* **Software:** A set of instructions or commands that tells a computer what to do.

# Types of Software:

* 1. System Software: Operating System, Device Drivers, Servers, Utilities etc.
  2. Programming Software: Compilers, Debuggers, and Interpreters etc.
  3. Application Software: Web Applications, Mobile Apps, Desktop Applications etc.

# Software Testing:

* 1. Activity to detect and identify the defects in the software (Not fix).
  2. The process of evaluating and verifying that a software product or application does what it is supposed to do.
  3. To check whether the **Actual** software product matches **Expected** requirements and to ensure that software product is **Defect** free.

# Benefits of Software Testing

* **Customer satisfaction:** Serve the best features and experience to their customers
* **Cost effective**: Save a big amount for the owner.
* **Product Quality:** Deliver a quality product to its clients
* **Bug-free Application**: Identify bugs and inform them the concerned developing team to fix.
* **Quality:** Justification of all the requirements of a customer in a product.

# QA Engineer/Test Engineer:

* 1. Hard Skills: Technical Domain
  2. Soft Skills: Business Domain

# Hard vs Soft skills

* **Hard Skills**
  + **Basic QA Concept:** Testing Principles, Testing Process, testing mindset, Test levels. Test types.
  + **Basic Programming Knowledge:** Programming logic and basic knowledge in some programming languages.
  + **Basic Database Knowledge**: SQL commands.
  + **Test Automation:** Selenium, Appium, TestProject.
  + **Test Techniques:** Black – box techniques, White – box techniques, Exploratory testing, Testing heuristics.
  + **Test Tools:** Bug trackers, Test case management tools.

# Soft Skills

* + Interpersonal Communication
  + Proactivity
  + Conflict Resolution
  + Ability to work under pressure
  + Sense of leadership
  + Ownership
  + Analytical Ability

# Types of Software Testing:

Figure: Types of Software Testing

* **Functional Testing**
* Verifies the operations and actions of an application
* Based on requirements of customer
* Tests what the product does
* Helps to enhance the behavior of the application
* Easy to execute manually
* Done before Non-functional testing
* Examples:

1. Unit Testing
2. Smoke Testing
3. Integration Testing
4. Regression Testing

* **Non-functional Testing**
* Verifies the behavior of an application
* Based on expectations of customer
* Describes how the product does
* Helps to improve the performance of the application
* Hard to execute non-functional testing manually
* Done after functional testing
* Example:

1. Performance Testing
2. Scalability Testing
3. Stress Testing
4. Load Testing

* **Levels of Testing:**
  1. **Unit Testing**: Test the individual testing
  2. **Integration Testing**: Test integrated component
  3. **System Testing**: Test the entire system
  4. **Acceptance Testing**: Test the final system
* **Testing Pyramid:** Manual Testing
* **Unit Testing:**
* Testing of each unit or an individual component of the software application
* Done by developer
* **Code Coverage Techniques used in Unit testing:**

1. Statement Coverage
2. Decision Coverage
3. Branch Coverage
4. Condition Coverage
5. Finite State Machine Coverage

* **Integration Testing:**
* Software modules are integrated logically and tested as a group
* Performed between 2 or 4 modules
* Two parts of Integration techniques

1. **Incremental Integration Testing:** Integrating two or more modules that are logically related to each other and then tested for proper functioning of the application.
2. **Big Bang Approach:** All the components or modules are integrated together at once and then tested as a unit.

* **System Testing:**
* Performed on a completely integrated system to evaluate the compliance of the system with the corresponding requirements.
* A black box testing technique
* **Focus on:**

1. User Interface Testing (GUI)
2. Functional Testing
3. Non-functional Testing
4. Usability Testing

* **Types of system testing:**

1. Performance Testing
2. Reliability Testing
3. Usability Testing
4. Functionality Testing
5. Installation Testing
6. Documentation Testing
7. Regression Testing
8. Security Testing
9. Recoverability Testing
10. Load Testing

* **Process of System Testing:**

1. **Test Environment Setup:** Create testing environment
2. **Create Test Case:** Generate test case for the testing process
3. **Create Test Data:** Generate the data that is to be tested
4. **Execute Test Case:** Test cases are executed
5. **Defect Reporting:** Defects in the system are detected
6. **Regression Testing:** Carry out the side effects of the testing process
7. **Log Defects:** All the identified defects are fixed at this stage
8. **Retest:** A test is repeated if it's unsuccessful

* **Tools used for system testing**

1. JMeter
2. Gallen Framework
3. Selenium etc.

* **Regression Testing**
* A type of software testing to confirm that a recent program or code change has not adversely affected existing features
* Performed whenever there are changes or updates to a software application
* **Sanity Testing**
* A subset of Regression Testing
* Verifies only a particular component
* Done by testers
* Verifies the new functionality like bug fixes
* Not documented and scripted
* Conducted after the completion of regression testing
* Done on stable builds or for the introduced new features in the software
* Stable
* **Smoke Testing**
* A subset of acceptance testing
* Verifies the entire system from end to end
* Done by both developers or testers
* Verifies the critical functionalities of the system
* Documented and scripted
* Performed either manually or by using automation tools
* Carried out when every new build release
* Performs on the initial build
* Stable or unstable
* **Performance Testing**
* Non-functional testing
* Checking the behavior of an application by applying some load
* Types of Performance Testing

1. **Load Testing:** Used to check the performance of an application by applying some load which is either less than or equal to the desired load
2. **Stress Testing:** Checks the behavior of an application by applying load greater than the desired load
3. **Scalability Testing:** Checking the performance of an application by increasing or decreasing the load in particular scales (no of a user).
4. **Stability Testing:** Checking the performance of an application by applying the load for a particular duration of time.

* **Volume Testing**
* Non-Functional Testing
* Known as flood testing
* Testing the data load capabilities of a product
* Examines the stability and response time of a system by transferring huge volumes of data
* **Soak Testing**
* Non-functional testing
* Used to measure performance of a software application under a huge volume of load for an extended period of time