B.Hanumanthu-ISTE60

Assignment Topic-wise Refresher on Java Full stack Development without keys

Time:60 min

Points:40

Instructions:

1. **Answer the given questions with illustrations or examples**
2. **Convert the document into pdf with responses**
3. **Upload the work in your respective folder given on the drive**
4. **Correct your responses using the keys given**
5. **Update the scorecard with the scores and upload the corrected copy along with the response copy on the drive**

Java and Core Java:

1. **What is the main difference between `==` and `.equals()` method in Java?**
Answer:

In Java, the == operator compares the two objects to see if they point to the same memory location; while the . equals() method actually compares the two objects to see if they have the same object value. It is the difference between identity and equivalence.

2. **Explain the difference between `ArrayList` and `LinkedList` in Java.**Answer:
ArrayList uses an array, which allows for fast random access but slow insertion and deletion. While LinkedList uses a doubly linked list, which allows for fast insertion and deletion but slow random access. Also one of the major difference lies in the access time.

3. **What is the purpose of the `static` keyword in Java?**Answer:
The most important reason why static keywords are heavily used in Java is to efficiently manage memory. Generally, if you want to access variables or methods inside a class, you first need to create an instance or object of that class.

4. **How does Java handle multiple inheritance?**
Answer:

The Java programming language supports multiple inheritance of type, which is the ability of a class to implement more than one interface. An object can have multiple types: the type of its own class and the types of all the interfaces that the class implements.

5. **What is the difference between an abstract class and an interface in Java?**
Answer:
An abstract class contains an abstract keyword on the declaration whereas an Interface is a sketch that is used to implement a class.

6. **Explain the concept of multithreading in Java.**Answer:
Multithreading in Java is an act of executing a complex process using virtual processing entities independent of each other. These entities are called threads. Threads in Java are virtual and share the same memory location of the process. As the threads are virtual, they exhibit a safer way of executing a process.

7. **What is the purpose of the `super` keyword in Java?**
Answer:
The super keyword refers to superclass (parent) objects. It is used to call superclass methods, and to access the superclass constructor. The most common use of the super keyword is to eliminate the confusion between superclasses and subclasses that have methods with the same name.

8. **How does exception handling work in Java?**Answer:
In Java, exception handling is implemented using the try-catch-finally blocks. When an exception is thrown, the Java Virtual Machine (JVM) starts a search for the appropriate exception handler by going up the call stack, checking each method in turn for a catch block that can handle the exception.

9. **What is a lambda expression in Java? Provide an example.**Answer:
Lambda Expressions were added in Java 8. A lambda expression is a short block of code which takes in parameters and returns a value. Lambda expressions are similar to methods, but they do not need a name and they can be implemented right in the body of a method.

10. **Explain the significance of the `final` keyword in Java.**
Answer:
The final keyword is a non-access modifier used for classes, attributes and methods, which makes them non-changeable (impossible to inherit or override). The final keyword is useful when you want a variable to always store the same value, like PI (3.14159...). The final keyword is called a "modifier".

SOLID Principles:

11. **Explain the Single Responsibility Principle (SRP).**
Answer:
The Single Responsibility Principle (SRP) is the concept that any single object in object-oriented programing (OOP) should be made for one specific function. SRP is part of SOLID programming principles put forth by Robert Martin. Traditionally, code that is in keeping with SRP has a single function per class.

12. **What does the Open/Closed Principle (OCP) state?**
Answer:
The open-closed principle states that software entities (classes, modules, functions, and so on) should be open for extension, but closed for modification.

13. **Describe the Liskov Substitution Principle (LSP).**
Answer:
The Liskov Substitution Principle (LSP) is a fundamental principle in object-oriented programming that states that objects of a superclass should be able to be replaced with objects of a subclass without affecting the correctness of the program.

14. **Explain the Interface Segregation Principle (ISP).**
Answer:
In the field of software engineering, the interface segregation principle (ISP) states that no code should be forced to depend on methods it does not use. ISP splits interfaces that are very large into smaller and more specific ones so that clients will only have to know about the methods that are of interest to them.

15. **What is the Dependency Inversion Principle (DIP) and how does it work?**
Answer:
The Dependency Inversion Principle (DIP) states that high level modules should not depend on low level modules; both should depend on abstractions. Abstractions should not depend on details. Details should depend upon abstractions.

DBMS:

16. **What is normalization in the context of databases?**
Answer:
Normalization is the process of organizing data in a database. It includes creating tables and establishing relationships between those tables according to rules designed both to protect the data and to make the database more flexible by eliminating redundancy and inconsistent dependency.

17. **Explain the difference between a primary key and a foreign key.
Answer:

A primary key generally focuses on the uniqueness of the table. It assures the value in the specific column is unique. A foreign key is generally used to build a relationship between the two tables. The table allows only one primary key.

18. **What is an index in a database?**Answer:
An index is a copy of selected columns of data, from a table, that is designed to enable very efficient search. An index normally includes a "key" or direct link to the original row of data from which it was copied, to allow the complete row to be retrieved efficiently.

19. **Describe the ACID properties in the context of database transactions.**
Answer:
ACID is an acronym that refers to the set of 4 key properties that define a transaction: Atomicity, Consistency, Isolation, and Durability. If a database operation has these ACID properties, it can be called an ACID transaction, and data storage systems that apply these operations are called transactional systems.

20. **Explain the concept of a stored procedure.**Answer:
A stored procedure is a prepared SQL code that you can save, so the code can be reused over and over again. So if you have an SQL query that you write over and over again, save it as a stored procedure, and then just call it to execute it.

