

**BS2\_GetUserSmallDatas** ..... 1

..... 1

..... 1

..... 2

..... 2

..... 3

# BS2\_GetUserSmallDatas

[+ 2.6.3] userMask 가 .  
 BS2UserBlob 3 BS2\_GetUserDatas BS2UserSmallBlob BS2UserPhoto가 .

```
#include "BS_API.h"

int BS2_GetUserSmallDatas(void* context, uint32_t deviceId, char* uids,
uint32_t uidCount, BS2UserSmallBlob* userBlob, BS2_USER_MASK userMask);
```

BS2UserSmallBlob

- [In] context : Context
- [In] deviceId :
- [In] uids : 가
- [In] uidCount :
- [Out] userBlob :
- [In] userMask : Mask

0x0000	
0x0001	data
0x0002	
0x0004	
0x0008	
0x0010	PIN
0x0020	
0x0040	
0x0080	
0x0100	
0x0200	
0x0400	
0x0800	(FaceStation F2)

0x1000	(FaceStation F2)
0xFFFF	

BS\_SDK\_SUCCESS , 가

C#

```

if (numUserIds > )
{
    IntPtr curUidObjs = outUidObjs;
    BS2UserSmallBlob[] userBlobs = new BS2UserSmallBlob[USER_PAGE_SIZE];

    Console.WriteLine("Number of users : ({0}).", numUserIds);
    for (UInt32 idx = ; idx < numUserIds; )
    {
        UInt32 available = numUserIds - idx;
        if (available > USER_PAGE_SIZE)
        {
            available = USER_PAGE_SIZE;
        }

        result = (BS2ErrorCode)API.BS2_GetUserSmallDatas(sdkContext, deviceID,
curUidObjs, available, userBlobs,
        (UInt32)BS2UserMaskEnum.ALL);
        if (result == BS2ErrorCode.BS_SDK_SUCCESS)
        {
            for (UInt32 loop = ; loop < available; ++loop)
            {
                print(userBlobs[loop].user);
                // don't need to release cardObj, FingerObj, FaceObj because we
get only BS2User
                if (userBlobs[loop].cardObjs != IntPtr.Zero)
                    API.BS2_ReleaseObject(userBlobs[loop].cardObjs);
                if (userBlobs[loop].fingerObjs != IntPtr.Zero)
                    API.BS2_ReleaseObject(userBlobs[loop].fingerObjs);
                if (userBlobs[loop].faceObjs != IntPtr.Zero)
                    API.BS2_ReleaseObject(userBlobs[loop].faceObjs);
                if (userBlobs[loop].user_photo_obj != IntPtr.Zero)
                    API.BS2_ReleaseObject(userBlobs[loop].user_photo_obj);
            }

            idx += available;
            curUidObjs += (int)available * BS2Environment.BS2_USER_ID_SIZE;

```

```
    }  
    else  
    {  
        Console.WriteLine("Got error({0}).", result);  
        break;  
    }  
}  
  
API.BS2_ReleaseObject(outUidObjs);  
}
```

[BS2\\_EnrollUserSmall](#)  
[BS2\\_EnrollUserSmallEx](#)  
[BS2\\_GetUserSmallInfos](#)  
[BS2\\_GetUserSmallInfosEx](#)  
[BS2\\_GetUserSmallDatas](#)  
[BS2\\_GetUserSmallDatasEx](#)

From:

<https://kb.supremainc.com/bs2sdk/> - **BioStar 2 Device SDK**

Permanent link:

[https://kb.supremainc.com/bs2sdk/doku.php?id=ko:bs2\\_getusersmalldatas&rev=1640764159](https://kb.supremainc.com/bs2sdk/doku.php?id=ko:bs2_getusersmalldatas&rev=1640764159)

Last update: **2021/12/29 16:49**