1. **Commit, Rollback, Savepoint, Set Transactions are**

A . DDL(Create, Alter, Rename, Truncate)

B. TCL Commands(Transaction Control language)

C. DCL Commands(Select where)

D. Not Sure

**Ans: B. TCL Commands**

1. **Maven Searches for dependencies in the order of**
2. Remote->Central->Local
3. Local->Central->Remote
4. Central->Local->Remote
5. Central->Remote->Cloud

**Ans: B. Local->Central->Remote**

1. **Java instance of operator used to**
2. test whether the object is an instance of the specified type (class or subclass or interface).
3. The instanceof java is also known as type *comparison operator* because it compares the instance with type.
4. It returns either true or false.
5. All of the above

**Ans: D. All of the above**

1. **Maven has main tasks like**
2. Makes project easy to build
3. Provide a uniform build process
4. Provides a unique dependency list
5. All of the above

**Ans: D. All of the above**

1. **How many types of JDBC Drivers are available**
2. 1
3. 2
4. 4
5. All of the above

**Ans: 4**

1. **What is mocking**
2. Developing objects that acts as mock clone of real objects
3. Mock objects are created for testing purpose
4. Dummy output for Dummy input
5. All of the above

**Ans: D. All of the above**

1. **Mockito uses Reflection API to generate mock objects**
2. TRUE
3. FALSE
4. Cannot say
5. None of the above

**Ans: A. TRUE**

1. **The classes Matcher and Pattern provide a facility for java regular expression**
2. They are in java.util.regex Package
3. TRUE
4. Both 1 and 2
5. None of the above

**Ans: C. Both 1 and 2**

1. **Liquibase is a database change management tool**
2. TRUE
3. FALSE
4. Cannot say
5. Not sure

**Ans. A. RTUE**

1. **Following are the types of maven repositories**
2. Cloud and local
3. Local, Central, Remote
4. Only remote
5. Only Central

**Ans: B. Local, Central, Remote**

1. **Mocking concept covers Stub, Fake, and Mock**
2. Mocking plays essential roles for under development components
3. TRUE
4. FALSE
5. Both 1 and 2

**Ans. B.TRUE**

1. **Five DDL Commands are Create, Drop, Alter, Truncate, Rename**
2. TRUE
3. FALSE
4. Cannot say
5. Not Sure

**Ans: A. TRUE**

1. **Liquibase can be used for Create, Update and Delete Operations**
2. TRUE
3. FALSE
4. Cannot say
5. Not sure

**Ans. A. TRUE**

1. **Main Tasks of JVM are**
2. Verify File, Load File, Execute Code, Provide Run Time
3. Load File, Verify File, Execute Code, Provide Run Time
4. Provide Run Time
5. All of the above

**Ans. B. Load File, Verify File, Execute Code, Provide Run Time**

1. **Insert, Update, Delete, Lock are**
2. DCL Commands
3. DML Commands
4. Cannot say
5. Not sure

**Ans. DML Commands**

1. **Derived classes must be substitutable by Base classes is**
2. Liskov substitution Principal
3. Barbara Liskov’s principal
4. Not Sure
5. Cannot say

**Ans. A. Liskov substitution Principal**

1. **Maven Address**
2. Adding Set of Jars/Dependencies to the project
3. Creating right project structure
4. Building and Deploying project
5. All of the above

**Ans: D. All of the above**

1. **Liquibase can allow to track difference between two databases**
2. Yes changelog as difference between two databases can be generated
3. No
4. Not Sure
5. Cannot say

**Ans: A. Yes changelog as difference between two databases can be generated**

1. **JDBC is an API to**
2. Connect with database
3. Create Database
4. Query Database
5. All of the above

**Ans. D. All of the above**

1. **Partial Mocking of an object is provided by**
2. spy() method
3. verify() method
4. setup() method
5. None of the above

**Ans. A. spy() method**

1. **Object class is**
2. Object class parent class of all the classes in Java by default in other words
3. Object class beneficial if you want to refer any object whose type you do not know
4. The Object class provides some common behaviours to all the objects such as object can be compared
5. All of the above

**Ans. D. All of the above**

1. **Method overloading is**
2. A class has multiple methods with same name and different parameters
3. Easy for same operation to be performed differently as required
4. Both 1 and 2
5. None of the above

**Ans: C. Both 1 and 2**

1. **In Liquibase queries are written in database neutral language like XML, JSON, YAML**
2. TRUE
3. FALSE
4. Cannot say
5. Not Sure

