

Node interactivo con REPL

Desarrollo de Aplicaciones Web Avanzado

PROGRAMADOR NOVATO



PROGRAMADOR EXPERTO

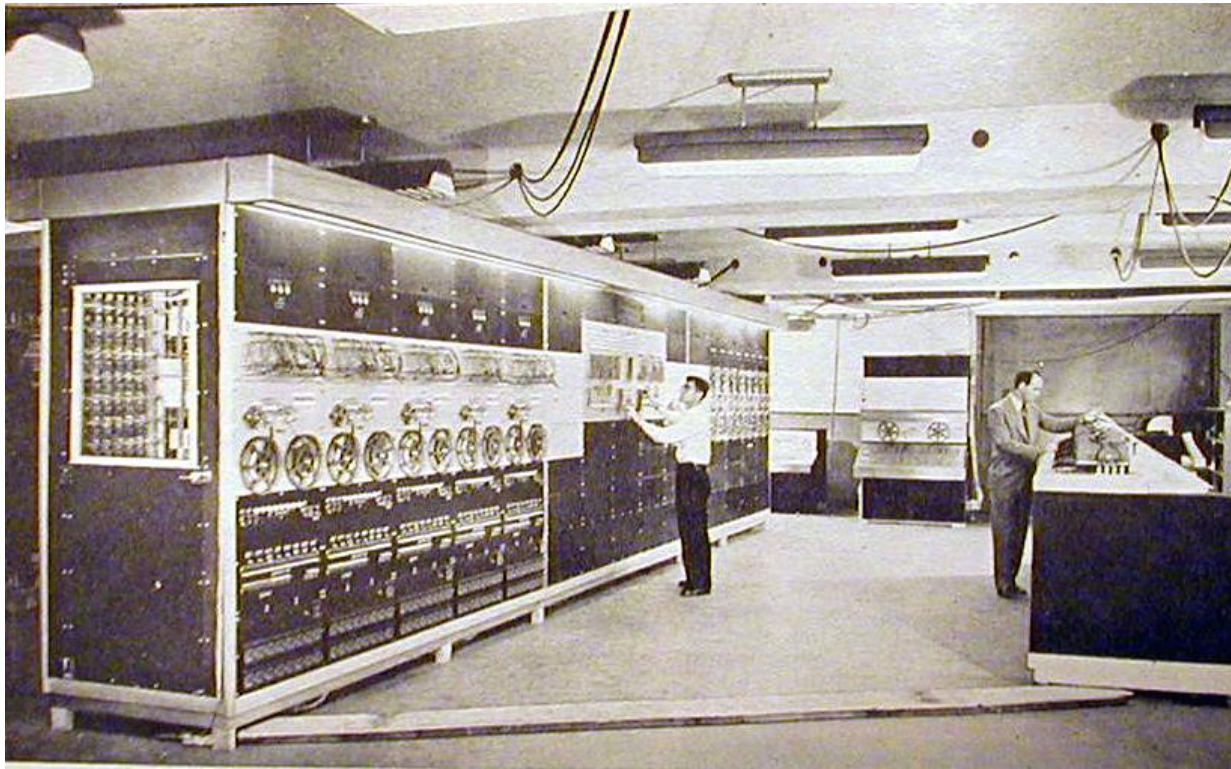


ERRORES EN PROGRAMACIÓN

- Un error de *software*, error o simplemente fallo, problema en un programa de computador o sistema de *software* que desencadena un resultado indeseado.
- A un error se le suele denominar BUG (insecto en inglés)
- En 1947, los creadores de Mark III informaron del primer caso de error en un ordenador causado por un bug.



- El Mark II, ordenador sucesor de ASCC Mark II, construido en 1944, sufrió un fallo en un relé electromagnético. Cuando se investigó ese relé, se encontró una polilla (bug) que provocó que el relé quedase abierto. Grace Murray Hopper, licenciada en física y destacada matemática que trabajó como programadora en el Mark II, pegó el insecto con cinta adhesiva en la bitácora.



9/9

0800 Antan started
 1000 " stopped - antan ✓
 1300 (032) MP-MC ~~1.982647000~~
 (033) PRO 2 2.130476415

{ 1.2700 9.037847025
 9.037846995 correct
 4.615925059(-2)

(033) PRO 2 2.130476415
 correct 2.130676415
 Relays 6-2 in 033 failed special speed test
 in relay " 10.000 test.

