**Software Testing**

Software testing is the process of:   
\* Evaluating a application is defects free,  
\*Ensure it meets the Stakeholders requirements.

Software Testing is the process of evaluating a software application to detect and fix defects before its release.

It ensures that the software meets the specified requirements and functions correctly under various conditions.

Testing is an essential part of the Software Development Life Cycle (SDLC) and helps improve software quality, performance, and security.

**Software testing can be categorized into:**

1. **Manual Testing** – Performed by testers without using automation tools.
2. **Automated Testing** – Uses testing tools and scripts to automate test execution.

**Types Of Testing:**

**1.Black Box Testing:**

Evaluates  the functionality  of  an  application  without  knowing  its internal  code  or  structure.

It  focuses on  inputs  and  expected  outputs  to   ensure  the  system  works  correctly.

**2.White Box Testing:**

Evaluates the functionality of an application while having knowledge of its internal code or structure.

It focuses on the internal workings of the system, ensuring that code logic, conditions, and paths function correctly.

**Principles of Software Testing**

Software testing follows fundamental principles to ensure effectiveness and efficiency.

**What is Testing Principles ?**

Testing principles are fundamental rules that guide the software testing process.

These principles help testers focus on the most critical aspects of testing, ensuring the quality and reliability of the software.

1.Early Testing.  
 2.Testing shows the presence of defects,not their Absence.  
 3.Exhaustive testing is Not Possibile.  
 4.Defects clustering.  
 5.Pesticide paradox.  
 6.Testing is context-dependent.   
7.Absence-of-errors fallacy

1. **Early Testing**
   * Testing should begin as early as possible in the development lifecycle to detect defects at an early stage
2. **Testing Shows the Presence of Defects**
   * Testing can confirm the presence of bugs but cannot prove the absence of defects.
3. **Exhaustive Testing is Impossible**
   * It is impossible to test all possible inputs and scenarios; instead, risk-based and priority-based testing is used. It is impossible to test all possible inputs and scenarios; instead, risk-based and priority-based testing is used.
4. **Defect Clustering**
   * A small number of modules typically contain most of the defects. Focusing on high-risk areas improves efficiency.
5. **Pesticide Paradox**
   * Running the same tests repeatedly may not uncover new defects. Test cases should be regularly reviewed and updated.
6. **Testing is Context-Dependent**
   * Different applications require different testing approaches based on their domain, complexity, and criticality.
7. **Absence-of-Errors Fallacy**
   * Finding and fixing defects does not guarantee a successful product. The software must also meet user requirements and expectations.

**? Why do you choose software testing as career.**

An) - Testing will be there in the market forever as long as quality demanded in the market.

* -Testing is mandatory for the industry.
* -Independent of technology.
* -Consistent and stable

**? Why software testing.**

* An) It is important bcz if there are any bug or errors in the software it can be identified early and can be resolved before delivery of the product.
* It is required to point out the defects & errors that were made during development phase .
* a tested software product ensures reliability, security & high performance .which result cost effectiveness & customer satisfaction.

**? What makes a good software testing engineer**

* A good test engineer should have a test to break attitude
* An ability to take the point of view of the customer
* Strong desire for quality and attention to minute details
* Tact and diplomacy to maintain a corporative relationship with develops
* Ability to communicate with both technical (developers)And non technical (customers, management people)
* Prior experience in the software development industry is always a plus
* Ability to judge the situations and make important decisions

**PDCA Cycle**

The plan-Do-Check-Act (PDCA) cycle is an iterative four step management method used in business to focus on continuous improvement of process, product or services and to resolve problems.

* P-**Plan**: define the objective, strategy and supporting methods to achieve the goal of a plan
* D-**Do**: here we implement the plan.
* C-**Check**: Evaluate the results to make sure whether we reach the goal as planned
* A-**Act**: Take action on what is not working as planned.

**Quality**

* It is defect free, robust, efficient and reliable, easy to learn and use product solutions
* It allowed to meet customer expectations, improves productivity.

**FIVE PERSPECTIVE OF QUALITY**

* Transcendent - I know it when I see it.
* Product based – processes desired features.
* User based – Fitness for use.
* Development and manufacturing based – Confirms to requirements.
* Value based – At an acceptable cost

**QA And QC**What is QA (Quality Assurance)?

 A set of rules and processes that help create high-quality software(Verification).

* Preventing defects before they happen.
* Ensuring that the software development process is good, so mistakes don’t occur.

**Example:  
Imagine a bakery. QA ensures that ingredients are fresh, ovens are set at the right temperature, and bakers follow a good recipe so that cakes come out perfect every time.**

**QA Activities in Software Testing:**

✅ Defining testing standards  
 ✅ Reviewing requirements and code  
 ✅ Process improvement  
 ✅ Training developers and testers

**What is QC (Quality Control)?**

* Inspecting the final product to catch and correct any mistakes.
* Finding and fixing defects (problems) in the software.
* Checking if the software actually works correctly before releasing it.

