**1.** Write about Metrics and Measurement  in s/w testing ?
Answer :  A **Metric** is a quantitative **measure** of the degree to which a system, system component, or process possesses a given attribute. **Metrics** can be defined as "STANDARDS OF **MEASUREMENT**". **Software Metrics** are used to **measure** the quality of the project.

In software projects, it is most important to measure the quality, cost, and effectiveness of the project and the processes. Without measuring these, a project can't be completed successfully.

Here controlling the projects means, how a project manager/lead can identify the deviations from the test plan ASAP in order to react in the perfect time. The generation of test metrics based on the project needs is very much important to achieve the quality of the software being tested.

*A Metric is a quantitative measure of the degree to which a system, system component, or process possesses a given attribute.*
*Metrics can be defined as "STANDARDS OF MEASUREMENT".*
Software Metrics are used to measure the quality of the project. Simply, a Metric is a unit used for describing an attribute. Metric is a scale for measurement.

Suppose, in general, "Kilogram" is a metric for measuring the attribute "Weight". Similarly, in software, "How many issues are found in a thousand lines of code?", h*ere No. of issues is one measurement & No. of lines of code is another measurement. Metric is defined from these two measurements.*
**Test metrics example:**
 * How many defects exist within the module?
 * How many test cases are executed per person?
 * What is Test coverage %?

2. # Measurement :
*Answer :  Measurement is the quantitative indication of extent, amount, dimension, capacity, or size of some attribute of a product or process.*
**Test Measurement example:** Total number of defects.
Please refer below diagram for a clear understanding of the difference between Measurement & Metrics

Generation of Software Test Metrics is the most important responsibility of the Software Test Lead/Manager.

**Test Metrics are used to,**
 1. Take the decision for the next phase of activities such as, estimate the cost & schedule of future projects.
 2. Understand the kind of improvement required to success the project

3. Take a decision on the Process or Technology to be modified etc
.

**Importance of Software Testing Metrics:**

Test Metrics are the most important to measure the quality of the software.
 **how can we measure the quality of the software by using Metrics**?
Suppose, if a project does not have any metrics, then how the quality of the work done by a Test Analyst will be measured?

**For Example,** A Test Analyst has to,
1. Design the test cases for 5 requirements
2. Execute the designed test cases
3. Log the defects & need to fail the related test cases
4. After the defect is resolved, we need to re-test the defect & re-execute the corresponding failed test case.

In the above scenario, if metrics are not followed, then the work completed by the test analyst will be subjective i.e. the Test Report will not have the proper information to know the status of his work/project.
If Metrics are involved in the project, then the exact status of his/her work with proper numbers/data can be published.

**i.e. in the Test Report, we can publish:**
1. How many test cases have been designed per requirement?
2. How many test cases are yet to design?
3. How many test cases are executed?
4. How many test cases are passed/failed/blocked?
5. How many test cases are not yet executed?
6. How many defects are identified & what is the severity of those defects?
7. How many test cases are failed due to one particular defect? etc.

**Based on the above metrics, the Test Lead/Manager will get the understanding of the below mentioned key points.**
- %ge of work completed
- %ge of work yet to be completed
- Time to complete the remaining work
- Whether the project is going as per the schedule or lagging? etc.

*Based on the metrics, if the project is not going to complete as per the schedule, then the manager will raise the alarm to the client and other stakeholders by providing the reasons for lagging to avoid the last-minute surprises.*

**Definitions and Formulas for Calculating Metrics:**

**#1) %ge Test cases Executed**: This metric is used to obtain the execution status of the test cases in terms of %ge.

%ge Test cases Executed = **(**No. of Test cases executed / Total no. of Test cases written) * 100.

So, from the above data,

%ge Test cases Executed = (65 / 100) * 100 = 65%

**#2) %ge Test cases not executed**: This metric is used to obtain the pending execution status of the test cases in terms of %ge.

%ge Test cases not executed = **(**No. of Test cases not executed / Total no. of Test cases written) * 100.

So, from the above data,

%ge Test cases Blocked = (35 / 100) * 100 = 35%