1-Diamond and Delegation Inheritance?

Delegation inheritance:

The delegation inheritance inherits only fields and methods are not inherited. It can be useful, when we need to embed a model in our current model without affecting the existing views, but we want to have the fields of inherited objects.

Diamond:

- If class B and class C inherit from class A, they will have attributes from class A, and if class D inherit from B and C (multiple inheritance), class D will inherit attributes of class A two times and it's a problem.
- So, we put 'virtual' keyword before class B and class C to solve this problem

Diamond inheritance problem" is some kind of steady expression which formed many years ago. You can easily find a lot of articles suggesting usage of "virtual public" to avoid the ambiguity and so on. For instance.

2-what is Modified Harvard architecture?

Those modifications are various ways to loosen the strict separation between code and data, while still supporting the higher performance concurrent data and instruction access of the Harvard architecture.

A pure Harvard architecture computer suffers from the disadvantage that mechanisms must be provided to separately load the program to be executed into

instruction memory and any data to be operated upon into data memory. Additionally, modern Harvard architecture machines often use a read-only technology for the instruction memory and read/write technology for the data memory. This allows the computer to begin execution of a pre-loaded program as soon as power is applied. The data memory will at this time be in an unknown state, so it is not possible to provide any kind of pre-defined data values to the program.

The solution is to provide a hardware pathway and machine language instructions so that the contents of the instruction memory can be read as if they were data. Initial data values can then be copied from the instruction memory into the data memory when the program starts. If the data is not to be modified (for example, if it is a constant value, such as pi, or a text string), it can be accessed by the running program directly from instruction memory without taking up space in data memory (which is often at a premium).

What are the types of platforms and example for each one.

1. Utility Platforms
    - Utility Platforms attract their users by providing useful and also usually free service.
    - ex. Google Search, Bing, Kayak, Skyscanner
2. Content Distribution Platforms
    - Such platforms connect owners with content who are wishing to deliver content (or ads) to users.
    - ex. Google AdSense, PropellerAds, and Millennial Media

3. Data Harvesting Platforms -These platforms offer useful services to users and generate data through usage of platform services. Data collected from all users of platform is fed back to service, thus making it more useful for users.
    - ex. Google Maps, Waze, Salesforce, OpenSignal, and InsideSales
4. Interaction Networks
    - These sorts of platforms facilitate interactions between specific participants (people and/or businesses). Digital interactions can be in form of message, voice call, image, or money transfer.
    - ex. Facebook, WeChat, Telegram, Ello, and Bitcoin
5. Technology Platforms
    - Technology Platforms provide building blocks or services that are reused during sizable amount of products.
    - ex. Amazon Web Services and Microsoft Azure
6. Marketplaces
    - These are two-sided platforms connecting supply with demand. Marketplaces enable transactions between demand-side participants, and supply-side participants - Amazon, eBay, Flipkart, Kickstarter, or UpWork
    - ex. Amazon, eBay, Flipkart, Kickstarter, or UpWork
7. On-demand Service Platforms
    - These types of platforms offer end-to-end services to be fulfilled by group of independent service providers or contractors.
    - ex. Uber, DoorDash, Go-Mart, and Doz
8. Computing Platforms
    - Computing Platforms render interactions between platform users and third-party developers.
    - ex. Apple iOS, Google Android

9. Content Crowdsourcing Platforms
    - These types of platforms collect content from users in form of videos, blog posts, reviews, etc, and share this content with wide range of users.
    - ex. YouTube, Crackle, Twitch, and Yelp.

What is virtual memory and when we use it ?

1-Virtual memory is a section of volatile memory created temporarily on the storage drive. It is created when a computer is running many processes at once and RAM is running low.

The operating system makes part of the storage drive available to use as RAM. Virtual memory is much slower than main memory because processing power is being taken up by moving data around, rather than just executing instructions. The guide on operating systems explains how the OS manages memory.

Latency is increased when the computer needs to use virtual memory.

Swapping is the process the OS uses to move data between RAM and virtual memory. The OS moves data from processes that are not immediately needed out of the RAM and stores them in virtual memory. It copies the data back into RAM when the process is needed again.

