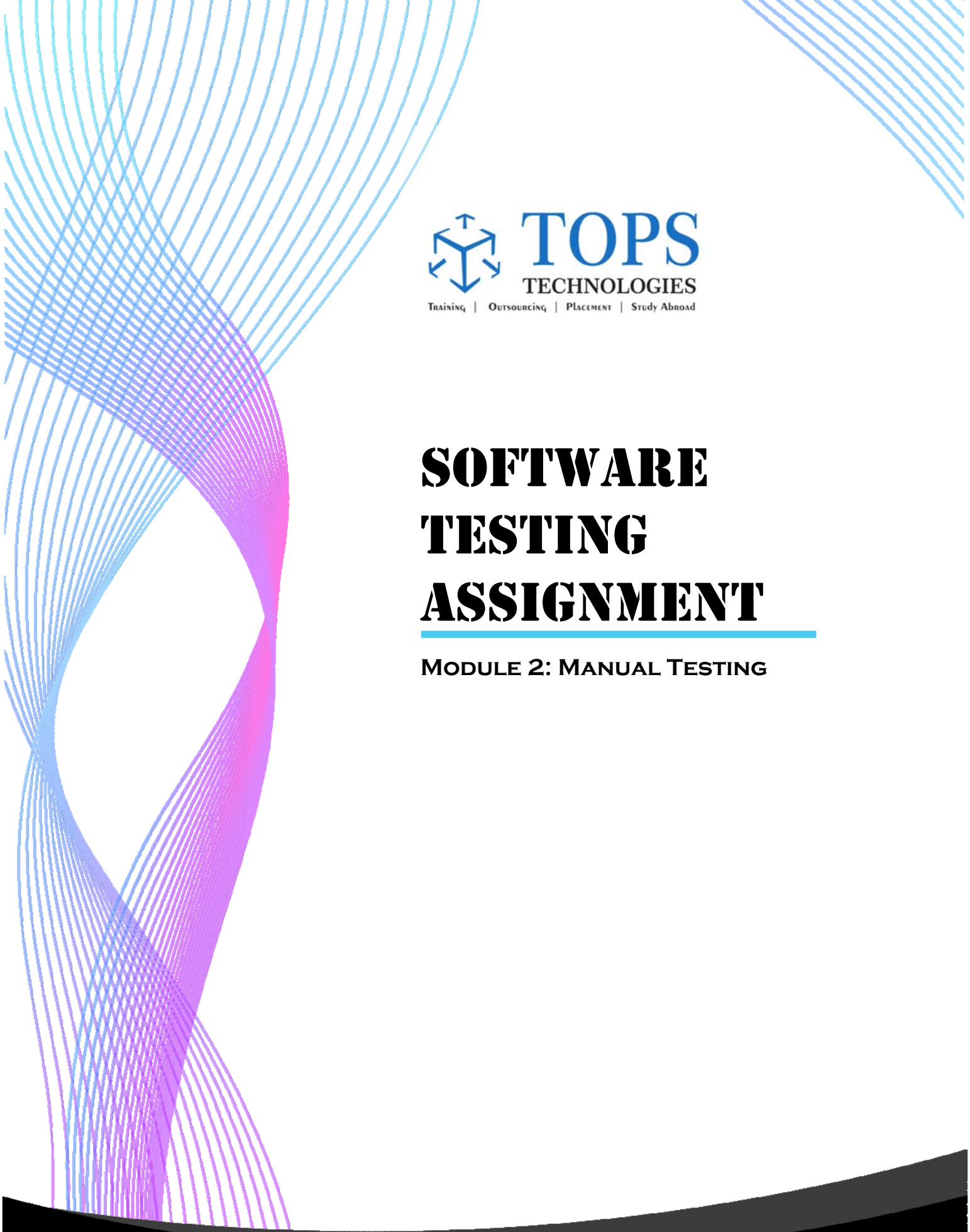


TOPS TECHNOLOGIES

Training | Outsourcing | Placement | Study Abroad

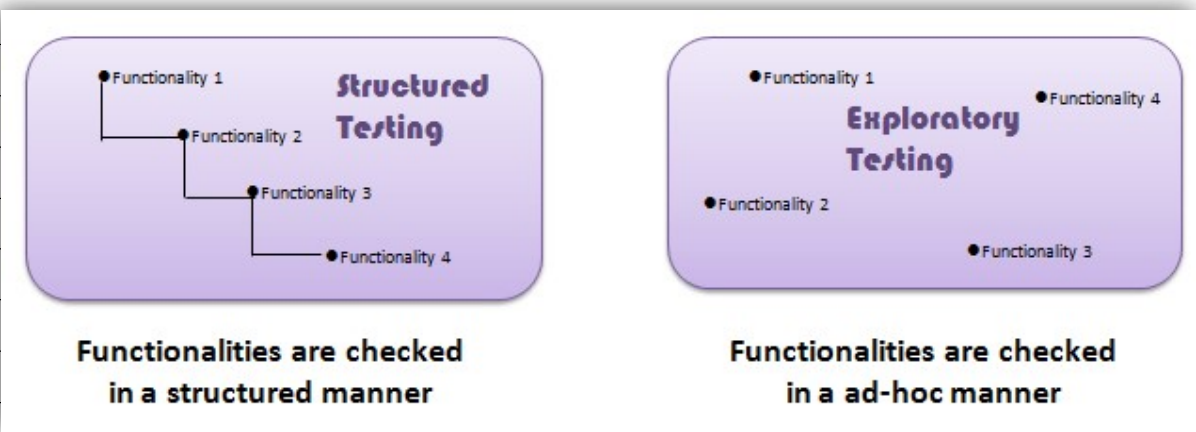
SOFTWARE TESTING ASSIGNMENT

Module 2: Manual Testing

Written by: Hardik Manharbhai Patel

Tops Technologies	**Name:** Patel Hardik Manharbhai **Assignment:** Software Testing: Module 2
	Module – 2 (Manual Testing)
Que: 1.	What is Exploratory Testing?
Que: 2.	What is traceability matrix?
Que: 3.	What is Boundary value testing?
Que: 4.	What is Equivalence partitioning testing?
Que: 5.	What is Integration testing?
Que: 6.	What determines the level of risk?
Que: 7.	What is Alpha testing?
Que: 8.	What is beta testing?
Que: 9.	What is component testing?
Que: 10.	What is functional system testing?
Que: 11.	What is Non-Functional Testing?
Que: 12.	What is GUI Testing?
Que: 13.	What is Adhoc testing?
Que: 14.	What is load testing?
Que: 15.	What is stress Testing?
Que: 16.	What is white box testing and list the types of white box testing?
Que: 17.	What is black box testing? What are the different black box testing techniques?
Que: 18.	Mention what are the categories of defects?
Que: 19.	Mention what big bang testing is?
Que: 20.	What is the purpose of exit criteria?
Que: 21.	When should "Regression Testing" be performed?
Que: 22.	What is 7 key principles? Explain in detail?
Que: 23.	Difference between QA v/s QC v/s Tester
Que: 24.	Difference between Smoke and Sanity?
Que: 25.	Difference between verification and Validation
