Day\_4

TEST DESIGN TECHNIQUES

Software testing techniques help you to design better test cases.

Testing techniques helps to reduce number of test cases to be executed while increasing test coverage

>>>>List of test design techniques:

1. Boundary value analysis

2. Equivalence class partitioning

3. Decision table testing

4. State transition

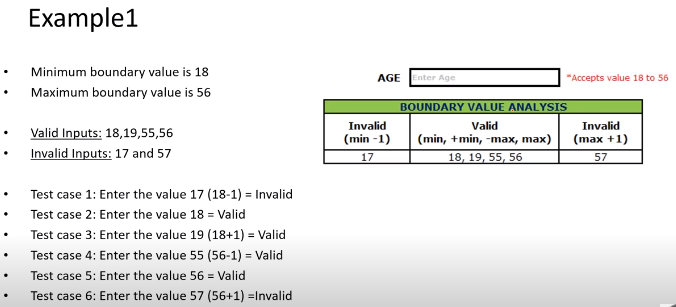
5. Error guessing

* 1. Boundary value analysis

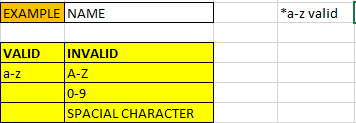
Boundary value analysis means testing at the boundaries between partitioning

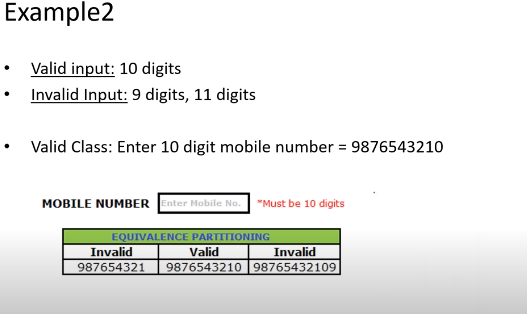
It includes maximum minimum inside and outside boundaries

Note -It focus on range



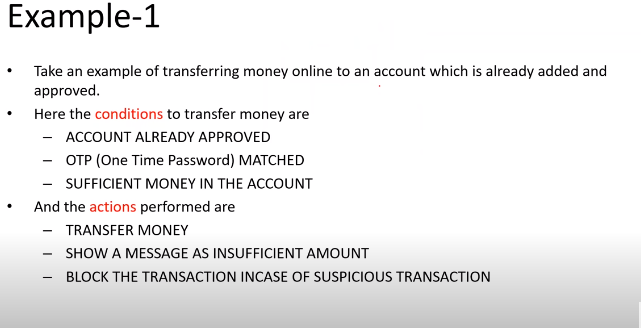
* 1. Equivalence class partitioning
* In equivalence class partitioning, input to the software are divided into groups and that are expected to exhibit similar behavior so they are likely to be proposed in the same way.
* Hence selecting one input from each group to design the test case.
* It helps to reduce the total number of test cases from infinite to finite ,selected test cases from the group ensure coverage of all possible scenario
* It focus on value

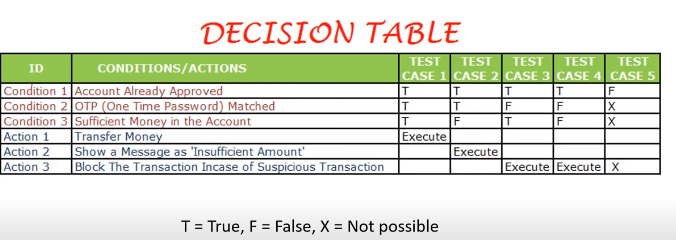




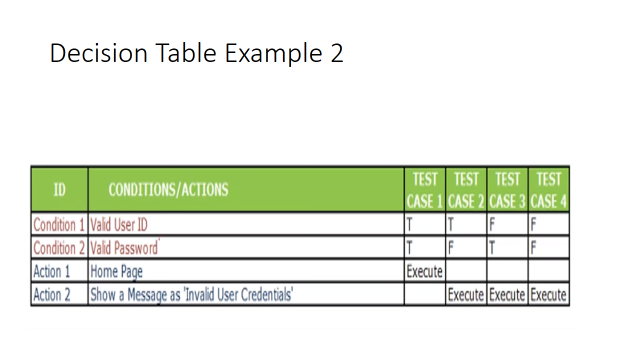
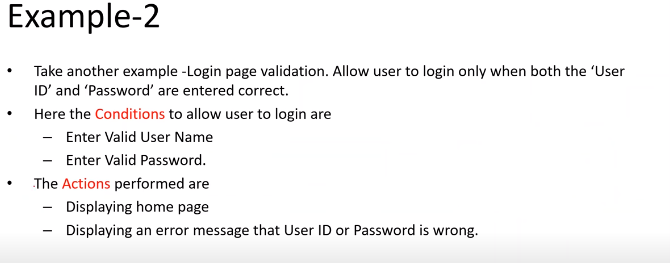
Decision table

* If we have more number of conditions then we go for decision table.
* Decision table is also called as cause - effect (decision –action) table.
* This test techniques appropriate for functionality which has logical relation between input (if –else logic).
* In decision table techniques we deal with combinations of input.
* To identify test case with decision table, we consider condition and action.
* We take condition as input and action as output.

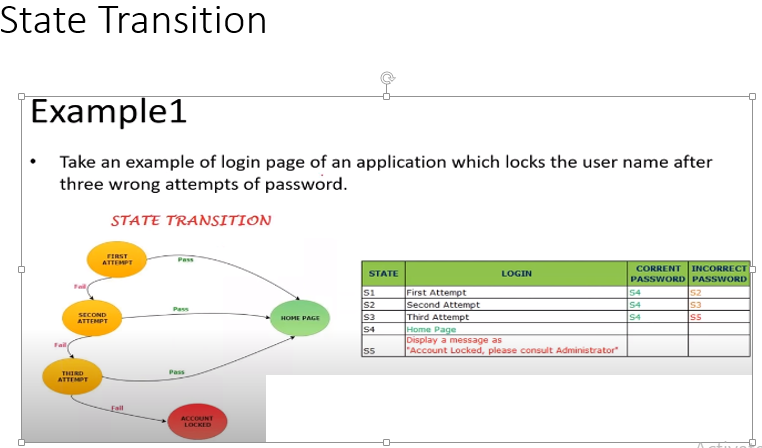


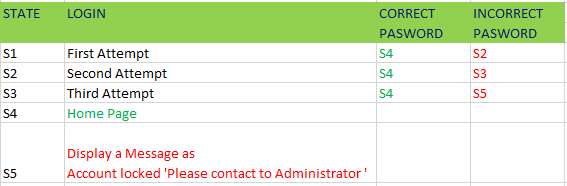


Example 2



State Transition

* In state transition changes in the input condition changes the state of application under test
* This testing techniques allow the tester to test the behavior of an (AUT )
* The tester can perform this action by entering various input condition in a sequence
* In state transition technique, testing team provide positive as well as negative input test value evaluating system behavior 



Error Guessing

Error guessing is one of the testing techniques used to find bugs in the software applications based on tester prior experience

In error guessing we don’t follow any specific rule

Example :

Submitting form without entering values

Entering invalid value such as entering alphabets in the numeric field