1. **Explain JNDI ?**

It is an Java Naming and Directory Interface is an API (Application Programming Interface) used to access the services which is reside on the application server by its name. Which allows distributed applications lookup services in an abstract and resource independent way. To use the JNDI , we need JNDI classes and service providers. JDK itself has the service providers to access the following JNDI services

* LDAP (Lightweight Directory Access Protocol)
* RMI (Remote Method Invocation)
* CORBA – Common Object Request Broker Architecture
* DNS – Domain Name Service

**When it is used?**

It is used to set up the database connection pool in the J2EE Server. Any applications that is deployed in that server get the database connection using the JNDI name (“java/testJNDI”) without know about the database connection details.

**Advantages:**

When we move code to the different environment ( dev->QA->SIT->UAT->PROD ) we can use the same JNDI name in all environments and hide the database details. Applications doesn’t require any change when they promote to different environments. In this way, we can restrict the production database credentials to be shared to many users except server admin.

1. **How to use the JNDI to lookup and get the data source connection ?**

* Obtain the Initial Context

InitialContext ctx=new InitialContext()

* Use JNDI lookup to obtain the datasource reference

DataSource ds=ctx.lookUp(“JNDIDataSourceName”)

* Use the datasource refrence to get the connection

Connection con=ds.getConnection()

1. **What is websphere objectgrid ?**

**Ref ObjectGrid API:** [**https://www.ibm.com/support/knowledgecenter/SSTVLU\_8.6.1/com.ibm.websphere.extremescale.javadoc.doc/topics/com/ibm/websphere/objectgrid/package-summary.html#package\_description**](https://www.ibm.com/support/knowledgecenter/SSTVLU_8.6.1/com.ibm.websphere.extremescale.javadoc.doc/topics/com/ibm/websphere/objectgrid/package-summary.html#package_description)

It is an distributed cache product available on websphere application server deployment platform. The main interface used here is ObjectGrid. The JVM needs to create atlease one instance of this type.

It is designed to be as data cache ,collects data from multiple source and available to the client of the ObjectGrid. It can store large quantities of the data. The stored data are access by using the ObjectGrid API’s.

**Working on ObjectGrid:**

An application can have multiple ObjectGrid Instances and each has its own config file.

An ObjectGrid instance needs to be created first,

ObjectGrid obg=new ObjectGridImpl(); 🡪 ObjectGrid instance is created

BackingMap bm=obg.defineMap(“TABLE”); -> Map defined with the created ObjectGrid

BackingMap setter method is used to configure the map.

**Working on the ObjectGrid Session:**

Each thread has its own session to work on the ObjectGrid . The OBjectgrid getSession() method returns the session. Once the thread gets the session , it can access the ObjectMap to work on the manipulation of the data available on the ObjectGrid. Commit / Rollback /begin method from the session are used to handle the transation.

Session session=obg.getSession();

ObjectMap map=session.getMap(“TABLE”);

Session.begin();

MyData d1=(MyData)map.get(“key1”);

Session.commit();

1. **Name some distributed cache products?**

* **SwarmCache**
* **JBoss TreeCache**

1. **How to handle singleton in distributed environment or clustered environment?**

Singleton means once per JVM. In distributed environment , there will be multiple JVM and there will be one singleton instance per JVM. Singleton is used when we have some configuration details that needs to be shared across the application. So Singleton instance holds this config data that can be accessed throughout the application. Singleton pattern restrict to create more than one instance at single JVM.

There are so Many options are available to maintain the singleton across multiple JVM’s in distributed environment.

1. **Usage of RMI :**

We can register the singleton instance in the RMI registry in one node and provide the stub in the clustered JNDI tree to make it available across all the nodes. All other nodes should be a RMI clients. The disadvantage is , single point of failure. We should have a failure over node for central node which also has the singleton instance should be registered with the clustered JNDI tree when there is no singleton stub registered by the central node.

1. **Usage of JMS:**

There may be situation, singletons are available in multiple JVM’s as data cache.(map). Whenever data changed in data cache it needs to published into TOPIC using the Java Messaging API. Every container has listener which listens the topic and get the message data and update into their cache. Disadvantage here is , JMS messages takes time to process , so there may be some time the caches are out of sync.

1. **Usage of Distributed Cache:**

To overcome the problem with JMS. We can go for distributed singleton cache. SwarmCache and JBoss TreeCache are the good example of distributed caches.

1. **Usage of Vendor Products:**

There are so many vendor products are available to maintain the singleton in the clustered environment.

* **Terracotta / Oracle Coherence:**

Which use the concept of in-memory object replication to provide the singleton view across all JVM’s.

* **JGroups :**

It is a toolkit for reliable messaging. It creates the cluster whose node can send messages to each other. It allows to form the group and applications and JGorups send messages to everyone in the group so that they can be in sync.

* **JBoss - HASingtonService :**

Singletone service is deployed in all nodes in a cluster but only one runs. This is called master node. When the master fails , another master will selected from the cluster and the service is restarted.

* **Weblogic - SingletonService:**

Only one instance runs in the cluster and all clients are lookup the same instance.