**Ans. TRUE**

1. **Java is Platform Independent**
2. TRUE
3. FALSE
4. Cannot say
5. Its for Linux

**Ans. TRUE**

1. **Oracle Database driver can be used to connect MySQL Database**
2. TRUE
3. FALSE
4. Note Sure
5. All of the above

**Ans. A. TRUE**

1. **Object Cloning is**
2. A way to create exact copy of an object
3. Easy to create a object copy
4. Both 1 and 2
5. None of the above

**Ans: C. Both 1 and 2**

1. **Jenkins is Server based system and runs in a servlet container like Apache Tomcat**
2. YES
3. NO
4. Not sure
5. Cannot say

**Ans: A. YES**

1. **Java is secured because**
2. No Explicit Pointers
3. JVM runs java programs
4. Class loader Loads classes into JVM
5. All of the above

**Ans. All of the above**

1. **The module should be open for extension but closed for modification is**
2. Closed Open Principal
3. Open Closed Principal
4. Not sure
5. Cannot say

**Ans. B. Open closed Principal**

1. **Java is Architecture Neutral because**
2. There are no implementation dependent features
3. Size of primitive data types is fixed
4. Note sure
5. Both 1 and 2

**Ans: A. There are no implementation dependent features**

1. **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Maintains cursor pointing to a row of a table**
2. Callable Statement
3. Statement
4. ResultSet
5. PreparedStatement

**Ans. C. ResultSet**

1. **BDDMockito is used to**
2. Develop a test in BDD Style
3. BDD style has 3 Given, When, Then to mock a behavior and expected response
4. assetThen() checks the operation similar to assertEqual()
5. All of the above

**Ans. B. BDD style has 3 Given, When, Then to mock a behavior and expected response**

1. **Steps to create a Jenkins CI CD pipe line are**
2. Define a Project, Specify Repository and Branch, Configure Build, Configure Post Build Actions
3. Configure Build and Post Build only
4. Post Build Only
5. Cannot say
6. **The JDBC operations are provided in packages**
7. java.sql and java.util
8. javax.sql and java.util
9. java.sql and javax.sql
10. All of the above

**Ans. C. java.sql and javax.sql**

1. **One Java Program with main () method execution needs one JVM instance**
2. Yes
3. No
4. Not sure
5. May Be

**Ans: A. Yes**

1. **When one large interface is split into smaller interfaces will**
2. Allow client to implement only the methods they want
3. This interface segregation principal
4. Not sure
5. Both 1 and 2

**Ans: B. This interface segregation principal**

1. **The verifyMoreInteractions() method is used to**
2. Check if any given mocks have unverified interactions
3. Check if nothing else was invoked on the mocks
4. Optional method
5. All of the above

**Ans: A. Check if any given mocks have unverified interactions.**

1. **When you perform the command git init, what does it do?**
2. Initializes the current directory with a folder called .git
3. Initializes the current directory with a folder called .init
4. Initializes the current directory a folder called .tree
5. Initializes the current directory with a folder called .project

**Ans. A. Initializes the current directory with a folder called .git**

1. **The fields in an interface are implicitly specified as,**
2. final & static both
3. private
4. static
5. final

**Ans: A. final & static both**

1. **Which of the core collection interface represents an implementation that will store unique values in sorted manner?**
2. HashSet
3. Set
4. List
5. None

**Ans: B. Set**

1. **Which of the following options can throw a NullPointerException?**

A. TreeSet<String> s= new TreeSet<String>();

s.add(null);

1. HashMap<String, String> m=new HashMap<String, String>();

m.put(null, null);

1. ArrayList<String> arr=new ArrayList< String >();

arr.add()null;

1. HashSet<String> s=new HashSet<String>();

s.add(null);

**Ans: A. TreeSet<String> s= new TreeSet<String>();**

**s.add(null);**

1. **Which of the following is not an exception in java?**
2. NullPointerException
3. ArrayOutOfException
4. ArithmeticException
5. None

**Ans: B. ArrayOutOfException**

1. **Daemon thread runs in the \_\_\_\_\_\_\_\_\_\_**
2. Background
3. Foreground
4. Daemon Thread is not Thread
5. None

**Ans: A. Background**

1. **All Java classes are derived from**
2. java.util.Name
3. java.lang.Object
4. java.awt.Window
5. None

