**JS Errors (later)**

|  |  |  |
| --- | --- | --- |
| try statement | let us to test a block of code for errors. | try {   *Block of code to try* } |
| catch statement | let us handle the error if an error occurs in the try block  (come in pairs with try statement) | catch(*err*) {   *Block of code to handle errors* } |
| throw statement | let us create custom errors. | throw "Too big";    // throw a text throw 500;          // throw a number |
| finally statement | let us execute code, after try and catch, regardless of the result. | try {   *Block of code to try* } catch(*err*) {   *Block of code to handle errors* } finally {   *Block of code to be executed regardless of the try / catch result* } |
| Error Object | provide two useful properties: name and message |  |
|  | EvalError: An error has occurred in the eval() function  (Newer version: not throw EvalError. Use SyntaxError instead) |  |
|  | RangeError: A number "out of range" has occurred | num.toPrecision(500);   // A number cannot have 500 significant digits |
|  | ReferenceError: An illegal reference has occurred | let x = 5; try {   x = y + 1;   // y cannot be used (referenced) } |
|  | SyntaxError: A syntax error has occurred | try {   eval("alert('Hello)");   // Missing ' will produce an error } |
|  | TypeError: A type error has occurred | num.toUpperCase();   // You cannot convert a number to upper case |
|  | URIError: An error in encodeURI() has occurred  (Uniform Resource Identifier) | decodeURI("%%%");   // You cannot URI decode percent signs |