**What is Automation?**

Automation is any action which can reduce human efforts.

**What all things can you automate?**

Regression test suite,Smoke / Sanity test suite,Build deployment

* Test data creation, Automating behind the GUI like testing of APIs and methods

**Can you achieve 100% automation?**

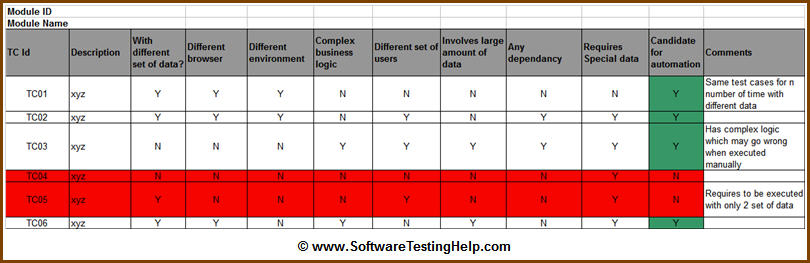
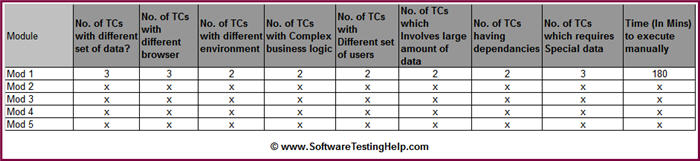
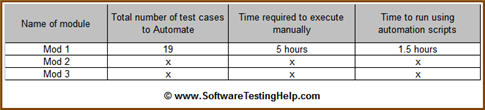
* 100% automation would be difficult to achieve because there would be many edge test cases and some cases which are executed seldom. Automating these cases which are not executed that often will not add value to the automated suite.

**How to Select Correct Test Cases for Automation Testing**

**Step 1:**

Identify the parameters on which you will base your test case as a candidate for automation.As of now I am identifying the below parameters, you can have your own parameters depending on your application.

different set of data, browser, environment,complex business logic,set of users, Involves large amount of data

* Test case has any dependency,Test case requires Special data
* **Step 2:**
* Break each application into modules. For each module, analyze and try to identify the test cases which should be automated based on the parameters. This list will vary for projects to projects and can also be enhanced to suite your needs:
* [](http://cdn.softwaretestinghelp.com/wp-content/qa/uploads/2014/03/Manual-to-automation-testing.jpg)
* Consolidate and group the number of test cases for each module shown below
* [](http://cdn.softwaretestinghelp.com/wp-content/qa/uploads/2014/03/Manual-to-automation-testing-1.jpg)
* **Step 4:**
* Once you have identified all the granular level details, you can present them in the below way. We are now progressing to calculate the ROI.
* [](http://cdn.softwaretestinghelp.com/wp-content/qa/uploads/2014/03/Manual-to-automation-testing-2.jpg)

**\*ROI = (Cumulative saving / Investment through automation)\*100**

**Automation Need:** Every test team is unique and has an exclusive need for automation. We cannot develop a fix standard, but we can tailor a standard which will suite our need.  Because of this reason, automation does require a good support from the management as well as from the development team.

**#2. Automating the complete application:**Automating 100% application is a big task. Not that it is impossible, but it requires proper planning and monitoring and of course; some time. There are lots of permutations and combinations of data, n number of environments with n number of authentication and authorization attributes which needs to be validated and hence requires a strategy to automate.

**#3. Manual Vs Automation mentality:** “**We normally automate which is important and repetitive, but we prefer to test the important functionality manually**”. Confused? Even I am!! But this is a fact. We should have a criteria which will decide which are the **important** test cases. These criteria can be based on multiple factors like complex business logics, areas which are of more interest to clients, risk prone areas etc.

**#4. Deciding on the framework:** [Designing the framework](http://www.softwaretestinghelp.com/qtp-tutorial-17-test-automation-frameworks/) is the most important facet of automation. I believe we should devote relatively more time to develop the framework than to script. Whenever we develop the automation plan, framework designing should be the main focus. PLAN to design the framework. Identify and make a checklist of the items which will form the framework. If the framework is rock solid, scripting and maintaining becomes easy.

**#5. Knowledge of the team:** Whenever we think of automation, we immediately jump to learn the programming language or scripting language. Learning these language will definitely help but more emphasis should be on building and developing logic. Automation should not be the responsibility of some handful resources, rather entire team should contribute towards it. This will help not only to enhance the skills of the resources, but also [keep them motivated](http://www.softwaretestinghelp.com/how-to-keep-motivation-alive-in-software-testers/).

**#6. Reporting:** Every tool has a standard to report the test results. To customize it; is a challenging task. Reporting the test results also requires coordination and maintenance which adds to the cost.

**Trust:** We should trust our automation stuff. We invest man hours to build an automation suite but still we don’t believe in the test results. Efforts should be given to maintain the scripts. Also we should see that the team who is doing the manual testing of the application, should be involved to automate it as they know their application. Most of the time, a third team does the automation so the actual testing team is not aware of the scripts and finally end up with running the tests manually because they feel follow up on the scripts,  add up to their tasks

**Currently I do not have any automation in place in my project, now I want to implement automation, `what would be my steps?**

* First identify which type of testing / test cases you want to automate
* Identify the tool
* Design the framework
* Create the utility files and environment files
* Start scripting
* Identify and work on the reporting
* Allocating time for enhancing and maintaining the scripts.

**How do you decide which tool you have to use?**

Do you have necessary skilled resource to allocate for automation tasks?

**What is your budget?**

**3)** Does the tool satisfy your testing needs? Is it suitable for the project environment and technology you are using? Does it support all tools and objects used in the code? Sometime you may get stuck for small tests due to inabilities of the tool to identify the objects used in the application.

4) Does the tool provide you the free trial version so that you can evaluate it before making a decision? Also does the tool have all features available in trial version?

**5)** Is the current tool version stable? Is the vendor company established with good customer support as well as online help resources and user manual?

How is the tool learning curve? Is the learning time acceptable for your goals?

**7)** Do you want automation tool for only your project needs or you are looking for a common tool for all projects in your company? It would be a good choice if you select a tool that supports most of the coding languages on your projects.

**)** Which testing types does it support? Tool which supports maximum testing types (Unit, functional, regression etc.) is always a better choice. Warning – Don’t go for a tool just because it is supporting all testing types. It’s also important that the tool should be powerful enough to automate your complex requirements.

**9)** Does the tool support easy interface to create and maintain test scripts? Record and playback tool with abilities to edit recorded scripts could be a good solution.

**10)**  Does it provide simple interface yet powerful features to accomplish complex tasks?

**11)** How easy it is to provide input test data for complex or load tests? Tool supporting test data input from various data files such as Excel, XML, text file etc. would be a big relief for the automation the testers.

Does it provide the powerful reporting with graphical interface? Clear and concise reports will always help you to conclude the test results quickly.

**13)**  Does it integrate well with your other testing tools like project planning and [test management tools](http://www.softwaretestinghelp.com/software-test-case-management-tools/)?