Que: 26.	Explain types of Performance testing.
Que: 27.	What is Error, Defect, Bug and failure?
Que: 28.	Difference between Priority and Severity
Que: 29.	What is Bug Life Cycle?
Que: 30.	Explain the difference between Functional testing and NonFunctional testing

Tops Technologies	**Name: Patel Hardik Manharbhai** **Assignment: Software Testing: Module 2**
Que: 1.	**What is Exploratory Testing?**
Ans.	**Exploratory Testing** is a type of software testing where Test cases are not created in advance but testers
	check system on the fly. They may note down ideas about what to test before test execution. The focus of
	exploratory testing is more on testing as a “thinking” activity. Exploratory Testing is widely used in Agile
	models and is all about discovery, investigation, and learning.
	It emphasizes personal freedom and responsibility of the individual tester.
	Though the current trend in testing is to push for automation, exploratory testing is a new way of
	thinking. Automation has its limits.
	- Is not random testing but it is Adhoc testing with purpose of find bugs
	- Is structured and rigorous
	- Is cognitively (thinking) Structured as compared to procedural structure of scripted testing. This structure
	of scripted testing. This structure comes from Charter, time boxing etc.
	- Is highly teachable and manageable
	- Is not a technique but it is a n approach. What actions you perform next is governed by what you are
	doing currently.



