

# Testing Principle

The background of the slide features abstract, overlapping green geometric shapes, primarily triangles and polygons, in various shades of green, creating a modern and dynamic visual effect on the right side.

# Testing Principle

- **Testing is context dependent**
- Testing is done differently in different contexts.
- For e.g. Safety-critical software is tested differently from an e-commerce site.
- **Exhaustive testing is impossible**
- Testing everything (all combinations of inputs and preconditions) is not feasible except for trivial cases. Instead of exhaustive testing we use risks and priorities to focus testing efforts.
- **Early testing**
- Testing activities should start as early as possible in the software or system development life cycle and should be focused on defined objectives.

- **Defect clustering**

- A small number of modules contain most of the defects discovered during pre-release testing or show the most operational failures.

- **Pesticide paradox**

- If the same test are repeated over and over again, eventually the same set of test cases will no longer find any new bugs.
- To overcome this 'pesticide paradox' ,the test case needs to be regularly reviewed and revised to potentially find more defects.

- **Testing shows presence of defects**
- Testing can show that defects are present, but cannot prove that there are no defects.
- **Absence of error fallacy(missing req)**
- Finding and fixing defects does not help if the system built is unusable and does not fulfill the users needs and expectation.

# Interview Questions

- What are the principles of testing?
- What are main benefits of designing test cases early?
- What is 80/20 Rule?
- Explain the principle "Exhaustive Testing"?
- What steps can be taken to overcome pesticide paradox?
- Explain principle "absence of error fallacy"?

# Any Question