Tool vendor refund policy

**15)** Existing customer reviews for the tool

**16)**  Is the vendor providing initial training?

**Tips:** Requirement gathering is by far the most important step for selecting the right tool. Make sure to categorize your requirements in must have, nice to have, and not required feature categories. This will help you to evaluate the tool quickly. Remember you won’t find a tool already available in the market which will support all your automation needs!

* **Once you identify the tool what would be your next steps?**
* Once we finalize the tool, our next step would be to design the framework.

**Q #8) What is a framework?**

* A framework is a set of structure of the entire automation suit. It is also a guideline, if followed can result in a structure which is easy to maintain and enhance. These guidelines inc
* **Once you identify the tool what would be your next steps?**
* Once we finalize the tool, our next step would be to design the framework.

**What are the attributes of a good framework?**

* Modular – The framework should be adaptable to change. Testers should be able to modify the scripts as per the environment or login information change
* Reusable – The commonly used methods or utilities should be written in a common file which is accessible to all the scripts.
* Consistent – The suite should be written in a consistent format by following all the accepted coding practices.
* Independent – The scripts should be written in such a way that they are independent of each other. In case one test fails, it should not holdback remaining test cases (unless it is a login page)
* Logger – It is good to have implemented the logging feature in the framework. This would help in case our scripts run for longer hours (say nightly mode), if the script fails at any point of time, having the log file will help us to detect the location and the type of error.
* Reporting – It is good to have reporting feature automatically embedded into the framework. Once the scripting is done, we can have the results and reports sent via an email.
* Integration – Automation framework should be such that it is easy to integrate it with other application like continuous integration or triggering the automated script as soon as the build is deployed.

**Q #10) Can you do without a framework?**

Frameworks are guidelines and not mandatory rules, so we can do without a framework, but if we create it and follow it, enhancing and maintaining would be easy to implement.

**#11) What are the different types of automation tool you are aware of?**

Open source tool   like Selenium, JMeter

Paid tools like QTP, Load Runner, Ranorex, RFT, and Rational Robot

**What generally is the structure of a framework?**

* A “src” (source) folder having the actual test scripts
* A”lib” (library) folder having all the libraries and common methods
* A “class” folder having all the class file (in case using java)
* A “log” folder having the log file(s)
* A file / folder having all the web element Ids
* A file containing the URL, environment and login information.

**Where you maintain information like URL, login, password?**

* This information should always be maintained in a separate file.

**Q #14) Why do you want to keep this kind of information in separate file and not directly in code?** URL, Login and Password are the kind of fields which are used very often and these changes as per the environment and authorization. In case we hardcode it into our code, we have to change it in every file which has its reference. In case there are say more than 100 files, then it becomes very difficult to change in all the 100 files and hence can lead to errors. So this kind of information is maintained in a separate file so that updating becomes easy.

URL, Login and Password are the kind of fields which are used very often and these changes as per the environment and authorization. In case we hardcode it into our code, we have to change it in every file which has its reference. In case there are say more than 100 files, then it becomes very difficult to change in all the 100 files and hence can lead to errors. So this kind of information is maintained in a separate file so that updating becomes easy.

**Can you tell some good coding practices while automation?**

Some of the good coding practices include:

* Add appropriate comments
* Identify the reusable methods and write it in separate file
* Follow the language specific coding conventions
* Maintain the test data in a separate file
* Run your scripts regularly

**Q #17) Any kind of test which you thing should not be automated?**

* Test which are seldom executed
* Exploratory testing
* Usability testing
* Test which are executed fairly quickly when donEmanually

**Do you think that testing can be done only at the UI level?**

* Today as we are moving to Agile mode, testing is not limited to the UI layer. Early feedback is imperial for any agile project. If we concentrate only on the UI layer, we are actually waiting until the UI is developed and available to test. Rather we can test even before the UI is actually developed. We can directly test the APIs or the methods using tools like Cucumber and [Fitnesse](http://www.softwaretestinghelp.com/getting-started-with-fitnesse-a-collaboration-tool-for-testers-and-developers/).
* In this way we are giving the feedback much early and even are testing before the UI is developed. Following this approach will help us to test only the GUI aspect like small cosmetic changes or some validations on the UI and will help the developers by giving more time to fix the bugs.

**How do you select which automation tool is best suited for you?**

* Scope of the application which we want to automate
* Management overhead like cost and budget
* Time to learn and implement the tool
* Type of support available for the tool.
* Limitation of the tool

**What do you think holds testers back to do automation? Is there a way to overcome it?**

The major hurdle for testers is to learn programming / coding when they want to automate. Since testers do not code, adapting to coding is a bit challenging for testers. We can overcome it by:

* Collaborating with developers when automating
* Considering that automation is the responsibility of the whole team and not only of the testers
* Giving a dedicated time and focus on automation.
* Getting proper management support.

**What are the primary features of good automation tool?**

* Test Environment support and easy to use
* Good debugging facility
* Robust object identification
* Object and Image testing abilities
* Object identification
* Testing of database
* Support multiple frameworks

**What is the scripting standard while performing automation testing?**

While writing the scripts for automation, you must consider following things,

* Uniform naming convention.
* 3 Lines of comments for every 10 lines of code
* Adequate indentation.
* Robust error handling and recovery scenario
* Use of Frameworks wherever possible

**On what basis you can map the success of automation testing?**

* Defect Detection Ratio, Reduction in Labour & other costs
* Automation execution time and time savings to release the product

**12) Can list out some disadvantage of manual testing?**

* Manual Software Testing requires more time and more resources.
* Inaccuracy
* Executing same test case repeatedly is error prone and boring.
* It is impractical to do manual testing on very large projects and time bounded projects.
* **Tell me about QTP**
* QTP (Quick Test Professional) is now known as HP UFT. It is a commercial automation tool and supports a very wide range of test environments Web, Desktop, SAP, Delphi, Net, ActiveX, Flex, Java, Oracle, Mobile, PeopleSoft, PowerBuilder, Siebel, Stingray, Visual Basic amongst others.
* The scripting language is VBScript. The tool gels well with HP ALM (Test Management Tool) and HP LoadRunner (Performance Testing Tool).
* Salient features of QTP include Business Process Testing, keyword driven framework, XML support, robust checkpoints, test results.

**Mention what is the difference between Selenium and Sikuli?**

|  |  |
| --- | --- |
| **Sikuli** | **Selenium** |
| * It provides extensive support to automate flash objects * It has simple API * It uses a visual match to find elements on the screen. So, we can automate anything we see on the screen * It can automate the web as well as windows application | * It cannot automate flash objects like video player, audio player, * It has got complicated API * It does not have visual match * It can automate only web applications |

* **What are the differences between open source tools, vendor tools, and in-house tools?**

Open source tools are free to use frameworks and applications. Engineers build the tool, and have the source code available for free on the internet for other engineers to use. Vendor tools are developed by companies that come with licenses to use, and often cost money. Because they are developed by an outside source, technical support is often available for use. Example vendor tools include WinRunner, SilkTest, Rational Robot, QA Director, QTP, LR, QC, RFT, and RPT. An in-house tool is a tool that a company builds for their own use, rather than purchasing vendor tools or using open source tools.