Using virtual memory slows the computer down because copying to a hard disk takes much longer than reading and writing RAM.

5-All types of structured and unstructured database!

1-Structure database:

1- Hierarchical database model

2- Relational model

3- Network model

4- Object-oriented database model

5- Entity-relationship model

6- Document model

7- Entity-attribute-value model

8- Star schema

9- The object-relational model

10-Inverted file model

11-Flat model

12-Multidimensional model

13-Semantic model

2-unstructured database:

- In applications:
    1. NoSQL (non-relational) databases
    2. Data lakes
    3. Data warehouses.
- Platforms:
    1. MongoDB Atlas: are especially well-suited for housing, managing, and using unstructured data.

What is Hadoop, Hive, Spark, MapReduce, and Hbase ?

Hadoop

- Apache Hadoop software is a framework that allows for the distributed processing of large data sets across clusters of computers using simple programming models.
- It is designed to scale up from single servers to thousands of machines, each offering local computation and storage.
- Rather than rely on hardware to deliver high-availability, the library itself is designed to detect and handle failures at the application layer, so delivering a highly-available service on top of a cluster of computers, each of which may be prone to failures.

Hive

- Apache Hive is a distributed, fault-tolerant data warehouse system that enables analytics at a massive scale.
- A data warehouse provides a central store of information that can easily be analyzed to make informed, data driven decisions.
- Hive allows users to read, write, and manage petabytes of data using SQL.
- Hive is built on top of Apache Hadoop.

Spark

- Apache Spark is a data processing framework that can quickly perform processing tasks on very large data sets.
- It can also distribute data processing tasks across multiple computers, either on its own or in tandem with other distributed computing tools.
- Spark can be deployed in a variety of ways, provides native bindings for the Java, Scala, Python, and R programming languages, and supports SQL, streaming data, machine learning, and graph processing.

MapReduce

- MapReduce is a programming model or pattern within the Hadoop framework that is used to access big data stored in the Hadoop File System (HDFS).
- It is a core component, integral to the functioning of the Hadoop framework.

- MapReduce facilitates concurrent processing by splitting petabytes of data into smaller chunks, and processing them in parallel on Hadoop commodity servers.
- In the end, it aggregates all the data from multiple servers to return a consolidated output back to the application.

Hbase

- HBase is a column-oriented non-relational database management system that runs on top of Hadoop Distributed File System (HDFS). HBase provides a fault-tolerant way of storing sparse data sets, which are common in many big data use cases. It is well suited for real-time data processing or random read/write access to large volumes of data.
- Unlike relational database systems, HBase does not support a structured query language like SQL; in fact, HBase isn't a relational data store at all. HBase applications are written in Java™ much like a typical Apache MapReduce application. HBase does support writing applications in Apache Avro, REST and Thrift.
- An HBase system is designed to scale linearly. It comprises a set of standard tables with rows and columns, much like a traditional database. Each table must have an element defined as a primary key, and all access attempts to HBase tables must use this primary key.

7- Comparison between Google Cloud Platform, AWS , Microsoft Azure , Cloudera ,is there replicable for them ? What are them?

1. Computer Services

AWS

1. AWS Beanstalk
2. Amazon EC2
3. Amazon Elastic Container Registry
4. Amazon Lightsail
5. AWS Batch, AWS Fargate, AWS Lambda, and AWS Outposts
6. VMware Cloud for AWS
7. AWS Serverless Application Reprository

Azure

1. Platform-as-a-service (Paas)
2. Function-as-a-service (Faas)
3. Service Fabric
4. Azure Batch
5. Azure Container Service
6. Virtual Machines Compute Engine
7. Virtual Machine Scale Sets

GCP

1. App Engine
2. Docker Container registry
3. Instant Groups
4. Compute Engine
5. Graphics Processing Unit (GPU)

6. Knative
7. Kubernetes

2. Storage Services

AWS

1. Simple Storage Service (S3)
2. Elastic Block Storage (EBS)
3. Elastic File Storage (EFS)
4. Storage Gateway
5. Snowball, Snowball Edge, and Snowmobile