**Example:  
 In the same bakery, QC is like a taste tester checking cakes before selling them. If a cake is undercooked or too salty, it is fixed or discarded.**

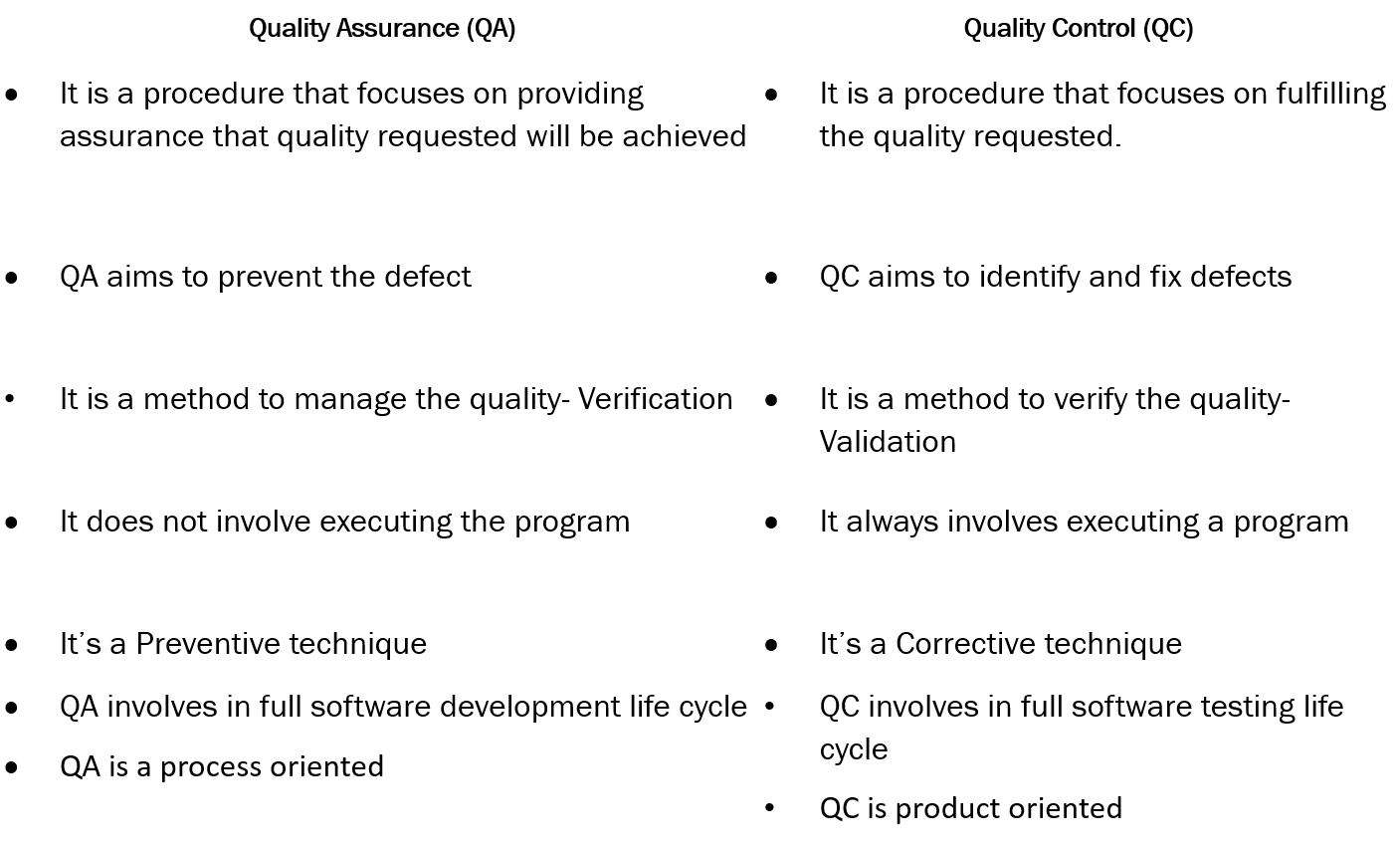
**QC Activities in Software Testing:**

**✅ Manual Testing  
 ✅ Automated Testing  
 ✅ Bug Identification & Reporting  
 ✅ Checking if the software meets requirements**

**Differences**

**QA:     QC:**

* Preventing defects \* Finding defects
* Process Improvement           \*Product Testing
* Proactive (before problems happen) \*Reactive (after problems happen)



**Conclusion:**

* QA is about making sure the right process is followed to avoid defects.
* QC is about checking the product to find and fix defects before release.