* **What kind of tests would you automate?**

Load tests, sanity tests, and regression tests are the sorts of tests a quality engineer would typically automate.

* **Does automation testing have any disadvantages?**

Designing the tools and tests to run software through takes a lot of manual, human effort, though there are frameworks and tests ready made for engineers to use. Even with automated testing, human error is still a factor – tools can be buggy, inefficient, costly, and sometimes even technologically limited in what kinds of tests they can run on their own.

* **Does the Selenium IDE have any drawbacks?**

The Selenium IDE lacks conditional statements, logging and reporting functionality, loops, database testing, and it can not handle exceptions or automatically re-run tests that have failed. It also can’t take screenshots. Another downside is that it’s Firefox only. If the Selenium IDE is used in the Firefox browser’s side-bar, then the quality engineer can’t use  it to record any actions undertaken by a user in a separate window.

* **List the elements locators that Selenium comes with.**
* *HTML name ,HTML id, CSS locators, XPath locators*
* **Can you list some technologies that support QTP?**
* *.NET,Delphi,Web,Java (Core),Java  (Advanced),Windows Mobile,Oracle,SAP,WPF*
* *Siebel,PeopleSoft,Flex,Silverlight*
* **What are some test assets and related extensions of QTP?**
* *test batch runner****.mtb***
* *results****.xml***
* *test file****.mts***
* *recovery scenario****.qrs***
* *shared object repository****.tsr***
* *local object repository****.mtr***
* *function library****.qfl***

**What environment does QTP run in? -** Windows only.

* **List the five essential types of test steps.**

1. *Test object*
2. *Functions*
3. *Utility*
4. *Comment*
5. *Programming logic*
6. **Explain how  QTP identifies objects.**

For each object class that QTP identifies, it has a set of properties stored. These include mandatory properties, which is essentially a description of any given object, which checks if it’s a child or parent object as well. Next are the assistive properties, which QTP resorts  to only if the mandatory properties are insufficient at identifying the object. If all else fails, QTP resorts, finally, to its ordinal identifier.

**What are the prerequisites to run selenium webdriver?**

Ans- JDK, Eclipse, WebDriver(selenium standalone jar file), browser, application to be tested.

**What does SQL stand for?**  
**Ans.** SQL stands for [Structured Query Language](http://en.wikipedia.org/wiki/SQL).

**Q#2. How to select all records from the table?**  
**Ans.** To select all the records from the table we need to use following syntax:

Select \* from table\_name;

**Q#3. Define join and name different type of joins?**  
**Ans.** Join keyword is used to fetch data from related two or more tables. It returns rows where there is at least one match in both the tables included in join. [Read more here](http://www.w3schools.com/sql/sql_join.asp).  
Type of joins are-

1. Right Join
2. Outer Join
3. Full Join
4. Cross Join
5. Self Join.

**Q#4. What is the syntax to add record to a table?**  
**Ans.** To add record in a table INSERT syntax is used.

Ex: INSERT into table\_name VALUES (value1, value2..);

**Q#5. How do you add a column to a table?**  
**Ans.** To add another column in the table following command has been used.

ALTER TABLE table\_name ADD (column\_name);

**Q#6. Define SQL Delete statement.**  
**Ans.** Delete is used to delete a row or rows from a table based on the specified condition.  
Basic syntax is as follows:

DELETE FROM table\_name

WHERE <Condition>

**Q#7. Define COMMIT ?**  
**Ans.** COMMIT saves all changes made by DML statements.

**Q#8. What is a primary key?**  
**Ans.** A Primary key is column whose values uniquely identify every row in a table. Primary key values can never be reused.

**Q#9. What are foreign keys?**  
**Ans.** When a one table’s primary key field is added to related tables in order to create the common field which relates the two tables, it called a foreign key in other tables.  
Foreign Key constraints enforce referential integrity.

**Q#10. What is CHECK Constraint?**  
**Ans.** A CHECK constraint is used to limit the values or type of data that can be stored in a column. They are used to enforce domain integrity.

**Q#11. Is it possible for a table to have more than one foreign key?**  
**Ans.** Yes, a table can have many foreign keys and only one primary key.

**Q#12. What are the possible values for BOOLEAN data field.**  
**Ans.** For a BOOLEAN data field two values are possible: -1(true) and 0(false).

**Q#13. What is a stored procedure?**  
**Ans.** A stored procedure is a set of SQL queries which can take input and send back output.

**Q#14. What is identity in SQL?**  
**Ans.** An identity column in the SQL automatically generates numeric values. We can defined a start and increment value of identity column.

**Q#15. What is Normalization?**  
**Ans.** The process of table design to minimize the data redundancy is called normalization. We need to divide a database into two or more table and define relationships between them.

**Q#16. What is Trigger?**  
**Ans.** Trigger allows us to execute a batch of SQL code when a table event occurs (Insert, update or delete command executed against a specific table)

**Q#17. How to select random rows from a table?**  
**Ans.** Using SAMPLE clause we can select random rows.

Example:  
SELECT \* FROM table\_name SAMPLE(10);

**Q#18. Which TCP/IP port does SQL Server run?**  
**Ans.** By default SQL Server runs on port 1433.

**Q#19. Write a SQL SELECT query that only returns each name only once from a table?**  
**Ans.** To get the each name only once, we need to use the DISTINCT keyword.

SELECT DISTINCT name FROM table\_name;

**Q#20. Explain DML and DDL?**  
**Ans.** DML stands for Data Manipulation Language. INSERT, UPDATE and DELETE  are DML statements.

DDL stands for Data Definition Language. CREATE ,ALTER, DROP, RENAME are DDL statements.

**Q#21. Can we rename a column in the output of SQL query?**  
**Ans.** Yes using the following syntax we can do this.

SELECT column\_name AS new\_name FROM table\_name;

**Q#22. Give the order of SQL SELECT ?**  
**Ans.** Order of SQL SELECT clauses is: SELECT, FROM, WHERE, GROUP BY, HAVING, ORDER BY. Only the SELECT and FROM clause are mandatory.

**Q#23. Suppose a Student column has two columns, Name and Marks. How to get name and marks of top three students.**  
**Ans.** SELECT Name, Marks FROM Student s1 where 3 <= (SELECT COUNT(\*) FROM Students s2 WHERE s1.marks = s2.marks)

**Q#24. What is SQL comments?**  
**Ans.** SQL comments can be put by two consecutive hyphens (–).

**Q#25. Difference between TRUNCATE, DELETE and DROP commands?**  
**Ans.** DELETE removes some or all rows from a table based on the condition. It can be rolled back.

TRUNCATE removes ALL rows from a table by de-allocating the memory pages. The operation cannot be rolled back

DROP command removes a table from the database completely.