Relay
 2145
 Relay 3370

1100 Started Cosine Tape (Sine check)
 1525 Started Multi-Adder Test.

1545

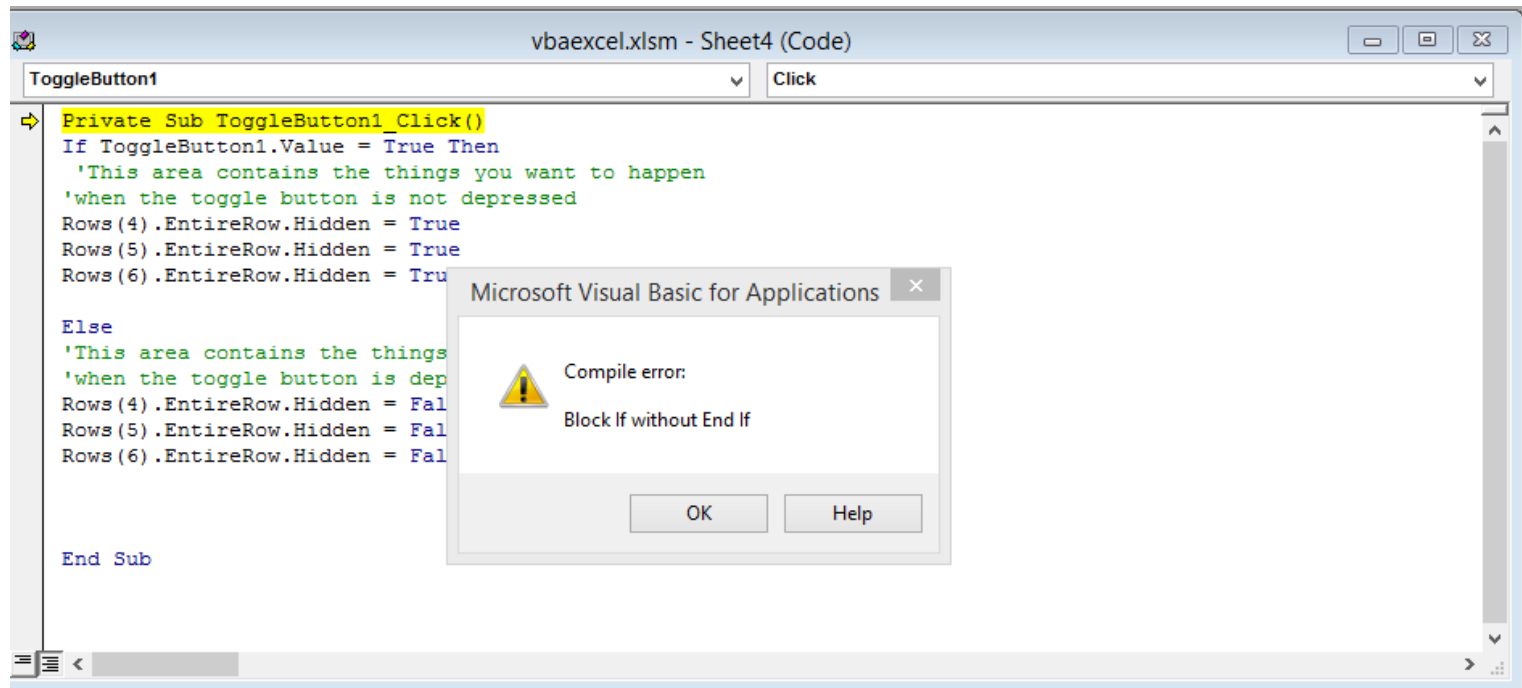


Relay #70 Panel F
 (moth) in relay.

