* EvalError
  + Creates an instance representing an error that occurs regarding the global function eval().
* RangeError
  + Creates an instance representing an error that occurs when a numeric variable or parameter is outside its valid range.
* ReferenceError
  + Creates an instance representing an error that occurs when de-referencing an invalid reference.
  + Ex->
    - If we try to access a let/const variable during a temporal dead zone.
    - If a variable is not in the memory.
* SyntaxError
  + Creates an instance representing a syntax error.
  + Ex->
    - if we only declare a const variable without initializing it a syntax error is thrown.
    - Redeclaration of let variable.
* TypeError
  + Creates an instance representing an error that occurs when a variable or parameter is not of a valid type.
  + Ex->
    - if we try to reinitialize const variable then this error is thrown.
    - If we try to use arrow function as a constructor i.e invoking arrow function using new keyword.
* URIError
  + Creates an instance representing an error that occurs when encodeURI() or decodeURI() are passed invalid parameters.
* AggregateError
  + Creates an instance representing several errors wrapped in a single error when multiple errors need to be reported by an operation, for example by Promise.any().
* InternalError Non-Standard
  + Creates an instance representing an error that occurs when an internal error in the JavaScript engine is thrown. E.g. "too much recursion".