목차

BS2_GetLog

주어진 로그 레코드 ID 이후의 로그를 최대 amount만큼 가져옵니다.

함수

```
#include "BS_API.h"

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

> BS2Event 구조체 보기

파라미터

- [In] *context* : Context
- [In] *deviceId* : 장치 식별자
- [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++

```
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);
}
```

함께 보기

BS2_GetFilteredLog
BS2_ClearLog