# 목차

SDK API > BS2_QueryDeviceLicense

# BS2_QueryDeviceLicense

[+ 2.9.1] 마스터 장치 및 그와 연결된 슬레이브 장치들의 라이선스 활성화/비활성화 상태를 가져옵니다.
이 기능은 장치가 라이선스 활성화 기능을 지원해야 하며, 현재 기능을 지원하는 장치 정보는 아래와 같습니다.

| 지원장치 | 펌웨어 |
|---|---|
| XS2-Finger | V1.2.0 |
| XS2-Card | V1.2.0 |
| BS3 | V1.1.0 |

## 함수

```
#include "BS_API.h"

int BS2_QueryDeviceLicense(void* context, uint32_t deviceId,
BS2_LICENSE_TYPE licenseType, BS2LicenseResult** outResultObj, uint32_t*
outNumOfResult);
```

> BS2LicenseResult 구조체 보기

## 파라미터

- [In] *context* : Context
- [In] *deviceId* : 장치 식별자
- [In] *licenseType* : 장치 라이선스 형태

| 값 | 설명 |
|---|---|
| 0x0000 | None |
| 0x0001 | Visual QR |

- [Out] *outResultObj* : 장치 라이선스 활성화 상태를 돌려받을 구조체의 포인터
- [Out] *outNumOfResult* : 장치 라이선스 활성화 결과 구조체의 갯수

> **참고**
> outResultObj 변수는 사용 후 BS2_ReleaseObject 함수를 통해 시스템에 메모리를 반환해야 합니다.

## 반환값

성공적으로 수행될 경우 BS_SDK_SUCCESS를 반환하고, 에러가 발생할 경우 상응하는 에러 코드를 반환합니다.

## 샘플코드(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;
}
```

## 샘플코드(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);
    }
```