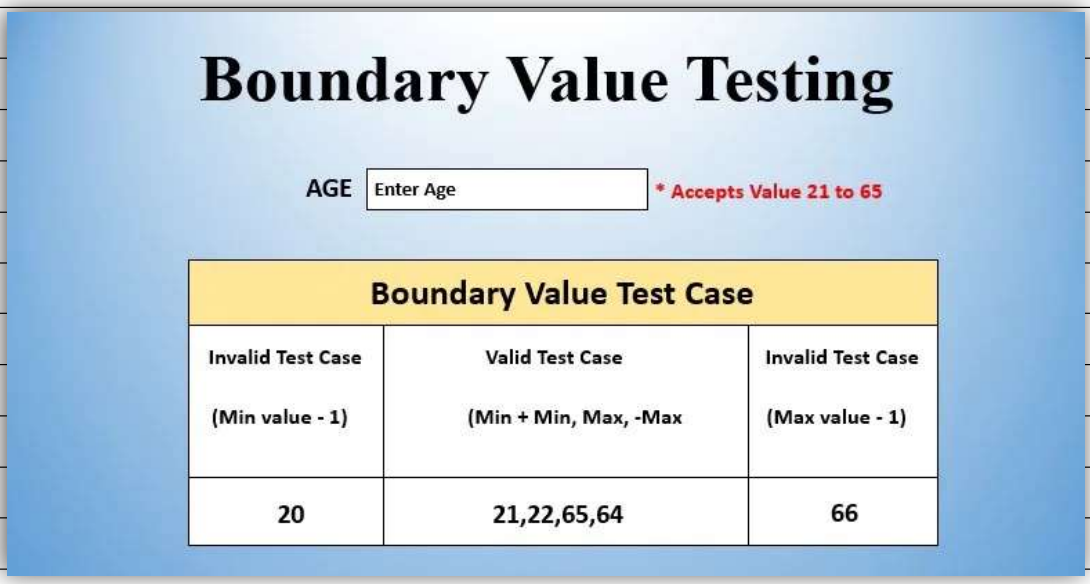
Que: 2.	**What is traceability matrix?**
Ans.	A Traceability Matrix is a document that co-relates any two-baseline documents that require a many-to-
	many relationship to check the completeness of the relationship.
	It is used to track the requirements and to check the current project requirements are met.
	To protect against changes, you should be able to trace back from every system component to the original
	Requirement that caused its presence.
	A software process should help you keeping the virtual table up-to-date, simple technique may be quite
	Valuable (naming convention)
	• Types of Traceability Matrix:
	- Forward Traceability – Mapping of Requirements to test cases
	- Backward Traceability – Mapping of Test Cases to Requirements
	- Bi-Directional Traceability – A Good Traceability matrix is the References from test cases to basis
	Documentation and vice versa.
	• **Pros:**
	- Easy to identify the missing functionalities.
	- To make sure that all requirements included in the test cases
	- To make sure that developers are not creating feature that no on has requested
	- Make obvious to that client that the software is being developed as per the requirements.
	- If there is a change request for a requirement, then we can easily find out which test cases need to
	update.
	- The completed system may have “Extra” functionality that may have not been specified in the
	design specification, resulting in wastage of manpower, time and effort.
	• **Cons:**
	- No traceability or Incomplete Traceability Result into:
	- Poor or unknown test coverage, more defects found in production
	- It will lead to miss some bugs in earlier test cycles which may arise in later test cycles. Then a lot of
	Discussions arguments with other teams and managers before release.
	- Difficult project planning and tracking, misunderstandings between different team over project
	dependencies, delays, etc.

Que: 3. What is Boundary value testing?

Ans. Boundary value analysis is a methodology for designing test case that concentrates software testing effort on cases near the limits of valid ranges

- Boundary value analysis is a method which refines equivalence partitioning.
- Boundary value analysis generates test cases that highlight errors better than equivalence partitioning.
- The trick is to concentrate software testing efforts at the extreme ends of the equivalence classes.
- At those points when input values change from valid to invalid errors are most likely to occur.
- Boundary value Analysis (BVA) uses the same analysis of partitions as EP and is usually used in conjunction with EP in test case design.

 # Let us assume a test case that takes the value of age from 21 to 65



From the above table, we can view the following inputs that are given.

- The minimum boundary value is given as 21.
- The maximum boundary value is given as 65.
- The valid inputs for testing purposes are 21, 22, 64 and 65.
- The invalid inputs for test cases are 20 and 66.

