

BS2_QueryDeviceLicense 1

..... 1

..... 1

..... 2

(C++) 2

(C#) 3

BS_SDK_SUCCESS , 가

(C++)

[sample_bs2_querydevicelicense.cpp](#)

```
int getDeviceLicense(void* context, BS2_DEVICE_ID id)
{
    DeviceControl dc(context);
    BS2LicenseBlob licenseBlob = { , };
    vector<BS2LicenseResult> licenseResult;
    int sdkResult = BS_SDK_SUCCESS;

    BS2_LICENSE_TYPE licenseType =
    (BS2_LICENSE_TYPE)Utility::getInput<uint32_t>("Enter the license type.
    (0: None, 1: Visual QR)");
    sdkResult = dc.queryDeviceLicense(id, licenseType, licenseResult);
    if (BS_SDK_SUCCESS == sdkResult)
        DeviceControl::print(licenseResult);

    return sdkResult;
}

int DeviceControl::queryDeviceLicense(BS2_DEVICE_ID id,
BS2_LICENSE_TYPE licenseType, vector<BS2LicenseResult>& licenseResult)
{
    BS2LicenseResult* result = NULL;
    uint32_t numOfResult = ;
    int sdkResult = BS2_QueryDeviceLicense(context_, id, licenseType,
    &result, &numOfResult);
    if (BS_SDK_SUCCESS != sdkResult)
    {
        TRACE("BS2_QueryDeviceLicense call failed: %d", sdkResult);
        return sdkResult;
    }

    licenseResult.clear();
    for (uint32_t idx = ; idx < numOfResult; idx++)
    {
        licenseResult.push_back(result[idx]);
    }

    return sdkResult;
}
```

(C#)[sample_bs2_querydevicelicense.cs](#)

```
Console.WriteLine("Trying to get a license");

Console.WriteLine("Enter the license type. (0: None, 1: Visual
QR)");
Console.Write(">>>> ");
UInt16 licenseType =
Util.GetInput((UInt16)BS2LicenseType.VISUAL_QR_MASK);

IntPtr resultObj = IntPtr.Zero;
UInt32 numOfResult = ;

BS2ErrorCode result =
(BS2ErrorCode)API.BS2_QueryDeviceLicense(sdkContext, deviceID,
licenseType, out resultObj, out numOfResult);

if (BS2ErrorCode.BS_SDK_SUCCESS != result)
{
    Console.WriteLine("Got error({0}).", result);
}
else
{
    IntPtr curResult = resultObj;
    int resultSize = Marshal.SizeOf(typeof(BS2LicenseResult));
    for (UInt32 idx = ; idx < numOfResult; idx++)
    {
        BS2LicenseResult item =
        (BS2LicenseResult)Marshal.PtrToStructure(curResult,
        typeof(BS2LicenseResult));
        print(item, idx);
        curResult += resultSize;
    }

    API.BS2_ReleaseObject(resultObj);
}
```

From:

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

Permanent link:

https://kb.supremainc.com/kbtest/doku.php?id=ko:bs2_querydevicelicenseLast update: **2023/03/02 16:03**