**Q#26. What are the properties of a transaction?**  
**Ans.** Generally these properties are referred as ACID properties. They are:

1. Atomicity
2. Consistency
3. Isolation
4. Durability.

**Q#27. What do you mean by ROWID ?**  
**Ans.** It’s a 18 character long pseudo column attached with each row of a table.

**Q#28. Define UNION, MINUS, UNION ALL, INTERSECT ?**  
**Ans.** MINUS – returns all distinct rows selected by the first query but not by the second.

UNION – returns all distinct rows selected by either query

UNION ALL – returns all rows selected by either query, including all duplicates.

INTERSECT – returns all distinct rows selected by both queries.

**Q#29. What is a transaction?**  
**Ans.** A transaction is a sequence of code that runs against a database. It takes database from one consistent state to another.

**Q#30. What is difference between UNIQUE and PRIMARY KEY constraints?**  
**Ans.** A table can have only one PRIMARY KEY whereas there can be any number of UNIQUE keys.

Primary key cannot contain Null values whereas Unique key can contain Null values.

**Q#31. What is a composite primary key?**  
**Ans.** Primary key created on more than one column is called composite primary key.

**Q#32. What is an Index ?**  
**Ans.** An Index is an special structure associated with a table speed up the performance of queries. Index can be created on one or more columns of a table.

**Q#33. What is the Subquery ?**  
**Ans.** A Subquery is sub set of select statements whose return values are used in filtering conditions of the main query.

**Q#34. What do you mean by query optimization?**  
**Ans.** Query optimization is a process in which database system compares different query strategies and select the query with the least cost.

**Q#35. What is Collation?**  
**Ans.** Set of rules that defines how data is stored, how case sensitivity and Kana character can be treated etc.

**Q#36. What is Referential Integrity?**  
**Ans.** Set of rules that restrict the values of one or more columns of the tables based on the values of primary key or unique key of the referenced table.

**Q#37. What is Case Function?**  
**Ans.** Case facilitates if-then-else type of logic in SQL. It evaluates a list of conditions and returns one of multiple possible result expressions.

**Q#38. Define a temp table?**  
**Ans.** A temp table is a temporary storage structure to store the data temporarily.

**Q#39. How we can avoid duplicating records in a query?**  
**Ans.** By using DISTINCT keyword duplicating records in a query can be avoided.

**Q#40. Explain the difference between Rename and Alias?**  
**Ans.** Rename is a permanent name given to a table or column whereas Alias is a temporary name given to a table or column.

**Q#41. What is a View?**  
**Ans.** A view is a virtual table which contains data from one or more tables. Views restrict data access of table by selecting only required values and make complex queries easy.

**Q#42. What are the advantages of Views?**  
**Ans.** Advantages of Views:

1. Views restrict access to the data because the view can display selective columns from the table.
2. Views can be used to make simple queries to retrieve the results of complicated queries. For example, views can be used to query information from multiple tables without the user knowing.

**Q#43. List the various privileges that a user can grant to another user?**  
**Ans.**   SELECT, CONNECT, RESOURCES.

**Q#44. What is schema?**  
**Ans.** A schema is collection of database objects of a User.

**Q#45. What is Table ?**  
**Ans.** A table is the basic unit of data storage in the database management system. Table data is stored in rows and columns.

**Q#46. Do View contain Data?**  
**Ans.** No, Views are virtual structure.

**Q#47. Can a View based on another View?**  
**Ans.** Yes, A View is based on another View.

**Q#48. What is difference between Having clause and Where clause?**  
**Ans.** Both specify a search condition but Having clause is used only with the SELECT statement and typically used with GROUP BY clause.  
If GROUP BY clause is not used then Having behaves like WHERE clause only.

**Q#49. What is difference between Local and Global temporary table?**  
**Ans.** If defined in inside a compound statement a local temporary table exists only for the duration of that statement but a global temporary table exists permanently in the db but its rows disappears when the connection is closed.

**Q#50. What is CTE?**  
**Ans.** A CTE or common table expression is an expression which contains temporary result set which is defined in a SQL statement.

**What are the different recording modes and how do they work**

* ***Normal Recording mode*** – The default recording method is always the normal mode. This method uses the model of Test Objects and Runtime objects to learn and act on the AUT.
* ***Analog recording mode*** – records the exact mouse and key strokes that the user performs in relation to either the screen or AUT window.  The steps that are recorded using this method cannot be edited.  
  The way this usually gets represented in a code is:  
  *Window/app.RunAnalog “Track1”*  
  One scenario in which this type of recording can be used is when we are trying  
  to capture a signature.
* ***Low Level recording mode*** – This mode records the co-ordinates in the application where the operation is performed, irrespective of whether QTP recognizes the specific Object or Operatio
* **How does QTP identify an object?**
* ***Mandatory properties:*** This is the list of properties for a certain class that QTP always stores. We could say that this is the object description. It also checks this in conjunction with the parent object to see if the description is sufficient to identify the object uniquely.
* ***Assistive properties:*** In case the description of mandatory properties is insufficient to identify the Object a set of non-mandatory properties will be added to the description one after the other until there is enough data to identify the object.
* ***Ordinal Identifier:*** If the assistive properties also do not result in unique identification of an object a special ordinal identifier is added by QTP, such as the object’s location on the page or in the source code.

**What are the two types of repositories available, explain them?**  
Local and shared repository – these are the two kinds of available repositories.  
Every action by default has a local repository of its own and has all the objects that are used within it.  
Alternately, the tester can have a common repository for multiple actions so that all of them can share the objects that it contains. The common repository is called a shared OR.

**) What is Smart identification?**  
If the recorded description does not enable QTP to identify a specific object then QTP uses “Smart identification” mechanism.  It uses the following additional properties to identify the object  
**a)** Base Filter Properties (primary) – The set of properties that cannot be changed without changing the object type  
**b)** Optional Filter Properties (secondary) – additional properties that help identify the object uniquely.

**14) What is Object Spy?**  
Object Spy is an extremely helpful tool that QTP has to view the properties and operations of an object in the AUT. It shows all the properties of the object and the corresponding values. It also shows the object hierarchy. It also has a provision that lets the users add a certain object to the OR.

**15) What is an object repository?**  
OR is like a warehouse where all the objects in a test are stored. OR has the list of Objects that QTP learned during the record process and the class to which they belong. It stores the set of properties that uniquely identifies the Object (description) and also names the object for the sake of identification in our test, based on its most prominent feature.

**What to test – different components**

**1) Transactions:**

When testing transactions it is important to make sure that they satisfy the ACID properties.

These are the statements commonly used:

* BEGIN TRANSACTION TRANSACTION#
* END TRANSACTION TRANSACTION#

The Rollback statement ensures that the database remains in a consistent state.

* ROLLBACK TRANSACTION#

After these statements are executed, use a Select to make sure the changes have been reflected.