**Ans: B. java.lang.Object**

1. **State True or False : The finalize() method is guaranteed to run once and only once before the garbage collector deletes an object.**
2. True
3. False

**Ans: A. True**

1. **Which of the following class is synchronized?**
2. HashTable
3. ArrayList
4. TreeSet
5. HashMap

**Ans. A. HashTable**

1. **What is the output following code?**

public class Test {

public static void main(String[] args){

int p=20;

int q=5;

int r=10;

System.out.println((p<q)&&((q++)<r));

System.out.println(q);

}

}

1. False

5

1. True

5

1. False

6

1. True

6

**Ans: A. False**

**5**

1. **How can you retrieve information from a ResultSet?**
2. By invoking the method get(…, String type) on the ResultSet, where type is the database data type
3. By invoking the method get(…, Type type) on the ResultSet, where Type is an object which represents a the database data type
4. By invoking the method getValue(…) and cast the result to desired java type
5. By invoking the special getter methods on the ResultSet like: getString(…), getBoolean(…), getClob(…)

Ans: D. By invoking the special getter methods on the ResultSet like: getString(…), getBoolean(…), getClob(…)

1. **The newly introduced Streams API is available in which java 8 packages?**
2. java.io.streams
3. java.io.stream
4. java.util. Streams
5. java.util.stream

**Ans. D. java.util.stream**

1. **A HashMap allows the existence of:**
2. One null key, multiple null values
3. One null key, one null value
4. Zero null key , multiple null values
5. No null key, no null values

**Ans. A. One null key, multiple null values**

1. **Prepared Statement object in JDBC used to execute…………. quires?**
2. Parameterized
3. Executable
4. Simple
5. High level

**Ans: A. Parameterized**

1. **Entire test classes or individual test methods may be disabled via the annotation \_\_**
2. @Disabled
3. @Skip
4. @Invalid
5. @Exclude

**Ans. A. @Disabled**

1. **Junit is used for what type of software testing for java language?**
2. Functional Testing
3. System Testing
4. Unit Testing
5. Integration Testing

**Ans. C. Unit Testing**

1. **The given statements stand true in respect to \_\_\_\_\_\_\_\_\_**
2. *This repository is managed by maven community*
3. *It is not required to be configured*
4. *It requires internet access to be searched*
5. Local repository
6. Central repository
7. Global repository
8. Both 2 and 3

**Ans: B. Central Repository**

1. **Which of the given Maven phase(s) run any checks to verify the package is valid and meets quality criteria?**
2. *Validate*
3. *Test*
4. *Verify*
5. 1 only
6. 2 only
7. 3 only
8. Both 1 and 2

**Ans. C. 3 only**

1. **In spring xml-based configuration, constructor injection is achieved using\_\_\_\_\_\_\_\_\_ attribute?**
2. constructor-arg
3. constructor
4. const-arg
5. property

**Ans. A. constructor-arg**

1. **Which of the following is correct?**
2. ApplicationContex extends BeanFactory
3. ApplicationContex implements BeanFactory
4. BeanFactory extends ApplicationContex
5. BeanFactory implements ApplicationContex

**Ans: A. ApplicationContex extends BeanFactory**

1. **Which annotation or annotations can be used as meta-annotation to create cutom injection and bean declare annotation**
2. @Autowired
3. @Value
4. @Bean
5. All of the above

**Ans: D. All of the above**

1. **Which of the following is the Spring Boot annotation alternative to Spring’s standard @Configuration annotation?**
2. @EnambleAutoConfiguration
3. @SpringBootConfiguration
4. @ConfigurationProperties
5. @ConfigurationPropertiesScan

**Ans. B. @SpringBootConfiguration**

1. **What is the reason to use Aspect Oriented Programming?**
2. It breaks the program logic into different parts
3. It is used to add the additional concern dynamically
4. It promotes boilerplate code
5. 1&2

**Ans. D. 1&2.**

1. **Identify the property under application.properties that can be used to disable restart feature of DevTools?**
2. spring.devtools.restart
3. spring.devtools.restart.enabled
4. spring.devtools.restart.diasabled
5. None of the above

**Ans. B.spring.devtools.restart.enabled**

1. **Is this correct? The Spring Application class allows an application to be initialized lazily?**
2. No
3. Yes

**Ans. B. Yes**

1. **Beans can be created by which of the following properties?**
2. Scope
3. Property
4. Class
5. It’s own constructor

**Ans. D. It’s own constructor**