

**BS2\_EnrollUserFaceEx**

.....

.....

.....

.....

.....

.....

.....

1

1

1

1

1

2

## BS2\_EnrollUserFaceEx

[ + 2.7.1 ]

FaceStation F2

```
#include "BS_API.h"

int BS2_EnrollUserFaceEx(void* context, uint32_t deviceId,
BS2UserFaceExBlob* userBlob, uint32_t userCount, uint8_t overwrite);
```

**BS2UserFaceExBlob**

- [In] *context* : Context
- [In] *deviceId* :
- [In] *userBlob* :
- [In] *userCount* :
- [In] *overwrite* :

BS\_SDK\_SUCCESS , 가

C++

```
sdkResult = BS2_EnrollUserFaceEx(context_, id, &userBlob, 1, 1);
if (BS_SDK_SUCCESS != sdkResult)
    TRACE("BS2_EnrollUserFaceEx call failed: %d", sdkResult);

if (userBlob.cardObjs)
    delete[] userBlob.cardObjs;

if (userBlob.fingerObjs)
    delete[] userBlob.fingerObjs;
```

```
if (userBlob.face0bjs)
    delete[] userBlob.face0bjs;

if (userBlob.faceEx0bjs)
{
    //for (uint32_t index = 0; index < user.numFaces; index++)
    //{
    //    if (userBlob.faceEx0bjs[index].flag == 0)    // raw image
    //        delete[] userBlob.faceEx0bjs;
    //}
}
```

C#

```
sdkResult = (BS2ErrorCode)API.BS2_EnrollUserFaceEx(sdkContext, deviceID,
userBlob, 1, 1);
List<BS2User> userList = new List<BS2User>();
Int32 selection = Util.GetInput();
cbFaceOnReadyToScan = new API.OnReadyToScan(ReadyToScanForFace);
BS2User user = userList[selection];
BS2UserBlob[] userBlob11 = Util.AllocateStructureArray<BS2UserBlob>(1);
```

[BS2\\_EnrollUserFaceEx](#)  
[BS2\\_GetUserInfosFaceEx](#)  
[BS2\\_GetUserDatasFaceEx](#)

From:

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

Permanent link:

[https://kb.supremainc.com/kbtest/doku.php?id=ko:bs2\\_enrolluserfaceex](https://kb.supremainc.com/kbtest/doku.php?id=ko:bs2_enrolluserfaceex)

Last update: **2021/12/30 10:54**