Software Validation

- **Software Validation and Verfication (V & V)**
  - Testing is part of a broader process of software verfication and validation
  - Verfication
    - The software should conform to its specification
      - The process of checking that a software achieves its goal without any bugs
  - Validation
    - The software should do what the user really requires
      - The process of checking whether the software product is up to the mark or in other words product has high level requirements
  - Aim
    - The system is "fit for purpose"
      - Software purpose
        - How critical the software is to an organization
      - User expectations
      - Marketing environment
- **Program Testing**
  - Definition
    - Program testing, where the system is executed using simulated test data, is the principal validation technique
  - Goals
    - To demonstrate to the developer and the customer that the software meets its requirements
      - Validation testing
        - Perform correctly
    - To discover situations in which the behavior of the software is incorrect, undesirable or does not conform to its specification
      - Defect testing
        - Expose defects -- perform incorrectly
  - Stages
    - 1️⃣ Development testing
      - 1️⃣ Unit testing
      - 2️⃣ Component testing -- serveral units are integrated
        - Focus on showing that the component interface behaves according to its specification
      - 3️⃣ System testing -- all components
        - Components are integrated
          - Use-case testing
            - Each use case involves several system components
      - Integration testing
        - Involved in step 2 and 3: focus on interfaces between the modules
    - 2️⃣ Release testing
      - Definition
        - The process of testing a particular release of a system that is intended for use outside of the development team
      - Black-box Testing

- A form of system testing
- Requirements based testing
- Performance testing

3. User testing
    - Definition
        - A stage where users or customers provide input and advice on system testing
    - Types
        - Alpha testing
            - Users test the software at the developer's site
        - Beta testing
            - A release is available and users can experiment and raise problems
        - Accceptance testing
            - Decide whether or not it's ready to be accepted

Unit Testing

- Definition
    - The process of testing individual components in isolation
- Defect testing
- Units
    - Individual functions or methods within an object
    - Object classes with serval attributes and methods
        - Object class testing
            - Testing all operations associated with an object
            - Setting and interrogating all object attributes
            - Exercising the object in all possible states
    - Composite components with defined interfaces used to access their functionality
- Advantages
    - Help to fix bugs early in the development cycle and save costs
    - Helps the developers to understand the testing code base and enables them to make changes quickly
    - Good unit tests serve as project documentation

Automated Testing -- JUnit

- Components
    - A setup part
    - A call part
    - An assertion part
        - Assert statement
            - Assertions.assertEquals (4, calculator.multiply (2, 2), "optional failure message")
            - Assertions.assertTrue ("a" < "b", () -> "optional failure message")
            - Assertions.assertFalse ("a" > "b", () -> "optional failure message")
            - Assertion.assertNotNull (yourObject, "optional failure message")
            - Assertions.assertNull (yourObject, "optional failure message")