* **Websphere - ObjectGrid**

Websphere supports singleton across cluster via ObjectGrid.

1. **What are the advantages of Clustering?**

* High availability
* High Performance
* High Fault Tolerance

There are two important steps involved in the clustering.

* **Load Balancing :**

When multiple request from multiple clients comes in ,there may a situation one server cannot handle these request. So there will be load balancer placed between the client and server and routes these request to different servers for doing the same function. The example is servlet. This makes the high availability and high performance.

* **Fail Over :**

when client makes lots of request and the server failed to handle this request in between , there will be a fail over systems which detedcts this failure and routes the request to different available server instance.

1. **Explain about Queue & Topic ?**

Queue is persistent . it is an point to point message communication. If the message comes into to the queue , once it is read by receiver , it gets deleted from the queue. If the receiver is offline , the message still exist in the queue.

Topics are non-persistent. It is an public-subscribe message communication. If the message comes into the topic, once its subscribed by all the subscriber it gets deleted. If the subscribers are offline , the messages gets deleted in general. IF we want to make the topic persistent and flexible like queue , we have to subscribe as durable subscriber.

Durable subscription use unique name to subscribe with the topic. So it the subscribers are offline the messages are stored in the topic with the subscriber subscription name. Once the subscriber the come to online , the messages with their name will be delivered.

1. **What is LDAP ?**

**Sanitize :-** To make cleanor hygienic , to remove unacceptable characters.

Lightweight Directory Access Protocol(LDAP) is an open source protocol used for querying and manipulating the information directories. It runs over transaport protocols such as TCP protocol.

1. **What is LDAP Injection ? how to prevent LDAP Injection Vulnerabilities ?**

**Arbitrary:-** based on Random choice without any reason or system.

Web application uses user input to create the LDAP statements. LDAP Injection is an technique used to create the custom LDAP statements based on user input in the web applications for dynamic web page requests. Which will create the security issue in web application. When an web application is not properly sanitize the user input , it will create some arbitrary commands such as granting permission to unauthorized queries , changing the data in LDAP tree. The attackers can use local proxy and they can change the data in the http request which constructing the LDAP statements.

Protecting the LDAP enabled web application is an responsible for the application developer and LDAP Aministrator.

**Incoming Data Validation:**

Prevent this the user input must need to be validated on server side. Most the of the code injection techniques uses special characters for the malicious attacks, so this special characters needs to be cleaned up from user input by the use of input based filters.

**Ex:**

To retrieve the employess who work on particular manager , here is the LDAP query

DirectorySearcher searcher=new DirectorySearcher(“(manager=’+**managerName.Text**+)”);

managerName.Text is an parameter value from httprequest. Under normal conditions,

manager=’smith’. The attacker can easily change the request parameter value as

(manager=’smith’)(|objectclass=\*)

It will return all the employees along with manager. To prevent this the user input which used as a LDAP search filter must be validated against list of valid values in application layer before sending it to the LDAP Server. This is called positive validation scheme.

However there is some situation special characters are required for the LDAP query .In this scenario the special characters are used with back slash character so that LDAP interpreter can assume this is part of LDAP query.

**LDAP Configuration :**

Implementing the tight access control on the LDAP may reduce the LDAP injection vulnerabilities . Restrict the LDAP access level used by the web application is also an prevention .

LDAP server should not be directly accessible on the internet.

1. **What is scalability of the application ?**

Scalability is the capacity of the system or application.

1. **How to scale our application ? What is horizontal scaling and vertical scaling?**

**Scalability: Capacity of the system**

**Horizontal scaling:** Horizontal scaling means adding more machines into the pool of resources. The productivity of each machine or node or actor is still same but the capacity is increased by adding more machines or actors or nodes. It depends on the network capability.

**Vertical scaling:** Vertical scaling means adding more power into an existing machine. Adding resources to a single node in the system. The capacity of the system is increased by adding more power(CPU) to an existing machine or actor or node.

**12) What is reengineering? What are the aspects needs to be considered while reengineering the application?**

Reengineering means restructuring or rewriting the part of the system or the entire system without affecting the existing functionalities. The reengineered system must be redocumented.

**When the system needs to be reengineered:**

* When hardware or software’s are out dated
* When the system requires frequent maintenance
* When one component or system changes affects the other system or component
* When the tools required for reengineering is readily available

**Why do we need to reengineer the system :**

* When we need to reuse the existing logic(Code base) and functionality of the legacy system , we can go for reengineering of the system. So the cost and effort spend for building the old system is not wasted.

**Objective of the reengineering:**

* Improve the maintenance
* Access to the new platform
* Enhancing the functionality
* Improved reliability

**How to reengineer:**

* Analysis the existing system
* Find out the characteristic and feature of the target system
* Create the standard set of tests to conform the transformation
* Reengineering starts with the code base of the existing systems and ends with target system source code.
* Reengineering is very complex since we have to analysis the existing code , determine the requirements , compare them with current requirement s, remove unwanted things , design the new system and code.

**Reengineering approaches:**

* There are three approaches available in software reengineering such as

1. **BIG-BANG approach:**

It replaces the entire system into new system(moving the system to different architectures)

The advantage is entire system is moved to new environment and new interfaces needs to be developed. There will not be any old piece.