Test Case Scenarios:

1. Input: Enter the value of age as 20 (21-1)

Output: Invalid

2. Input: Enter the value of age as 21

Output: Valid

3. Input: Enter the value of age as 22 (21+1)

Output: Valid

4. Input: Enter the value of age as 65

Output: Valid

5. Input: Enter the value of age as 64 (65-1)

Output: Valid

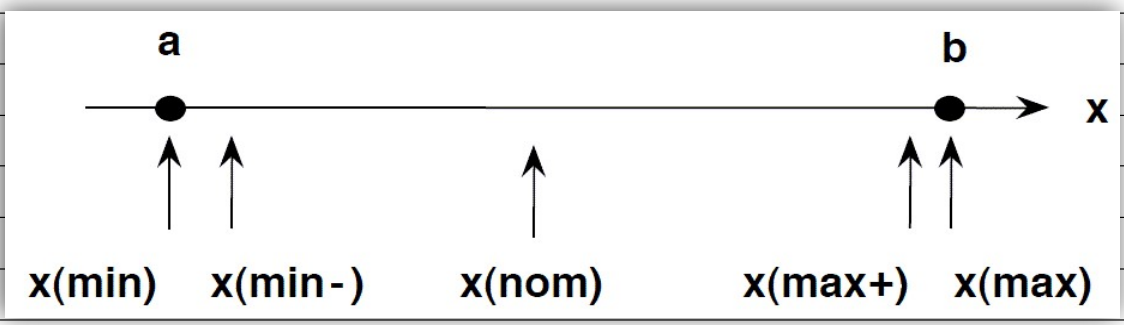
6. Input: Enter the value of age as 66 (65+1)

Output: Invalid

So these extreme ends like Start- End, Lower- Upper, Maximum-Minimum, Just Inside-Just Outside values are called boundary values and the testing is called “boundary testing”.

The basic idea in normal boundary value testing is to select input variable values at their:

1. Minimum
2. Just above the minimum
3. A nominal value
4. Just below the maximum
5. Maximum



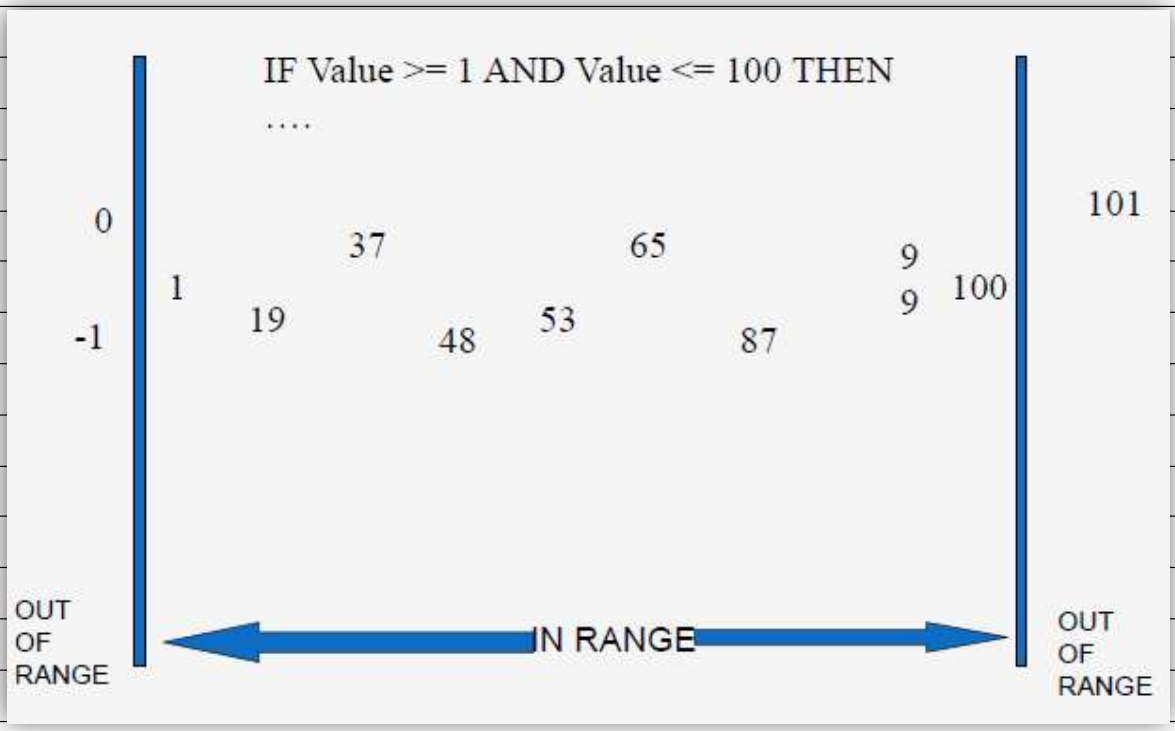
- In Boundary Testing, Equivalence Class Partitioning plays a good role
- Boundary Testing comes after the Equivalence Class Partitioning.