Azure

1. Blob Storage
2. Queue Storage
3. File Storage
4. Disk Storage
5. Data Lake Store

GCP

1. Cloud Storage
2. Persistent Disk
3. Transfer Appliance
4. Transfer Service

3. AI/ML

AWS

1. SageMaker and Comperhend
2. DeepLans
3. Lex, Polly, and Rekognition
4. Machine Learning
5. Deep Learning AMIs
6. TensorFlow on AWS
7. Apach MXNet on AWS

Azure

1. Machine Learing
2. Azure Bot Service
3. Cognitive Service

GCP

1. Cloud Machine Learning Engine
2. Cloud Natural Language
3. Cloud Speach API
4. Cloud Translation API
5. Cloud Video Intelligence
6. Cloud Job Discovery (Private Beta)
7. Dialogflow Enterprise Edition

4. Database

AWS

1. Aurora
2. RDS
3. DynamoDB
4. ElastiCache
5. Redshift
6. Neptune

Azure

1. SQL Database
2. Database for MySQL
3. Data Warehouse
4. Database for PostgreSQL
5. Cosmos DB
6. Server Streach Database
7. Table Storage
8. Redis Cache
9. Data Factory

GCP

1. Cloud SQL
2. Cloud Bigtable
3. Cloud Spanner
4. Cloud Datastore

5. Backup Services

AWS

1. Glacier

Azure

1. Archive Storage
2. Backup
3. Site Recovery

GCP

1. Nearline (frequently accessed data)
2. Coldline (infrequently accessed data)

6. Networking Services

AWS

1. Amazon Virtual Private Cloud (VPC)

Azure

1. Azure Virtual Network (VNET)

GCP

1. Cloud Virtual Network

7. Automation Services

AWS

1. AWS Opsworks

Azure

1. Azure Automation

GCP

1. Compute Engine Management

Free alternatives

1. pCloud, Sync.com and Icedrive are the best free cloud storage providers due to their combination of features, ease of use, security and privacy.
2. MEGA offers more storage than any other free cloud storage service with 20GB by default, which can be further expanded for a single year.
3. Google Drive, OneDrive and iCloud are all good free cloud storage providers if you're already invested in their respective ecosystems and aren't overly concerned with privacy.
4. IDrive, despite being an online backup service, offers enough hybrid features to qualify as one of the best free cloud storage providers.

What are the 11 V's for Big Data ?

1. Volume
    - Volume refers to the huge amuont of data.
2. Velocity
    - Velocity refers to the speed at which data is being generated, produced, created, or refreshed.
3. Varity
    - Varity refers to that data is different.
4. Value
    - Value refers to that data is meaningful.
5. Veracity
    - Veracity refers more to the provenance or reliability of the data source, its context, and how meaningful it is to the analysis based on it.
6. Validity
    - validity refers to how accurate and correct the data is for its intended use.
7. Variability
    - Variability refers to the number of inconsistencies in the data.
8. Venue
    - Venue refers to Various types of data arrived from different sources via different platforms.
9. Volatility
    - Volatility refers to the stored data and how long is useful to the user.
10. Visualization
    - Visualization refers to the process of representing abstract.
11. Complexity
    - Complexity refers to Correlation of Data.

9- What is sql injection ?

SQL injection (SQLi) is a web security vulnerability that allows an attacker to interfere with the queries that an application makes to its database. It generally allows an attacker to view data that they are not normally able to retrieve. This might include data belonging to other users, or any other data that the application itself is able to access. In many cases, an attacker can modify or delete this data, causing persistent changes to the application's content or behavior.

A successful SQL injection attack can result in unauthorized access to sensitive data, such as passwords, credit card details, or personal user information. Many high-profile data breaches in recent years have been the result of SQL injection attacks, leading to reputational damage and regulatory fines. In some cases, an attacker can obtain a persistent backdoor into an organization's systems, leading to a long-term compromise that can go unnoticed for an extended period.

Some common SQL injection examples include:

Retrieving hidden data, where you can modify an SQL query to return additional results.

Subverting application logic, where you can change a query to interfere with the application's logic.

UNION attacks, where you can retrieve data from different database tables.

Examining the database, where you can extract information about the version and structure of the database.

Blind SQL injection, where the results of a query you control are not returned in the application's responses.