Disadvantage is results are not appropriate. For large systems it requires more resources or requires large amount of time to generate the new system

1. **Added Method approach:**

It is also called phase out. It this approach , only part of the old system needs to be reengineered and moved to new system. And added all new incremental updates into the new system.

The advantage is the components of the system are produced faster. It takes less time to create the new component when compared to the first approach. And it is easy to control. Since its an interim version , customer can see the progress and easily identify the gaps. This approach has lower risk when compared to first approach. The risk of the code to be identified and tracked.

Disadvantage is whole system will not be changed only the components has changed. This requires careful identification of the components of the existing system

1. **Evolutionary approach:**

This is called as method of evaluation. The part of the system is reengineered into new system. The part of the system is selected based on the functionality not based on the structure or component. The new system is built based on the functionality. The components of the current system is divided by function and the new system is created.

The advantage is, this approach gives modular design and well suited for the object oriented technology.

Disadvantage is same function must be identified in different components of the current system.

**Stages involved in the Reengineering:**

There are five stages involved in the reengineering.

1. **Establish Reengineering:**

In this stage, the project manager is responsible to handle the technical challenge and maintenance. They are also responsible for identifying, purchasing and testing the new tools or software that are required for the reengineering. The have to make sure that all employees (team members) are get trained well in the new technology and use them effectively. Though the team members have the software knowledge , they have to get trained in the new technology.

1. **Analyze the feasibility of the project:**

The reengineering team must assess the needs of the organization and the goal of the new system. They have to properly analyze the value of the current system and determine what has to be improved in the new systems such as maintenance efficiency , maintenance cost , software quality and value of the current system.

1. **Analysis and Planning:**

This involves 3 steps:

1. Current system characteristic, specification and quality needs to be identified. This starts with identifying the requirement documents, functional documents, design documents , use cases . These are needs to be helpful to determine the direction. Once the this is done , current status of the current system , maintenance needs to be determined for the capital of the target system. Then set of software metrics needs to be set and asses the current system to determine the quality . The same metrics should be used tat the end of reengineering to identify the quality of the target system.
2. Once the current system characteristic, specification, quality and value is determined , we have to identify the specification and value of the target system such as hardware, software, operating systems, language and design.
3. Then create the set of test standards to explain the functionality of the new system is equivalent of the old system.
4. **Implement Reengineering:**

Once the analysis of the current system is done , the implementation of the new system starts. This involves two steps

1. **Reverse Engineering:**
2. **Forward engineering:**
3. **Conversion and Testing:**

Once the implementation is done , testing needs to be done to detect the errors. The same test cases should executed on legacy and new system and results to be compared to make sure the functionality is achieved. Test results should be douemented.

1. **What are non-functional requirements?**

**Functional Requirements: -** It describes what the system should do. It specifies the function and behavior of the system or component. It is described as documents or use cases. Use cases are the diagrammatic representation of the system behavior. It mostly covers the following,

1. Calculations
2. Technical Details
3. Data manipulation
4. Data processing

**Non Functional Requirements: -** It specifies how the system do. It specifies how the system should behave. It is used to judge the systems operation rather than specific behaviour. It describes the performance characteristic of the system. It is not the straight forward requirement of the system.

The non-functional requirements fall into the following areas.

1. Security
2. Scalability
3. Maintainability
4. Reliability
5. Efficiency
6. Stability
7. Testability
8. Supportability
9. Accessibility
10. Extensibility
11. Portability
12. Interoperability

**14) What is software development methodlogy ? How do you deliver your project ? What is the Agile methodology ?**

Software development methodology is a framework for structure , plan and control the software development process.

There are so many software development methodologies available, the most frequently used are,

1. Agile Methodology
2. Scrum Methodology
3. Waterfall Methodology
4. **What are the sections available in the technical design document?**

Refer the sample documents.

1. **What is Agile development methodology? Explain**

Agile is an iterative and incremental software development methodology. The most popular agile methodology is SCRUM.

**Scrum :**

Scrum model concentrates on how to manage the task on the team based development environment. This model is suitable for the team of 7-9 members. There are 3 roles involved in this scrum.

**Roles:**

**Scrum Master:**

He is responsible for setting up the team or sprint meeting and remove obstacles(meaning is the thing that blocks one way) for progress. Scrum meeting usually happens 15 mins on daily basis.

**Product Owner:**

He creates the product backlog and prioritize the product backlog and responsible to deliver the product for each iteration. The product backlog is the repository where the requirements are maintained with details on the no of requirement to be completed for each release. It should be distributed to team members. Team members can raise request to add, modify or delete the requirements from product backlog.

**Scrum Team :**

They are responsible for their own work and organize their work and deliver the product at end of each iteration

**Process Flow:**

* Each iteration of the scrum is called as sprint
* Product backlog is the list where all requirements are entered to get the end product
* During each spring. some list of requirements is fetched from product backlog and turned to sprint backlog
* Team members work on spring backlog
* Team checks their daily work
* At the end of each product, team delivers product functionality