Que: 4. What is Equivalence Partitioning (E.P.) testing?

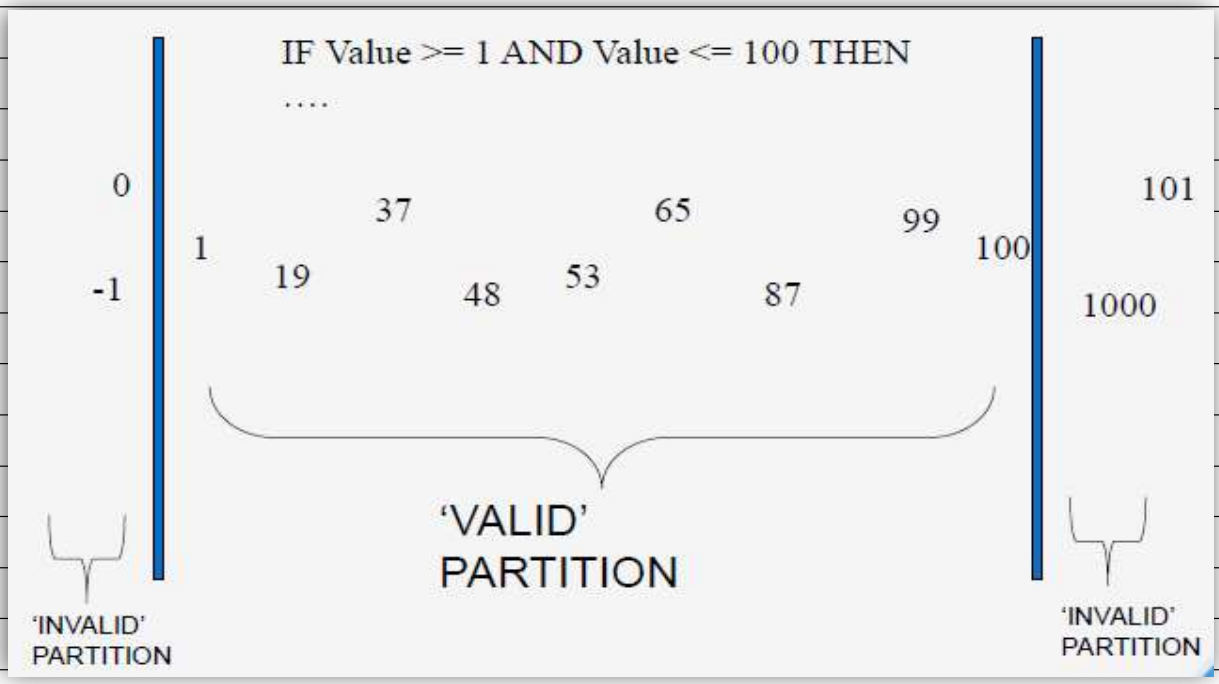
Ans. **Equivalence Partitioning** or Equivalence Class Partitioning is type of black box testing technique which can be applied to all levels of software testing like unit, integration, system, etc. In this technique, input data units are divided into equivalent partitions that can be used to derive test cases which reduces time required for testing because of small number of test cases.

- It divides the input data of software into different equivalence data classes.
- You can apply this technique, where there is a range in the input field.

- EP can be used for all Levels of Testing
- Aim is to treat groups of inputs as equivalent and to select one representative input to test them all



- The numbers fall into a partition where each would have the same, or equivalent, result i.e. an Equivalence Partition (EP) or Equivalence Class.
- EP says that by testing just one value we have tested the partition (typically a mid-point value is used). It assumes that:
  - If one value finds a bug, the others probably won't either
  - If one doesn't find a bug, the others probably won't either
- In EP we must identify Valid Equivalence partitions and Invalid Equivalence Partitions where Applicable (typically in range tests)
- The valid partition is bounded by the values 1 and 100
- Plus, there are 2 invalid partitions.