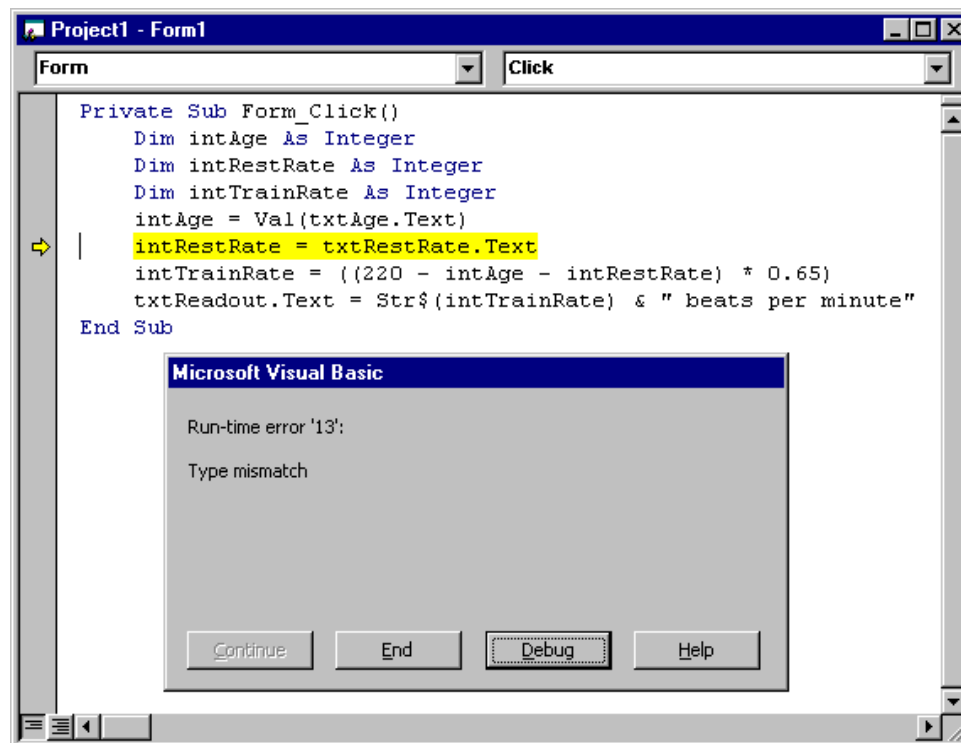
First actual case of bug being found.
 1630 Antan started.
 1700 closed down.

Tipos de errores

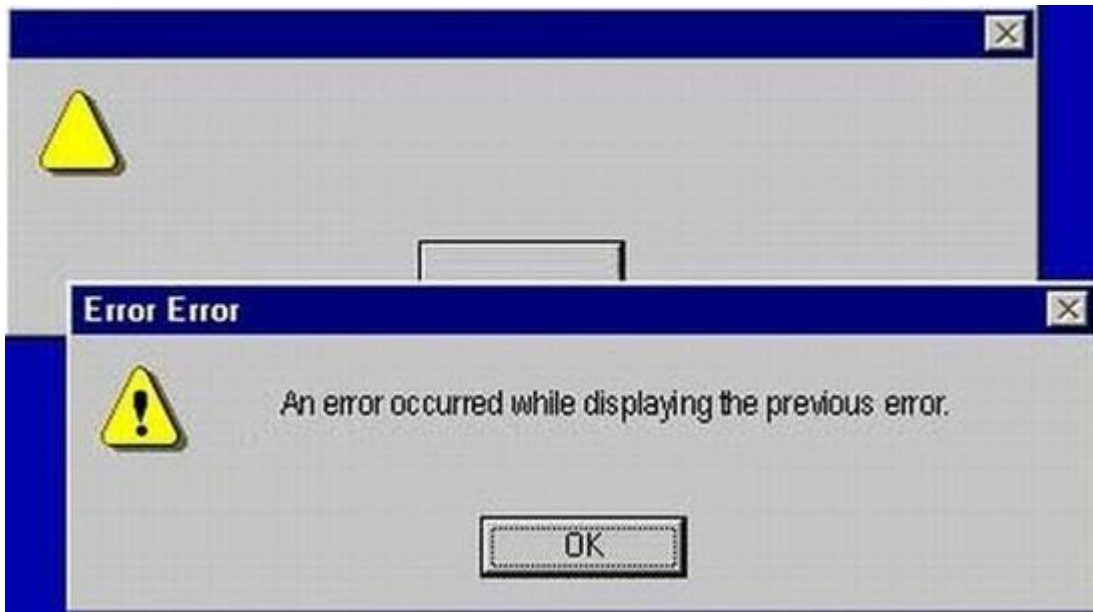
- **Errores de compilación:** Los errores de compilación, también conocidos como errores del compilador, son errores que impiden que su programa se ejecute. La mayoría de los errores del compilador se deben a errores cometidos al escribir el código.