* SELECT \* FROM TABLENAME <tables which involve the transactions>

**2) Database schema:**

A database schema is nothing more than a formal definition of the how the data is going to be organized inside a DB. To test it:

* Identify the requirements based on which the database operates. Sample requirements:
  + Primary keys to be created before any other fields are created.
  + Foreign keys should be completely indexed for easy retrieval and searching.
  + Field names starting or ending with certain characters.
  + Fields with a constraint that certain values can or cannot be inserted.
* Use one of the following methods according to the relevance:
  + SQL Query *DESC<table name>* to validate the schema.
  + Regular expressions for validating the names of the individual fields and their values
  + Tools like SchemaCrawler

**3) Trigger:**

When a certain event takes places on a certain table, a piece of code (a trigger) can be auto-instructed to be executed.

**For example**, a new student joined a school. The student is taking 2 classes: math and science. The student is added to the “student table”.  A trigger could add the student to the corresponding subject tables once he is added to the student table.

The common method to test is to execute the SQL query embedded in the trigger independently first and record the result. Follow this up with executing the trigger as a whole. Compare the results.

These are tested during both the black box and white box testing phases.

* **White box testing**:  Stubs and drivers are used to insert or update or delete data that would result in the trigger being invoked. The basic idea is to just test the DB alone even before the integration with the front end (UI) is made.
* **Black box testing**:

**a)**Since the UI and DB integration is now available; we can insert/delete/update data from the front end in a way that the trigger gets invoked. Following that, Select statements can be used to retrieve the DB data to see if the trigger was successful in performing the intended operation.

**b)**The second way to test this is to directly load the data that would invoke the trigger and see if it works as intended.

**4) Stored Procedures:**

Stored procedures are more or less similar to user defined functions. These can be invoked by Call Procedure/Execute Procedure statements and the output is usually in the form of result sets.These are stored in the RDBMS and are available for applications.

These are also tested during:

* **White box testing:** Stubs are used to invoke the stored procedures and then the results are validated against the expected values.
* **Black box testing:** Perform an operation from the front end (UI) of the application and check for the execution of the stored procedure and its results.

**5) Field constraints – Default value, unique value and foreign key:**

* Perform a front end operation which exercises the database object condition
* Validate the results with a SQL Query.

Checking the default value for a certain field is quite simple. It is part of business rule validation. You can do it manually or you can use tools like QTP. Manually, you can perform an action that will add a value other than the default value into the field from the front end and see if it results in an error.

The result of the above code is True if the default value exists or False if it doesn’t.

Checking the unique value can be done exactly the way we did for the default values. Try entering values from the UI that will violate this rule and see if an error is displayed.

**What is Database testing?**  
**Ans:** Database testing/Back-end testing is a process of checking an impact of back-end database operation on front-end web/desktop application.  
Data base testing is divided into following different categories,

* Data validity testing: While doing this testing, testers should be having good knowledge of SQL queries.
* Data Integrity testing: While doing this testing, testers/developers should know referential integrity and different constraints.
* Database Performance testing: While doing this testing, testers/developers should be good in designing the structure of table.
* Testing of Procedure, triggers and functions: While doing this testing, testers/developers should be having perfect understanding of testing procedure, triggers and functions.

**Q#2: Why database testing is important?**  
**Ans:** Although, we all know database is not as like small table, it is a big container of many tables and full of data, delivers data at a time to many web/desktop applications. Database testing process ensures that the correct and unique data (without bug) delivers to the correct location. These bugs may cause some serious issues like; dead-locking, data corruption, poor performance, inconsistency, etc….

**Q#3: In the Database Testing process, what do we usually check?**  
**Ans:**The field size validation, Check constraints,Indexes are done or not (for performance related issues)

Stored procedures, The field size defined in the application is matching with that in the db.

**Q#4: Can you test database manually, if yes; then how, explain with example?**  
**Ans:** It needs observing operation; observe the operations of front-end and its effect on the back-end database. While adding a record (XYZ) from the front-end to the back-end database, manually check the record has been affected on the back-end database or not. Similarly, it will work for delete, update,… etc.  
For Example: To keep record of students who are giving exam; enter the students detail from front-end system and manually check the effect of this addition on back-end database.

**Q#5: How to test database procedures and triggers?**  
**Ans:**The process of testing database procedures and triggers requires the knowledge of input parameters, output parameters, and EXEC statement. EXEC statement is helpful in running the procedure and observing the behavior of the tables.  
Let’s see, how to test database procedures and triggers,

* First, open the database project that will be displayed in the solution explorer
* Once the desired project is opened, go to the View menu, click on the database schema, and then open the project folder from schema View menu
* Right click on the object that has to be tested, and then click on the Create Unit Tests. A dialog box “Create Unit Tests” opens and displays all the folders and objects of the database project and the check box for the selected object is checked.
* Next, create a new language test project
* Choose to either insert the unit test in an existing test class or create a new test class and click on OK
* To configure the project, the Project configuration dialog box appears to configure the test project settings. The process requires database connection and data generation plan. Database connection is for executing test result.
* At last, configure the project and click on OK.

**Q#6: What do you mean by data-driven test?**  
**Ans:**Data-driven testing is a term used in the testing process whereas test scripts while execution read test data and/or output values from data files (Data pools, Excel files, ADO objects, CSV files, ODBC sources) instead of reading the same hard-coded values each time. The test helps testers in checking the efficiency of the application in handling various inputs.

**Q#7: What is the database trigger, how to verify the trigger is fired or not and can you invoke trigger on demand?**  
**Ans:**Database trigger: A trigger is basically a stored procedure used to maintain the integrity of the data present in the database, executes automatically to response to a certain event of a table/view in a database.  
To verify, the trigger is fired or not; use the query of common audit log will display the trigger of the data table.  
Triggers can’t be invoked on demand, it invoked when table displays an action (insert, delete & update) defined on that particular table.

**Q#8: After entering the data from the front-end application interface, how do you test whether a database in updated or not?**  
**Ans:**It is totally depends on the application interface that you are using. These are some following ways,

* You can verify only from the front-end when application interface shows view functionality of the data you enter. Mainly, Black box test engineers do this functionality verification test in this way.
* If application interface doesn’t provide view functionality of the data you enter, then you can check for database update by using relevant SQL/Oracle query.
* Checkpoint function of WinRunner/QTP can also be used to check for database update.

**Q#9: How to test the Stored Procedures?**  
**Ans:**Test engineer needs to follow some steps to test the [Stored Procedures](https://technet.microsoft.com/en-us/library/aa174792%28v=sql.80%29.aspx),

* First the test engineer should understand the requirement and also the purpose of a particular Stored Procedure.
* Next verify whether all the indexes, joins, updates, deletions are precise in comparison with the tables mentioned in the Stored Procedure and also make sure that the Stored Procedure is in common standard format, like; comments, updated by, etc.
* Then, for different sets of input parameters, verify the procedure calling name, calling parameters, and expected reponses.
* Manually, run the procedure with database client programs like; TOAD, mysql, or Query Analyzer.
* To verify results against expected values; re-run the procedure by giving different parameters.
* Finally, automate the tests with QTP.