JSP and Servlet:
21. **What is the difference between JSP and Servlet?**
Answer:
Servlets are Java-based codes. JSP are HTML-based codes. Servlets are harder to code, as here, the HTML codes are written in Java. JSPs are easier to code, as here Java is coded in HTML.

22. **Explain the life cycle of a Servlet.**
Answer:
Servlet life cycle can be defined as **the stages through which the servlet passes from its creation to its destruction. Stages of the Servlet Life Cycle**: The Servlet life cycle mainly goes through four stages,

- Loading a Servlet.
- Initializing the Servlet.
- Request handling.
- Destroying the Servlet.

23. **What is a JSP expression?**
Answer:

A JSP expression is used to insert the value of a scripting language expression, converted into a string, into the data stream returned to the client.

24. **What is the purpose of the `web.xml` file in a Java web application?**Answer:
web. xml defines mappings between URL paths and the servlets that handle requests with those paths. The web server uses this configuration to identify the servlet to handle a given request and call the class method that corresponds to the request method.

25. **How can session management be handled in Servlets and JSP?**
Answer:
There are four techniques used in Session tracking:

1. Cookies.
2. Hidden Form Field.
3. URL Rewriting.
4. HttpSession.

Tomcat:
26. **What is Apache Tomcat, and what is its role in Java web development?**
Answer:
Apache Tomcat (called "Tomcat" for short) is a free and open-source implementation of the Jakarta Servlet, Jakarta Expression Language, and WebSocket technologies. It provides a "pure Java" HTTP web server environment in which Java code can also run.

27. **Explain the difference between Tomcat and Apache HTTP Server.**Answer:
The Apache server is an HTTP web server, while the Apache Tomcat server is mainly a Java application server. Tomcat is written in Java, while Apache is written in C. Tomcat is used to serve dynamic content such as Java Servlets and JSP files, while Apache is used to serve static content.

28. **How can you deploy a web application in Tomcat?**
Answer:
Tomcat is a very popular web server/servlet container that can host Java web applications which are made up of servlets, JSP pages (dynamic content), HTML pages, javascript, stylesheets, images. The most common ways about how to deploy a Java web application on Tomcat, include the followings:

- Copying web application archive file (.war).
- Copying unpacked web application directory.
- Using Tomcat’s manager application.

29. **What is the purpose of the `server.xml` file in Tomcat?**Answer:
Tomcat server. xml file is used to set the configurations of each of the modules and their corresponding properties related to the tomcat web server. There are various modules present in the tomcat server each of them having certain properties all of which are specified inside the server. xml file.

30. **How does Tomcat handle HTTP requests and responses?**
Answer:
Tomcat uses connectors (such as the HTTP Connector or AJP Connector) to handle incoming client requests. Each connector is associated with a pool of worker threads. These worker threads are responsible for processing client requests, reading incoming data, and generating responses.

Git:

31. **What is Git, and how does it differ from other version control systems?**Answer:
Git is known for its speed, workflow compatibility, and open source foundation. Most Git actions only add data to the database, and Git makes it easy to undo changes during the three main states. Git has three file states: modified, staged, and committed .

32. **Explain the difference between Git commit and Git push.**
Answer:
Git commit saves repository changes on local but not remote repository. Contrarily, Git push then updates your git commit changes and sends it to remote repository.

33. **What is a branch in Git, and why is it useful?**
Answer:
Git branches are effectively a pointer to a snapshot of your changes. When you want to add a new feature or fix a bug—no matter how big or how small—you spawn a new branch to encapsulate your changes.

34. **Describe the process of resolving a merge conflict in Git.**
Answer:
Suppose two persons are working on the same file called index. html.

1. Fetch and merge the latest changes from the remote repository: $ git pull.
2. Identify the one or more conflicting files: $ git status.
3. Open the conflicting file using a text editor: $ vim index.html.
4. Resolve the conflict.

35. **Explain the purpose of the `.gitignore` file.**
Answer:

The purpose of gitignore files is to ensure that certain files not tracked by Git remain untracked. To stop tracking a file that is currently tracked, use git rm --cached to remove the file from the index. The filename can then be added to the .gitignore file to stop the file from being reintroduced in later commits.

UML:

36. **What does UML stand for, and what is its purpose in software development?**
Answer:
UML, which stands for Unified Modeling Language, is a way to visually represent the architecture, design, and implementation of complex software systems.

37. **Explain the difference between class diagrams and sequence diagrams in UML.**
Answer:
A class diagram describes the structure of the system or the details of an implementation, while a sequence diagram shows the interaction between objects over time. In addition, models can also contain profiles which can be predefined or customized to a particular system or application.

38. **What is the purpose of an activity diagram in UML?**
Answer:
An activity diagram is a type of Unified Modeling Language (UML) flowchart that shows the flow from one activity to another in a system or process. It's used to describe the different dynamic aspects of a system and is referred to as a 'behavior diagram' because it describes what should happen in the modeled system.

39. **Describe the use of the association relationship in UML.**Answer:
In UML models, an association is a relationship between two classifiers, such as classes or use cases, that describes the reasons for the relationship and the rules that govern the relationship. An association represents a structural relationship that connects two classifiers.

40. **What is the purpose of the UML use case diagram?**
In UML, use-case diagrams model the behavior of a system and help to capture the requirements of the system. Use-case diagrams describe the high-level functions and scope of a system. These diagrams also identify the interactions between the system and its actors.