**How to work with** [**error handling in solidity - Require() and assert()**](https://community.letsupgrade.in/post/error-handling-in-solidity-require-and-assert-5f8e7b5c09c4c55d7fe7254a) **:**

* The assert(), require() and revert() functions for error handling were introduced in Solidity version **0.4.10.**
* The assert() and require() “guard” functions improve the readability of contract code.
* **require() is used to**:
  1. Validate user inputs.
  2. Validate the response from an external contract .i.e. require(external.send(amount)).
  3. Validate the state condition prior to execution.
  4. Generally it is used at the beginning of a function.
* **assert() is used to**:
  1. Check for overflow/underflow .i.e. c=a\*b; assert(c > b)
  2. Check invariants i.e. assert(this.balance > totalBalance).
  3. Validate state after making changes.
  4. Prevent condition which should never ever be possible.
  5. Generally it is used towards the end of a function.

* **Difference between require() and assert()**:

                1. They behave identically, but their bytecode output is slightly different.

* assert() uses the 0xfe opcode to cause an error condition
* require() uses the 0xfd opcode to cause an error condition

                2. require() refunds any leftover gas but assert() doesn't.

* It should be considered a normal occurrence for a require() statement to fail (same with revert()). When an assert() statement fails, something wrong has happened, and your code needs to be fixed.