**Q#10: What are joins and mention different types of joins?**  
**Ans:**Join is used to combine two or more than two table and displays as a single set of data.  
In SQL, the [types of joins](http://www.w3schools.com/sql/sql_join.asp) are, Inner Join,Outer Join,Left Join, Right Join

In ORACLE/PLSQL, there are four different types of joins, Inner Join, Left-Outer Join,Right-Outer Join,Full-Outer Join

**Q#11: What do you mean by index and explain different types of indexes?**  
**Ans:**Index is a database object, created on a column of the table to find data more quickly and efficiently (or we can say; it is used to speed up the searches/queries).

B-Tree index,Bitmap index,Clustered index,Covering index,Non-unique index,Unique index

**Q#12:** **What do you mean by DBMS?**  
**Ans:**A database is a structured set of data and a database management system (DBMS) is a computer software applications that create communication network between users, others applications and database to store, modify, and extract information from a database. Famous DBMSs are My SQL, Postgre SQL, Microsoft SQL Server, Oracle, SAP and IBM DB2.

**Q#13: What do you mean by DML?**  
**Ans:**A data manipulation language (DML) is a member of computer programming language facilitates us to select, insert, delete and update data in a database. DML is basically involved in Structured Query Language (SQL), facilitates users to retrieve and use data in a relational database.

**Q#14: What do you mean by DCL commands and explain the types of commands used by DCL?**  
**Ans:**Data Control Language (DCL) is a component of SQL, controls the use of data stored in a database. There are two types of DCL Commands,  
Grant = The command facilitates users to access privilege to database  
Revoke = Stop permitting users to access the database

**Q#15:** **What do you mean by white box testing and black box testing?**  
**Ans:**Black-box testing is a software testing technique to test the functional behavior of an application without getting into its internal structures of an application. Core application’s code/internal structure and programming knowledge is not required for black-box testing, but core knowledge of application functionality is required to test the application.

White Box Testing (WBT) is also known as Code-Based Testing or Structural Testing to test the internal structure of an software application. In case of white-box testing, tester should be having good understanding of internal structure of an application as well as good knowledge of programming skills to design test cases and internal structure of an application.

**Q#16: Does QTP support SQL queries, how to use**[**SQL queries in QTP**](http://www.softwaretestinghelp.com/qtp-tutorial-25-what-is-descriptive-programming/)**?**  
**Ans:**Although, QTP doesn’t carry any built-in function for database connectivity, tester uses VBScript language to connect and interact with databases using ADODB objects.  
ADODB objects are divided into four different properties/methods will help in successfully creating database connection,

------------

* Connection: Useful in forming a connection with a Database
* Command: Useful in executing a SQL command(Queries/Stored Procedures)
* Fields: Useful in retrieving a particular column from a record set after executing a query/stored Procedure.
* Recordset: Used to retrieve data from a database

**Q#17: Which SQL statements can be used in Database Testing?**  
**Ans:** All DDL, DML, DCL, and TCL statements of SQL can be used in database testing,

**DDL (Data Definition Language):** All statements related to this can be used in defining the database structure or schema.

**Statements:** CREATE, ALTER, DROP, TRUNCATE, COMMENT, and RENAME

**DML (Data Manipulation Language):** All statements related to this can be used for managing data within schema objects.

**Statements:** SELECT, INSERT, UPDATE, DELETE, MERGE, CALL, EXPLAIN PLAN, and LOCK TABLE.

**DCL (Data Control Language):** GRANT Statement and REVOKE Statement.

**TCL (Data Manipulation Language):** All statements related to this can be used to manage the changes made by DML statements. Using TCL, statements can be grouped together into logical transactions.

**Statements:** COMMIT, SAVEPOINT, ROLLBACK, and SET TRANSACTION.

**Q#18: In database testing, how to do the data load testing?**  
**Ans:**For data load testing,

* You need to have knowledge about source database (data tables, columns, associated constraints, data types) and destination database (data tables, columns, data types, and constraints).
* You need to check the compatibility between source database and destination database via the DTS package.
* You need to Open corresponding DTS package in SQL Enterprise Manager and run the DTS package (If you are using SQL Server).
* Then you need to compare the column’s data of Source and the Target.
* You have to check the number to rows of Source and Target.
* After updating data in the Source database, you need to check whether the changes have been reflected on destination database or not.
* You need to check about junk character and NULLs.
* Database load testing, check the volume of the database server to process the queries, and also check the response time of database server and client.

**Q#19: Is a “A fast database retrieval rate” a testable requirement?**  
**Ans:**No. I don’t think so because the requirement appears to be unclear. The SRS should clearly display the performance or transaction requirements, i.e. It should mention like; “Database retrieval rate of 5 microseconds”.

**Q#20: Without involving database checkpoints, how you can test a SQL Query in QTP?**  
**Ans:**The given below script procedure help us in connecting to the database where we can test both; the database and queries.  
1) The script procedure for database connection,  
db\_connect(“query1”,DRIVER={driver name};SERVER=server\_name;

UID=uidname;PWD=password;DBQ=database\_name “);

2) The script procedure to execute the query,  
db\_excecute\_query(“query1″,”write query u want to execute”);

-Condition to be mentioned-

3) The script procedure to disconnect the connection with database,  
db\_disconnect(“query”);

**Q#21: In database testing, what all things are required for writing good test cases?**  
**Ans:**Knowledge of following things is required before writing the database test cases,

* At first, understand the application completely and functional requirement of the application.
* Second, check-out other entities that have been used in an application; back-end database tables, joins between the tables, cursors (if any), triggers (if any), stored procedures (if any), input parameter and output parameters for developing that requirement.
* After collection all necessary information, write down the test case with different input values for examining all the resources.
* Writing test cases for back end testing is opposite to functional testing, one should use the white box testing technique.

**Q#22: What is retesting & how it is different from**[**data driven testing**](http://www.softwaretestinghelp.com/test-automation-frameworks-selenium-tutorial-20/)**?**  
**Ans:**After execution of the test in terms of finding the defect that has been already detected and fixed. Re-execution of the same test with different input values to confirm the original defect has been successfully removed is called Re-testing or confirmation Testing.  
Retesting is also called Data Driven Testing, but the difference between both is,  
Retesting:- It is a manual testing process whereas application testing done with entire new set of data.

DataDriven Testing(DDT):-It is an Automation testing process where application will be tested with multiple test data.It is simple and easy than retesting where tester just sit in front of system and enter different new input valuesmanually from front-end interface, it is really boring technique.

**Q#23: What are the types of data driven testing?**  
**Ans:**There are four types of data driven testing,

* Dynamic test data submission through keyboard (key driven test)
* Data Driven Tests via flat files ( .txt, .doc)
* Data Driven Tests via front-end objects
* Data Driven Tests via excel sheet

Dynamic test data submission through keyboard (key driven test): In some cases, testers retest a particular application with different input values to validate the calculation via dynamic submission. So, to submit the input value, testers can use the following function in TSL; script– create\_input\_dialog (“label”);

Data Driven Tests via flat files (.txt,.doc):In some cases, testers do the re-testing by considering the data of the flat file. Testers gather those flat files from old databases/customers.

