

BS2_GetLogSmallBlob 1

..... 1

..... 1

..... 1

..... 2

..... 2

BS2_GetLogSmallBlob

[+ 2.6.4] EventMask 가 .
 BS2EventBlob BS2_GetLogBlob imageObj가
 6 BS2EventSmallBlob

```
#include "BS_API.h"
```

```
int BS2_GetLogSmallBlob(void* context, uint32_t deviceId, uint16_t eventMask, uint32_t eventId, uint32_t amount, BS2EventSmallBlob** logsObj, uint32_t* numLog);
```

BS2EventSmallBlob

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



BS_SDK_SUCCESS , 가

C++

```
int LogControl::getLogSmallBlob(BS2_DEVICE_ID id)
{
    uint16_t mask = BS2_EVENT_MASK_USER_ID;
    const uint32_t FROM_FIRST = ;
    BS2_EVENT_ID eID = FROM_FIRST;
    uint32_t amount(), numLog();
    BS2EventSmallBlob* blobObj = NULL;

    int sdkResult = BS2_GetLogSmallBlob(context_, id, mask, eID, amount,
    &blobObj, &numLog);
    if (BS_SDK_SUCCESS != sdkResult)
        TRACE("BS2_GetLogSmallBlob call failed: %d", sdkResult);

    print(blobObj, numLog);

    if (blobObj)
    {
        if (blobObj->imageObj)
            BS2_ReleaseObject(blobObj->imageObj);

        BS2_ReleaseObject(blobObj);
    }
    return sdkResult;
}
```

[BS2_GetLog](#)[BS2_GetFilteredLog](#)[BS2_ClearLog](#)[BS2_GetLogBlob](#)

From:

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

Permanent link:

https://kb.supremainc.com/kbtest/doku.php?id=ko:bs2_getlogsmallblobLast update: **2021/12/30 13:04**