Documentazione linguaggio LoliCode - 22/09/2021

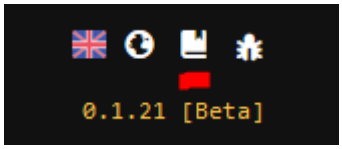
NioBrAin · Editor

GuideLoliCode

Ruri ·Sviluppatore

Introduzione

Questa è una raccolta di tutta la documentazione relativa a LoliCode che può essere trovata anche all'interno della webapp ufficiale openBullet 2 come mostrato nell'immagine.



Fare riferimento a questo argomento per qualsiasi problema relativo alla sintassi di LoliCode, alle variabili disponibili e allo zucchero di sintassi disponibile per le direttive C#.

Blocchi

Sintassi dei blocchi in LoliCode

Gli elementi opzionali sono racchiusi tra parentesi quadre.

BLOCK:Id

[LABEL:Custom label]

[DISABLED]

[settingName = settingValue]

[=> VAR/CAP @outputVariable]

ENDBLOCK

- Id: l'identificatore univoco del blocco
- settingName: il nome univoco di una determinata impostazione del blocco
- settingValue: vedi sotto
- => VAR/CAP @outputVariable: se il blocco restituisce un valore,
  è possibile definire se deve essere una variabile normale () o anche contrassegnato come acquisizione (VARCAP)

Impostazione dei valori

I valori di impostazione possono avere 3 tipi:

- Risolto (ad es. o "hello"123)
- Interpolato (ad es. $"My name is <name>")

- Variabile (ad es. @name)

Tipi a valore fisso

- Bool (o truefalse)
- Int (ad es. 123)
- Galleggiante (ad es. 0.42)
- Stringa (ad es. "hello")
- Byte Array (come base64 ad es. plvB6Yer)
- Elenco delle stringhe (ad es. ["one", "two", "three"])
- Dizionario delle stringhe (ad es. {("one", "first"), ("two", "second"), ("three", "third")})

Tipi di valori interpolati

- Stringa (ad es. $"This is my <name>")
- Elenco delle stringhe (ad es. $["one", "<secondNumber>", "three"])
- Dizionario delle stringhe (ad es. ${("one", "first"), ("<secondNumber>", "second"), ("three", "third")})

Note finali

- C'è il casting automatico del tipo, quindi è possibile utilizzare una variabile di tipo in un'impostazione di tipo IntString

Variabili

La variabile dati

Questa variabile contiene tutti i dati relativi al bot corrente.

Proprietà utili

- data.UseProxy (bool) se utilizzare il proxy assegnato al bot
- data.STATUS (string) lo stato attuale del bot
- data.RAWSOURCE (byte[]) il contenuto dell'ultima risposta http ricevuta
- data.SOURCE (string) come sopra ma come stringa
- data.ERROR (string) contiene il messaggio dell'ultima eccezione rilevata quando si utilizza la modalità provvisoria (nei blocchi che la supportano)
- data.ADDRESS (string) l'uri assoluto dell'ultima risposta http (dopo il reindirizzamento)
- data.RESPONSECODE (int) il codice di stato dell'ultima risposta http

- data.COOKIES (Dictionary<string, string>) i cookie inviati o ricevuti finora (es. data.COOKIES["PHPSESSID"])
- data.HEADERS (Dictionary<string, string>) le intestazioni dell'ultima risposta http (ad es. data.HEADERS["Location"])
- data.Objects (Dictionary<string, object>) contiene oggetti con stato per l'uso cross-block (verranno eliminati automaticamente alla fine dello script)
- data.MarkedForCapture (List<string>) tutti i nomi delle variabili contrassegnate per l'acquisizione

Linea

- data.Line.Data (string) l'intera riga di dati (non disasso) assegnata al bot
- data.Line.Retries (int) il numero di volte in cui i dati sono stati ritentati

Proxy

Nota: è null se i proxy sono disattivati, quindi fai sempre prima un controllo nulldata.Proxy

- data.Proxy.Host (string)
- data.Proxy.Port (int)
- data.Proxy.Username (string)
- data.Proxy.Password (string)
- data.Proxy.Type (ProxyType) può essere Http/Socks4/Socks5/Socks4a

Logger

- data.Logger.Enabled (bool) abilita o disabilita il logger (ad es. quando ci sono troppi dati da stampare)