Data Driven Tests via front-end objects: In some cases, testers create automation scripts by considering the front-end objects values, such as; (a) list (b) menu (c) table (d) data window (e) ocx etc.  
Data Driven Tests via excel sheet: In some cases, testers take the help of this data driven test to run the script for multiple inputs. These multiple inputs reside in columns of an excel sheet. We have to collect this test data from the backend tables.

**Q#24: How to write a query to get the second largest value from a given column of a table?**  
**Ans:**This is a query to get the second largest value from a given column of a table,  
SELECT MAX(COLUMN\_NAME) FROM TABLE\_NAME  
WHERE COLUMN\_NAME < (SELECT MAX(COLUMN\_NAME) FROM TABLE\_NAME)

For Example:-  
To get the second largest marks from “Marks” column of a “Students” table

Select Max(Marks) from Students Where Marks< (Select Max(Marks) from students)

**Q#25: How to write a query to get 10thhighest salary from an employee table?**  
**Ans:**This is a query to get nth highest salary from an employee table,  
SELECT \* FROM Employee Emp1   
WHERE ( n ) = (SELECT COUNT( DISTINCT ( Emp2.Employee\_Salary ) )  
FROM Employee Emp2  
WHERE Emp2.Employee\_Salary >= Emp1.Employee\_Salary)

Here, you can replace the n with any number. For example, if you want to get10th highest salary, then replace n with 10.  
SELECT \* FROM Employee Emp1 WHERE (10) = (  
SELECT COUNT( DISTINCT ( Emp2.Employee\_Salary ) )  
FROM Employee Emp2 WHERE Emp2.Employee\_Salary >= Emp1.Employee\_Salary)

**Q#26: How to write test cases from requirements, do the requirements signify the exact functionality of AUT?**  
**Ans:**Yes, the requirements should signify the exact functionality of AUT.  
To do so,  
First examine the requirement to understand overall functionality.  
Next, decide which test design methods are suitable to write test case – Black Box test design methods like; Equivalence Partitioning, Boundary Value Analysis, Error Guessing and Cause Effect Graphing.  
Once you decide which test design method is suitable, write your test cases in requirement analysis and design phase. In this way you will be ensure that all the requirements are testable.

**Q#27: What is the test scenario to test a database migrated from one SQL Server to another?**  
**Ans:**First of all, we need to check what all enhancements and changes happened to the SQL Server where we are planning to migrate.  
Next, design the test case according to the following consideration,

* Data type that has been used.
* Length of the data field of SQL Server (Server into which we are migrating the data) should be same as the SQL Server from where we are taking out the data.
* Each and every task should be organized correctly.

**Q#28: What does the RDBMS stands for and what are the major RDBMS used by SQL?**  
**Ans:**RDBMS stands for Relational Database Management Systems. Major RDBMS that are involved with SQL are Sybase, Oracle, Access, Ingres, Microsoft SQL server etc.

**Q#29: What is performance testing and the bottlenecks of it?**  
**Ans:**Performance testing is a software testing technique to determine that how a system performance in terms of speed, sensitivity and stability under a heavy workload.  
The performance testing requires expensive tools and well-trained and experienced testers for operation.

**Q#30:** **What is CMMI and describe different levels of CMM?**  
**Ans:**CMMI stands for Capability Maturity Model Integration is a process development training and evaluation model of various business processes in software engineering.  
Five CMM Maturity levels for an organization are,  
**1) Initial:** At this level, organizations don’t have any verified technique and environment, so that; usually organizations budget and schedule time go over.

**2) Repeatable:** At this level, organizations have basic techniques and guidelines to limit the cost and schedule time, capable in repeating the same in the next similar project.

**3) Defined:** At this level, all techniques are well organized and explained clear and standard form.

**4) Managed:** At this level, organizations are much more developed than Defined level. Here, techniques are contacted using statistical and other quantitive technique.

**5) Optimizing:** Organizations determinedly attempt to develop performance via modern technical progress.

**Q#31: What is the meaning of Record in a database?**  
**Ans:** In database, a record is the set of values/fields of a specific entity. For example; Office\_Employee Record, Book Record etc

**1. What is OOPS?**

OOPS is abbreviated as Object Oriented Programming system in which programs are considered as a collection of objects. Each object is nothing but an instance of a class.

**2. Write basic concepts of OOPS?**

1. Abstraction.
2. Encapsulation.
3. Inheritance.
4. Polymorphism.

**3. What is a class?**

A class is simply a representation of a type of object. It is the blueprint/ plan/ template that describe the details of an object.

**4. What is an object?**

Object is termed as an instance of a class, and it has its own state, behavior and identity.

**5. What is Encapsulation?**

Encapsulation is an attribute of an object, and it contains all data which is hidden. That hidden data can be restricted to the members of that class.

Levels are Public,Protected, Private, Internal and Protected Internal.

**6. What is Polymorphism?**

Polymorphism is nothing butassigning behavior or value in a subclass to something that was already declared in the main class. Simply, polymorphism takes more than one form.

**7. What is Inheritance?**

Inheritance is a concept where one class shares the structure and behavior defined in another class. Ifinheritance applied on one class is called Single Inheritance, and if it depends on multiple classes, then it is called multiple Inheritance.

**8. What are manipulators?**

Manipulators are the functions which can be used in conjunction with the insertion (<<) and extraction (>>) operators on an object. Examples are endl and setw.

**9. Define a constructor?**

Constructor is a method used to initialize the state of an object, and it gets invoked at the time of object creation. Rules forconstructor are:.

Constructor Name should be same asclass name.

* Constructor must have no return type.

**10. Define Destructor?**

Destructor is a method which is automatically called when the object ismade ofscope or destroyed. Destructor name is also same asclass name but with the tilde symbol before the name.

**11. What is Inline function?**

Inline function is a technique used by the compilers and instructs to insert complete body of the function wherever that function is used in the program source code.

**12. What is avirtual function?**

Virtual function is a member function ofclass and its functionality can be overridden in its derived class. This function can be implemented by using a keyword called virtual, and it can be given during function declaration.

Virtual function can be achieved in C++, and it can be achieved in C Languageby using function pointers or pointers to function.

**13. What isfriend function?**

Friend function is a friend of a class that is allowed to access to Public, private or protected data in that same class. If the function is defined outside the class cannot access such information.

Friend can be declared anywhere in the class declaration, and it cannot be affected by access control keywords like private, public or protected.

**14. What is function overloading?**

Function overloading is defined as a normal function, but it has the ability to perform different tasks. It allowscreation of several methods with the same name which differ from each other by type of input and output of the function.

Example

void add(int& a, int& b);

void add(double& a, double& b);

void add(struct bob& a, struct bob& b);

**15. What is operator overloading?**

Operator overloading is a function where different operators are applied and depends on the arguments. Operator,-,\* can be used to pass through the function , and it has their own precedence to execute.