IF Value >= 1 AND Value <= 100 THEN
....
0
-1
1
19
37
48
53
65
87
99
100
101
1000
'VALID'
PARTITION
'INVALID'
PARTITION
'INVALID'
PARTITION

Que: 5.	**What is Integration testing?**
Ans.	Integration Testing - Testing performed to expose defects in the interfaces and in the interactions
	between integrated components or systems. **2) A**
	- Integration Testing is a level of the software testing process where individual units are combined
	and tested as a group.
	- The purpose of this level of testing is to expose faults in the interaction between integrated units.
	Test drivers and test stubs are used to assist in Integration Testing.
	- Integration testing tests integration or interfaces between components, interactions to different parts of
	the system such as an operating system, file system and hardware or interfaces between systems.
	- Integration testing is done by a specific integration tester or test team.
	- Components may be code modules, operating systems, hardware and even complete systems
	• **There are 2 levels of Integration Testing**
	- Component Integration Testing
	- System Integration Testing
	Need of Integration Testing
	• A Module in general is designed by an individual software developer who understanding and
	programming logic may differ from other programmers. Integration testing becomes necessary to verify
	the software modules work in unity.
	• At the time of module development, there wide chances of change in requirements by the clients.
	These new requirements may not be unit tested and hence integration testing becomes necessary.
	• Interfaces of the software modules with the database could be erroneous
	• External Hardware interfaces, if any, could be erroneous
	• Inadequate exception handling could cause issues.
	Methods of Integration Testing:
	During the process of manufacturing a ballpoint pen, the cap, the body, the tail
	and clip, the ink cartridge and the ballpoint are produced separately and unit tested separately. When two
	or more units are ready, they are assembled and Integration Testing is performed. For example, whether
	the cap fits into the body or not.
	- Any of Black Box Testing, White Box Testing, and Gray Box Testing methods can be used. Normally, the
	method depends on your definition of 'unit'.
	❖ **There are two types methods of Integration Testing:**
	▪ Bing Bang Integration Testing
	▪ Incremental Integration Testing
	✓ Top-Down Approach
	✓ Bottom-Up Approach

- Integration Testing is performed after Unit Testing and before System Testing.
- Either Developers themselves or independent Testers perform Integration Testing.

⚫ Entry Criteria:

- Unit Tested Components/Modules
- All High prioritized bugs fixed and closed
- All Modules to be code completed and integrated successfully.
- Integration test Plan, test case, scenarios to be signed off and documented.
- Required Test Environment to be set up for Integration testing

⚫ Exit Criteria:

- Successful Testing of Integrated Application.
- Executed Test Cases are documented
- All High prioritized bugs fixed and closed
- Technical documents to be submitted followed by release Notes.

⚫ Limitations:

Any condition not specified in integration tests, apart from the confirmation of the execution of the design items is usually not tested.

Que: 6.	**What determines the level of risk?**
Ans.	

Que: 7.	**What is Alpha testing?**
Ans.	Alpha testing is the initial phase of validating whether a new product will perform as expected.
	- It is always performed by the developers at the software development site.
	- Sometimes it is also performed by independent testing team.
	- Alpha testing is not open to the market and public.
	- It is conducted for the software application and project.
	- it is always performed in virtual environment.
	- it is always performed within the organization.
	- It is the form of Acceptance testing.
	- Alpha Testing is definitely performed and carried out at the developing organization location with the involvement
	of developers.
	- It comes under the category of both white box testing and black box testing.

Que: 8.	**What is beta testing?**
Ans.	It called also field testing.
	- It is always performed by the customers at their own site.
	- It is not performed by Independent Testing Team.
	- Beta Testing is always open to the market and public.
	- It is usually conducted for software product.
	- It is performed in Real Time Environment.
	- It is always performed outside the organization.
	- It is also the form of Acceptance Testing.
	- Beta Testing (field testing) is performed and carried out by users or you can say people at their own locations and site using customer data.
	- It is only a kind of Black Box Testing.
	- Beta Testing is always performed at the time when software product and project are marketed.
	- It is always performed at the user's premises in the absence of the development team.
	- It is also considered as the User Acceptance Testing (UAT) which is done at customers or users area.
	- Beta testing can be considered "pre-release" testing.
	- Pilot Testing is testing to product on real world as well as collect data on the use of product in the classroom.