Compare between Relational and Non-relational database.

Relational Database

- A relational database is structured, meaning the data is organized in tables. Many times, the data within these tables have relationships with one another, or dependencies.

SQL Databases (Relational)

- Basically meaning a very firm way of sorting through data in the form of tables, columns, and rows.
- How is data structured in an SQL database?
    - The table itself would be made up really of one variable or object that we would be looking through.
    - The column would represent the data point itself that needs to be stored and the row is a record of the data points per column.
- Some popular SQL database systems include:
    - Oracle
    - Microsoft SQL Server
    - PostgreSQL
    - MySQL
    - MariaDB

What is DevOps and what are the DevOps tools

- DevOps is a process of software development automation combines between development process and IT operation process.
- DevOps can be best explained as people working together to conceive, build and deliver secure software at top speed. DevOps practices enable software developers (devs) and operations (ops) teams to accelerate delivery through automation, collaboration, fast feedback, and iterative improvement.
- The DevOps methodology comprises four key principles that guide the effectiveness and efficiency of application development and deployment. These principles, listed below, center on the best aspects of modern software development.
- 1-Automation of the software development lifecycle
-
- 2-Collaboration and communication
-
- 3-Continuous improvement and minimization of waste
-
- 4-Hyperfocus on user needs with short feedback loops
-

DevOps Tools

1. QuerySurge
    - QuerySurge is the smart data testing solution that is the first-of-its-kind full DevOps solution for continuous data testing.
2. Basis Technologies
    - Basis Technologies offers the only DevOps and test automation platform engineered specifically for SAP. Their DevOps for SAP tool, ActiveControl allows businesses to move their SAP applications from

fixed release cycles to an on-demand delivery model based on CI/CD and DevOps.

3. Datadog
    - Datadog provides a complete monitoring solution so that you have full visibility as you transform or adopt your DevOps culture.
4. Solarwinds DevOps
    - SolarWinds DevOps helps you to reduce server load with an agentless architecture. It enables you to check the performance of your network, servers, and VMware and simplify the daily tasks of your organization.
5. Jenkins
    - Jenkins a DevOps tool for monitoring execution of repeated tasks. It is one of the best software deploy tools which helps to integrate project changes more easily by quickly finding issues.

12- Difference between white box and black box testing

Black Box:

1-The Black Box Test is a test that only considers the external behavior of the system; the internal workings of the software is not taken into account.

2-It is carried out by testers.

3-This method is used in System Testing or Acceptance Testing.

4-It is the behavior testing of the software.

5-It is also known as data-driven testing, functional testing, and closed box testing.

6-Black Box Test is not considered for algorithm testing.

White Box:

1-The White Box Test is a method used to test a software taking into consideration its internal functioning.

2-It is carried out by software developers.

3-This method is used in Unit Testing or Integration Testing.

4-It is the logic testing of the software.

5-It is also known as clear box testing, code-based testing, structural testing, and transparent testing.

6-White Box Test is well suitable for algorithm testing.

13- Difference between verification and validation testing

Verification Testing

- Verification in Software Testing is a process of checking documents, design, code, and program in order to check if the software has been built according to the requirements or not.

- The main goal of verification process is to ensure quality of software application, design, architecture etc. The verification process involves activities like reviews, walk-throughs and inspection.

Validation Testing

- Validation in Software Engineering is a dynamic mechanism of testing and validating if the software product actually meets the exact needs of the customer or not.
- The process helps to ensure that the software fulfills the desired use in an appropriate environment. The validation process involves activities like unit testing, integration testing, system testing and user acceptance testing

---

14- What is V model in software engineering ?

- The V-model is a type of SDLC model where the process executes in a sequential manner in a V-shape. It is also known as the Verification and Validation model. It is based on the association of a testing phase for each corresponding development stage. The development of each step is directly associated with the testing phase. The next phase starts only after completion of the previous phase i.e., for each development activity, there is a testing activity corresponding to it.
- The V-Model is a software development life cycle (SDLC) model that provides a systematic and visual representation of the software development process. It is based on the idea of a “V” shape, with the two legs of the “V” representing the progression of the software development process from

requirements gathering and analysis to design, implementation, testing, and maintenance.