granted membership of a group role
* If the user doesn't have the inherit role they must be granted the role with ( SET ROLE command)

Q3. Handling dependencies

* Handles relationship between two attributes
* Database packages that can be brought in

Q4.Privileges and database objects

Privileges

* Database privileges
* System privileges
* Object privileges

Objects

* Table
* View
* Sequences
* Index
* Synonyms

Q5. Principle of least privilege

Means giving users only the least required privileges they require to do their function.

Importance of principle of least privilege

* Reduces cyber attacks
* Stops spread of malware
* Improves end user productivity
* It helps streamline compliance and auditing

Q6. Secure password hashing

* Bcrypt
* Scrypt

Week 3 Auditing

Q1. Why do we audit in databases

* Human error
* Security risks
* Data Cleaning
* Custom experience

Q2. What is logging and what is auditing?

Logging

It's the process in which the user can have a recovery database and help recover any lost data from a sub database section, it also helps with syncing a primary and secondary database. When creating a DB a log file is created to track any changes that occur in that specific database.

Auditing

It's the process in the database that tracks each resource or authority in the database. It helps give a roadmap of each section of the database.

Q3. Postgres auditing options

Out-of-box auditing

* Statement - level auditing
* pgAudit extension

Q4. Database triggers

Before, After, Instead

DDL Triggers

* Trigger statements such as CREATE, ALTER, DROP
* System defined stored procedures
* DDL triggers are used to observe and control server actions

DML Triggers

* DML triggers such as INSERT, UPDATE, DELETE
* Can be used after triggers using the FOR, AFTER clause, Only fires after sql server finishing performing an action

CLR Triggers

* Special type triggers based on common language runtime
* Supported languages such as C#, Visual basic and F#

Drawbacks of auditing

* Costly
* Rely on experts
* Impossible to check everything
* Unsuitable for small companies
* Chances of fraud

Week 4 Indexing

Q1. How indexing works

* Sorting numbers on multiple fields
* Helps order an unordered table to improve reads

Q2. Hash indexing vs B-Tree Index

Hashing indexing

- Everything is stored in sections

- Keeps all unique values together

- Easy to find any unique value / speeds up searching

- Best way to search up specific values for specific attributes

B-Tree indexing

- Creates multi- level structure

- Divides data in half e.g. over 25 or smaller than 50 etc

- Helps avoid large sorting operations

Q3. Drawbacks of indexing

* Affects the performance
* Disk space
* Decreases performance of insert, updates, deletes

Q4. Finding slow queries

* Generate an actual execution plan
* Monitor resource usage
* Database engine tuning advisor
* Find slow sql DMVs – used to manage postgres
* Slow query logging#
* Explain Analyze

Q5. Explain Analyse

* Query to estimate how long each step would take
* Explain gives the estimate
* A way to see if indexing is needed

Q6. pgstat\_statments

* Views all sql statements executed in database
* Needs to be enabled by adding pg\_statstatements

Week 5 Partitioning and Tablespaces

Q1. Partitioning vs indexing

Partitioning

* Divide tables into sub tables
* Improve performance
* Makes it easier to manage
* Can be transferred into cheaper, slower disks or faster etc

Indexing

* Sorting numbers on multiple fields
* Helps order an unordered table

Q2. List vs Range partitioning

List

* Used for tables with range values
* Helps list unrelated data into partitions

Range

* Type of relational database with predefined specific values such as
* Ids
* Dates
* Currency

Q3. Benefits of partitioning

* Improves query performance
* Simplifies common admin tasks
* Allows accessing a large part of the database in a single partition.

Q4.Problems with partitioning

* Harder to maintain DB
* Requires additional equipment and expertise
* More complicated to understand for normal users.

Q5.Partitioning with triggers

* Can use trigger to insert in another table
* Can modify values inserted into table

Q6. Partitioning with inheritance

* Creates link between tables
* More flexible but had constraint limits
* Not very suitable with partitioning

Maybe? ^^^

Q7. Tablespace

* Helps control disk layout in postgres
* If postgres partition runs out another partition in another disk can be made for a temp solution
* Tablespace helps move most used data or frequently used data as the fastest disk if multiple are used

Week 6 Backup and Recovery

Q1.Planning for recovery

SLA - Service level agreements?

* Negotiated between customer and provider
* Establish the needs, priorities etc
* Give out the cost of overall work before in case of increase of demand.

DBA's Roles

To increase the MTBF - Mean time between failure

* Some systems require 100% availability

To Decrease MTTR - Mean time to recover

* Refers to down time following a failure
* Can be costly

To Reduce data loss

* Data should be stored in external devices
* Network performance is key
* Have Redundant hardware

5 categories of failure

Statement failure

* Individuals SQL statement fails
* Roll back of statements
* All statements have not been committed

Network failure

* Network crash
* Database timeout
* High volume cause errors

User Errors

* User does something wrong
* DML code was not committed

Media Error

* Hardware failure
* Disk damage
* System/ database deleted files
* Data lost from configuration choice

Insurance Failure

* Disorderly crash or shutdown
* Power Cut
* Rebooting server
* Hardware problems

Durability in PgSQL

Write- Ahead logs to prevent data loss

* Relational databases provide ACID guarantee
* Durability dbs guarantee that if a command is committed it will not be rolled back
* Any changes that are made and put on a Write-ahead-log ( WAL)

The CHECKPOINT command

* Checkout forces postgres to write all transactions to disk
* IO intensive will degrade db performance when running

Backup in PgSQL

SQL Dump

* Contains records of table structure
* Often used to recover database content

Filesystem Backup

* All required databases save to root
* It follows an all or nothing system e.g. look up all files or no files
* Database server must be shutdown to perform backup

WAL Log Archiving

* When segment is filled we can archive it

Week 7: Isolation levels

Importance of isolation

Database isolation refers to the ability of a database to allow a transaction to execute as if there are no other concurrently running transactions

Combined databases

* Could cause problems due to combined effort
* Conflicting code
* Miss matched statements

ACID

* **A**tomicity
* **C**onsistency
* **I**solation
* **D**urability

Locks and Mutexes

* Access must be (1)checked in, (2) checked out and lastly (3) checked back in
* Requesters form queue for resources to become available
* Some databases have a lock in place meaning specific open data cant be accessed unless released.
* Trade off is performance of server
* If locks are done incorrectly can cause system to stop working

SQL lock types

* Access Share
  + Conflicts only with read only query
* Share Row Exclusives
  + Prevents two updates happening at the same time on the table
* Access Exclusives
  + No other process is allowed to access the table

Deadlock

* When two processes need the same transaction and cause it to halt the system or both processes would fail.

Isolation Levels

* Locks can be explicitly removed to protect session
* Very complex and error prone method
* Best to let SQL engine to do

Isolation Trade offs

* More isolations provide stronger guarantees but bigger performance hits.

Week 9 Eventual Consistency

Client-side consistency vs server side consistency

Client-side

* A Storage system
* Process A
  + This process writes to and reads the storage system
* Process B and C
  + Independent of Process A and write and read from a storage system.