Example:



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | class complex {  double real,  imag; public: complex(double r, double i) : real(r),  imag(i) {} complex operator+(complex a, complex b);  complex operator\*(complex a, complex b);  complex& operator=(complex a, complex b);  } |

 a=1.2, b=6

**16. What is an abstract class?**

An abstract class is a class which cannot be instantiated. Creation of an object is not possible with abstract class , but it can be inherited. An abstract class can contain only Abstract method. Java allows only abstract method in abstract class while for other language it allows non-abstract method as well.

**17. What is a ternary operator?**

Ternary operator is said to be an operator which takes three arguments. Arguments and results are of different data types , and it is depends on the function. Ternary operator is also called asconditional operator.

**18. What is the use of finalize method?**

Finalize method helps to perform cleanup operations on the resources which are not currently used. Finalize method is protected , and it is accessible only through this class or by a derived class.

**19. What are different types of arguments?**

A parameter is a variable used during the declaration of the function or subroutine and arguments are passed to the function , and it should match with the parameter defined. There are two types of Arguments.

Call by Value – Value passed will get modified only inside the function , and it returns the same value whatever it is passed it into the function.

Call by Reference – Value passed will get modified in both inside and outside the functions and it returns the same or different value.

**20. What is super keyword?**

Super keyword is used to invoke overridden method which overrides one of its superclass methods. This keyword allows to access overridden methods and also to access hidden members of the superclass.

It also forwards a call from a constructor to a constructor in the superclass.

**21. What is method overriding?**

Method overriding is a feature that allows sub class to provide implementation of a method that is already defined in the main class. This will overrides the implementation in the superclass by providing the same method name, same parameter and same return type.

**22. What is an interface?**

An interface is a collection of abstract method. If the class implements an inheritance, and then thereby inherits all the abstract methods of an interface.

**23.   What is exception handling?**

Exception is an event that occurs during the execution of a program. Exceptions can be of any type – Run time exception, Error exceptions. Those exceptions are handled properly through exception handling mechanism like try, catch and throw keywords.

**24. What are tokens?**

Token is recognized by a compiler and it cannot be broken down into component elements. Keywords, identifiers, constants, string literals and operators are examples of tokens.

Even punctuation characters are also considered as tokens – Brackets, Commas, Braces and Parentheses.

**25. Difference between overloading and overriding?**

Overloading is static binding whereas Overriding is dynamic binding. Overloading is nothing but the same method with different arguments , and it may or may not return the same value in the same class itself.

Overriding is the same method names with same arguments and return types associates with the class and its child class.

**26. Difference between class and an object?**

An object is an instance of a class. Objects hold any information , but classes don’t have any information. Definition of properties and functions can be done at class and can be used by the object.

Class can have sub-classes, and an object doesn’t have sub-objects.

**27. What is an abstraction?**

Abstraction is a good feature of OOPS , and it shows only the necessary details to the client of an object. Means, it shows only necessary details for an object, not the inner details of an object. Example – When you want to switch On television, it not necessary to show all the functions of TV. Whatever is required to switch on TV will be showed by using abstract class.

**28. What are access modifiers?**

Access modifiers determine the scope of the method or variables that can be accessed from other various objects or classes. There are 5 types of access modifiers , and they are as follows:.

* Private.,Protected.,Public.,Friend.,Protected Friend.

**29. What is sealed modifiers?**

Sealed modifiers are the access modifiers where it cannot be inherited by the methods. Sealed modifiers can also be applied to properties, events and methods. This modifier cannot be applied to static members.

**30. How can we call the base method without creating an instance?**

Yes, it is possible to call the base method without creating an instance. And that method should be,Static method. Doing inheritance from that class.-Use Base Keyword from derived class.

**31. What is the difference between new and override?**

The new modifier instructs the compiler to use the new implementation instead of the base class function. Whereas, Override modifier helps to override the base class function.

**32. What are the various types of constructors?**

There are three various types of constructors , and they are as follows:.

–  Default Constructor – With no parameters.

–  Parametric Constructor – With Parameters. Create a new instance of a class and also passing arguments simultaneously.

–  Copy Constructor – Which creates a new object as a copy of an existing object.

**33. What is early and late binding?**

Early binding refers to assignment of values to variables during design time whereas late binding refers to assignment of values to variables during run time.

**34. What is ‘this’ pointer?**

THIS pointer refers to the current object of a class. THIS keyword is used as a pointer which differentiates between the current object with the global object. Basically, it refers to the current object.

**35. What is the difference betweenstructure and a class?**

Structure default access type is public , but class access type is private. A structure is used for grouping data whereas class can be used for grouping data and methods. Structures are exclusively used for dataand it doesn’t require strict validation , but classes are used to encapsulates and inherit data which requires strict validation.

**36. What is the default access modifier in a class?**

The default access modifier of a class is Private by default.

**37. What is pure virtual function?**

A pure virtual function is a function which can be overridden in the derived classbut cannot be defined. A virtual function can be declared as Pure by using the operator =0.

|  |  |
| --- | --- |
| 1  2  3 | Virtual void function1() // Virtual, Not pure    Virtual void function2() = 0 //Pure virtual |

**38. What are all the operators that cannot be overloaded?**

Following are the operators that cannot be overloaded -.

1. Scope Resolution (:: )
2. Member Selection (.)
3. Member selection through a pointer to function (.\*)

**39. What is dynamic or run time polymorphism?**

Dynamic or Run time polymorphism is also known as method overriding in which call to an overridden function is resolved during run time, not at the compile time. It means having two or more methods with the same name,same signature but with different implementation.

**40. Do we require parameter for constructors?**

No, we do not require parameter for constructors.

**41. What is a copy constructor?**

This is a special constructor for creating a new object as a copy of an existing object. There will be always only on copy constructor that can be either defined by the user or the system.

**42. What does the keyword virtual represented in the method definition?**

It means, we can override the method.

**43. Whether static method can use non static members?**

False.

**44. What arebase class, sub class and super class?**

Base class is the most generalized class , and it is said to be a root class.

Sub class is a class that inherits from one or more base classes.

Super class is the parent class from which another class inherits.

**45. What is static and dynamic binding?**

Binding is nothing but the association of a name with the class. Static binding is a binding in which name can be associated with the class during compilation time , and it is also called as early Binding.

Dynamic binding is a binding in which name can be associated with the class during execution time , and it is also called as Late Binding.

**46. How many instances can be created for an abstract class?**

Zero instances will be created for an abstract class.

**47. Which keyword can be used for overloading?**

Operator keyword is used for overloading.

**48. What is the default access specifier in a class definition?**

Private access specifier is used in a class definition.

**49. Which OOPS concept is used as reuse mechanism?**

Inheritance is the OOPS concept that can be used as reuse mechanism.

**50. Which OOPS concept exposes only necessary information to the calling functions?**

Data Hiding / Abstraction