Metodi utili

- data.MarkForCapture(string varName) aggiunge il nome della variabile all'elencodata.MarkedForCapture
- data.Logger.Log(string message, string htmlColor, bool canViewAsHtml) htmlColor deve essere ad es. o #fffwhite
- data.Logger.Log(IEnumerable<string> enumerable, string htmlColor, bool canViewAsHtml)
- data.Logger.Clear() cancella il registro

Dichiarazioni

LOG

Stampa il testo nel registro del debugger.
Esempio:

LOG "hello"

CLOG

Stampa testo colorato nel registro del debugger.
Un elenco completo dei colori è disponibile qui (rimuovere trattini e spazi e applicare PascalCase).
Esempio:

CLOG YellowGreen "hello"

JUMP

Passa a un punto specificato nel codice. Ricordati di fare attenzione ai loop infiniti!
Esempio:

...

#HERE

...

JUMP #HERE

REPEAT

Ripete qualcosa N volte.
Esempio:

REPEAT 5

LOG "hello"

END

FOREACH

Itera in una variabile elenco.
Esempio:

BLOCK:ConstantList

value = ["one", "two", "three"]

=> VAR @list

ENDBLOCK

FOREACH elem IN list

LOG elem

END

WHILE

Esegue qualcosa mentre una condizione è vera.
Esempio:

WHILE INTKEY 1 LessThan 2

...

END

IF / ELSE / ELSE IF

Esegue qualcosa o qualcos'altro.
Esempio:

IF INTKEY 5 LessThan 1

LOG "nope"

ELSE IF INTKEY 5 LessThan 3

LOG "nope again"

ELSE

LOG "yep"

END

TRY / CATCH

Esegue qualcosa. Se fallisce, esegue qualcos'altro.
Esempio:

TRY

// request to an unreliable URL

CATCH

// fallback request to a reliable URL

END

LOCK

Molto utile se si vogliono eseguire operazioni sincrone su variabili globali.
Si assicura che solo 1 bot possa inserire un determinato pezzo di codice alla volta, in modo che più bot non modifichino la stessa variabile globale contemporaneamente.
Spesso usato in combinazione con TRY / CATCH.
Esempio:

```
LOCK globals
TRY
// Try to increase globals.Count by 1 if it exists
globals.Count++;
CATCH
// Create globals.Count if it doesn't exist
globals.Count = 1;
END
END
```

ACQUIRELOCK / RELEASELOCK

Molto utile se si desidera eseguire operazioni asincrone su variabili globali.
Si assicura che solo 1 bot possa inserire un determinato pezzo di codice alla volta, in modo che più bot non modifichino la stessa variabile globale contemporaneamente.
È NECESSARIO utilizzarlo in combinazione con TRY/CATCH/FINALLY.
Esempio:

```
ACQUIRELOCK globals
TRY
// Do some async operation here
CATCH
throw; // Rethrow any exception
FINALLY
RELEASELOCK globals
END
```

SET VAR/CAP

Sets a string variable, and optionally also marks it for capture. Introduced for consistency with OB1.
Example:

```
SET VAR @myString "variable"
```

LOG myString

SET CAP @myCapture "capture"

LOG myCapture

SET USEPROXY

Sets whether to use the currently set proxy or not.
Example:

SET USEPROXY TRUE

SET USEPROXY FALSE

SET PROXY

Imposta un determinato proxy. I tipi disponibili sono: HTTP, SOCKS4, SOCKS4A, SOCKS5.
Esempio:

SET PROXY "127.0.0.1" 9050 SOCKS5

SET PROXY "127.0.0.1" 9050 SOCKS5 "username" "password"

MARK

Aggiunge una variabile all'acquisizione.
Esempio:

MARK **@myVar**

UNMARK

Rimuove una variabile dall'acquisizione.
Esempio:

UNMARK @myVar

TAKEONE

Prende un singolo elemento da una risorsa. È possibile configurare le risorse in Impostazioni di configurazione > Risorse di > dati.
È necessario fornire il nome della risorsa e il nome della variabile che verrà creata (di tipo).
Esempio:string

TAKEONE **FROM** "resourceName" => @myString

LOG myString

TAKE

Accetta più elementi da una risorsa. È possibile configurare le risorse in Impostazioni di configurazione > Risorse di > dati.
È necessario fornire il nome della risorsa e il nome della variabile che verrà creata (di tipo).
Esempio:List<string>

TAKE 5 **FROM** "resourceName" => @myList

AUTHOR : Ruri

EDITOR : NioBrAin