목차

Log Management API > BS2_GetLogSmallBlob

BS2_GetLogSmallBlob

[+ 2.6.4] EventMask에 맞게 일정량의 로그를 가져옵니다.
BS2EventBlob을 이용하는 BS2_GetLogBlob와의 차이는
메모리 효율을 위해 6번째 인자 BS2EventSmallBlob 멤버 imageObj가 포인터 처리된 것을 제외하고 모두 동일합니다.

함수

```
#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);
```

> BS2EventSmallBlob구조체 보기

파라미터

- [In] *`context`* : Context
- [In] *`deviceId`* : 장치 식별자
- [In] *`eventMask`* : event mask
- [In] *`eventId`* : 로그 레코드 ID이며, 0일 경우 처음부터 가져옵니다.
- [In] *`amount`* : 최대 로그 레코드 갯수이며, 0일 경우 eventId 이후 모든 레코드를 가져옵니다.
- [Out] *`logsObj`* : 로그 레코드의 주소를 저장할 포인터
- [Out] *`numLog`* : 로그 레코드의 갯수

참고
logsObj 변수는 사용한 뒤 BS2_ReleaseObject 함수를 이용해 시스템에 메모리를 반환해야 합니다.

반환값

성공적으로 수행될 경우 BS_SDK_SUCCESS를 반환하고, 에러가 발생할 경우 상응하는 에러 코드를 반환합니다.

샘플코드

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;
}
```

함께 보기

BS2_GetLog
BS2_GetFilteredLog
BS2_ClearLog
BS2_GetLogBlob

From:
http://kb.supremainc.com/bs2sdk/ - **BioStar Device SDK**

Permanent link:
http://kb.supremainc.com/bs2sdk/doku.php?id=ko:bs2_getlogsmallblob

Last update: **2021/12/30 13:04**