

**BS2\_GetLog** ..... 1

..... 1

..... 1

..... 1

..... 1

..... 1

..... 3

# BS2\_GetLog

ID                      amount      가                      .

```
#include "BS_API.h"

int BS2_GetLog(void* context, uint32_t deviceId, uint32_t eventId, uint32_t amount, BS2Event** logsObj, uint32_t* numLog);
```

## BS2Event

- [In] *context* : Context
- [In] *deviceId* :
- [In] *eventId* :                      ID                      , 0                      가                      .
- [In] *amount* :                      , 0                      eventId                      가                      .
- [Out] *logsObj* :
- [Out] *numLog* :



BS\_SDK\_SUCCESS                      ,                      가

### C++

```
if (BS_SDK_SUCCESS == sdkResult)
```

```

{
    for (uint32_t index = ; index < numOfLog; ++index)
    {
        BS2Event& event = logObj[index];
        latestIndex = event.id;
        cout << Utility::getEventString(id, event, timezone) << endl;

        if (event.image & 0x01)
        {
            uint32_t imageSize();
            uint8_t* imageBuf = new uint8_t[MAX_SIZE_IMAGE_LOG];
            memset(imageBuf, 0x0, sizeof(uint8_t) * MAX_SIZE_IMAGE_LOG);
            if (BS_SDK_SUCCESS == getImageLog(context, id, event.id,
imageBuf, imageSize))
            {
                // Your job.
                cout << "Image log received from " << id << " dateTime:" <<
event.dateTime + timezone
                    << " Event:" << event.id << endl;
            }

            delete[] imageBuf;
        }
    }

    if (logObj)
    {
        BS2_ReleaseObject(logObj);
        logObj = NULL;
    }
}

```

## C#

```

if (outNumEventLogs > 0)
{
    IntPtr curEventLogObjs = outEventLogObjs;
    for (UInt32 idx = 0; idx < outNumEventLogs; idx++)
    {
        BS2Event eventLog =
(BS2Event)Marshal.PtrToStructure(curEventLogObjs, structureType);
        Console.WriteLine(Util.GetLogMsg(eventLog));
        Console.WriteLine((eventLog.deviceID));
        Console.WriteLine((eventLog.code));
        Console.WriteLine((Encoding.ASCII.GetString(eventLog.userID)));
        Console.WriteLine((Util.ConvertFromUnixTimestamp(eventLog.dateTime).AddHours
(8)).ToString("yyyyMMddHHmmss")));
        curEventLogObjs += structSize;
        lastEventId = eventLog.id;
    }
}

```

```
API.BS2_ReleaseObject(outEventLogObjs);  
}
```

[BS2\\_GetFilteredLog](#)

[BS2\\_ClearLog](#)

From:

<https://kb.supremainc.com/kbtest/> - **BioStar 2 Device SDK**

Permanent link:

[https://kb.supremainc.com/kbtest/doku.php?id=ko:bs2\\_getlog&rev=1640829693](https://kb.supremainc.com/kbtest/doku.php?id=ko:bs2_getlog&rev=1640829693)

Last update: **2021/12/30 11:01**