

**BS2\_GetAllAuthGroup** ..... 1

..... 1

..... 1

..... 1

..... 1

..... 2

# BS2\_GetAllAuthGroup

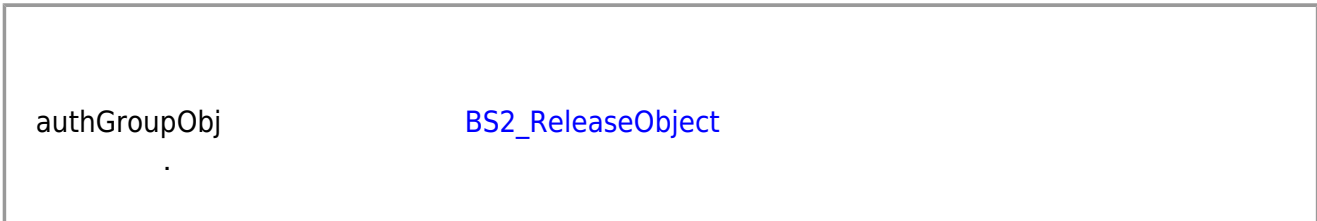
가 .

```
#include "BS_API.h"

int BS2_GetAllAuthGroup(void* context, uint32_t deviceId, BS2AuthGroup**
authGroupObj, uint32_t* numAuthGroup);
```

BS2AuthGroup

- [In] *context* : Context
- [In] *deviceId* :
- [Out] *authGroupObj* : auth group
- [Out] *numAuthGroup* : auth group



BS\_SDK\_SUCCESS , 가

- [BS2\\_GetAuthGroup](#)
- [BS2\\_SetAuthGroup](#)
- [BS2\\_RemoveAuthGroup](#)
- [BS2\\_RemoveAllAuthGroup](#)

C#

```
if (Util.IsYes())
{
    Console.WriteLine("Trying to get all auth gruops from device.");
    result = (BS2ErrorCode)API.BS2_GetAllAuthGroup(sdkContext, deviceID, out
authGroupObj, out numAuthGroup);
}
else
{
    Console.WriteLine("Enter the ID of the access group which you want to
get: [ID_1,ID_2 ...]");
    Console.Write(">>>> ");
    char[] delimiterChars = { ' ', ',', '.', ':', '\\t' };
    string[] authGroupIDs = Console.ReadLine().Split(delimiterChars);
    List<UInt32> authGroupIDList = new List<UInt32>();

    foreach (string authGroupID in authGroupIDs)
    {
        if (authGroupID.Length > )
        {
            UInt32 item;
            if (UInt32.TryParse(authGroupID, out item))
            {
                authGroupIDList.Add(item);
            }
        }
    }

    if (authGroupIDList.Count > )
    {
        IntPtr authGroupIDObj = Marshal.AllocHGlobal(4 *
authGroupIDList.Count);
        IntPtr curAuthGroupIDObj = authGroupIDObj;
        foreach (UInt32 item in authGroupIDList)
        {
            Marshal.WriteInt32(curAuthGroupIDObj, (Int32)item);
            curAuthGroupIDObj = (IntPtr)((long)curAuthGroupIDObj + 4);
        }

        Console.WriteLine("Trying to get auth gruops from device.");
        result = (BS2ErrorCode)API.BS2_GetAuthGroup(sdkContext, deviceID,
authGroupIDObj, (UInt32)authGroupIDList.Count, out authGroupObj, out
numAuthGroup);

        Marshal.FreeHGlobal(authGroupIDObj);
    }
    else
```

```
{  
    Console.WriteLine("Invalid parameter");  
}  
}  
BS2_ReleaseObject(uidObj);
```

From:

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

Permanent link:

[https://kb.supremainc.com/kbtest/doku.php?id=ko:bs2\\_getallauthgroup](https://kb.supremainc.com/kbtest/doku.php?id=ko:bs2_getallauthgroup)

Last update: **2022/06/27 15:02**