- **Errores en tiempo de ejecución:** Los errores en tiempo de ejecución son errores que aparecen mientras se ejecuta su programa. Estos errores aparecen normalmente cuando su programa intenta una operación que es imposible que se lleve a cabo.



- **Errores lógicos:** Los errores lógicos son errores que impiden que su programa haga lo que estaba previsto. Su código puede compilarse y ejecutarse sin errores, pero el resultado de una operación puede generar un resultado no esperado.



- Incluso los programadores más experimentados cometen errores; y conocer cómo depurar una aplicación y encontrar esos errores es una parte importante de la programación.



REPL

- Node.js viene con un ambiente virtualizado llamado REPL (alias, Shell de node.js). REPL es una forma fácil de testear código Node.js/JavaScript.
- REPL viene del acrónimo de Read-Eval-Print-Loop

READ, EVAL, PRINT, LOOP

- **READ**, función que lee una expresión del usuario y la parsea en una estructura de información en memoria.
- **EVAL**, función que evalúa la información antes leída.
- **PRINT**, función que imprime el resultado.
- **LOOP**, función que repite los tres comandos arriba mencionados, hasta su culminación.

Para acceder al REPL

C:\> Administrador: Símbolo del sistema - node

```
Microsoft Windows [Versión 10.0.14393]  
(c) 2016 Microsoft Corporation. Todos los derechos reservados.
```

```
C:\WINDOWS\system32>node
```

```
>
```

Lista de funciones

- Presionamos TAB

```
>
Array
EvalError
Math
RangeError
SyntaxError
decodeURIComponent
isFinite
undefined

Boolean
Function
NaN
ReferenceError
TypeError
encodeURIComponent
isNaN

Date
Infinity
Number
RegExp
URIError
encodeURIComponent
parseFloat

Error
JSON
Object
String
decodeURI
eval
parseInt

ArrayBuffer
COUNTER_HTTP_SERVER_REQUEST
DTRACE_HTTP_CLIENT_REQUEST
DTRACE_HTTP_SERVER_CONNECTION
DataView
Int16Array
Uint32Array
buffer
clearTimeout
dgram
events
https
path
readline
setInterval
tls
util

Buffer
COUNTER_HTTP_SERVER_RESPONSE
DTRACE_HTTP_CLIENT_RESPONSE
DTRACE_NET_SOCKET_READ
Float32Array
Int32Array
Uint8Array
child_process
cluster
dns
fs
module
process
require
setTimeout
tty
vm

COUNTER_HTTP_CLIENT_REQUEST
COUNTER_NET_SERVER_CONNECTION
DTRACE_HTTP_SERVER_REQUEST
DTRACE_NET_SOCKET_WRITE
Float64Array
Int8Array
Uint8ClampedArray
clearImmediate
console
domain
global
net
punycode
root
stream
unescape
zlib

COUNTER_HTTP_CLIENT_RESPONSE
COUNTER_NET_SERVER_CONNECTION_CLOSE
DTRACE_HTTP_SERVER_RESPONSE
DTRACE_NET_STREAM_END
GLOBAL
Uint16Array
assert
clearInterval
crypto
escape
http
os
querystring
setImmediate
string_decoder
url

__defineGetter__
constructor
toLocaleString

__defineSetter__
hasOwnProperty
toString

__lookupGetter__
isPrototypeOf
valueOf

__lookupSetter__
propertyIsEnumerable

> █
```


Comando REPL	Descripción
.help	Muestra ayuda en todos los comandos
tab Keys	Muestra una lista de todos los comandos
Up/Down Keys	Muestra los comandos antes utilizados en el REPL
.save filename	Guarda la sesión actual de REPL a un archivo
.load filename	Carga el archivo especificado a la sesión actual de REPL
ctrl + c	Termina el comando actual
ctrl + c (twice)	Salir de REPL
ctrl + d	Salir de REPL
.break	Salir de expresión multilínea
.clear	Salir de expresión multilínea

GRACIAS POR SU ATENCIÓN