

IMSI stands for "International Mobile Subscriber Identity".

IMSIApp.h

```
#include <e32base.h>
#include <Etel3rdParty.h>

class CIMSIApp : public CActive
{
private:
void ConstructL();
    * iTelephony
    ::TSubscriberIdV1 iSubscriberIdV1;
    CTelSubscriberIdV1Pckg iSubscriberIdV1Pckg;

public:
    (TDes& aIMSI)
static void GetIMSI(TDes& aIMSI;
    & IMSI;    TDes

private:
    /*
    These are the pure virtual methods from CActive that
    MUST be implemented by all active objects
    */
void RunL();
void DoCancel();
};
```

IMSIApp.cpp

```
#include "IMSIApp.h"

void CIMSIApp::GetIMSI(TDes& aIMSI)
{
    CIMSIAppnew (ELeave) CIMSIApp(aIMSI);
    CleanupStack(self);
    ->ConstructL();
    CleanupStackAndDestroy(self);
}

void CIMSIApp::ConstructL()
{
    iTelephony::NewL();
    CActiveScheduler(this);

    iTelephonySubscriberId(iStatus, iSubscriberIdV1Pckg);
    SetActive
    CActiveScheduler();
}

CIMSIApp:: CIMSIApp(TDes& imsi): CActive(EPriorityStandard), IMSI(imsi), iSubscriberIdV1Pckg(iSubs
{
//default constructor
}

void CIMSIApp::RunL()
```

Reading_IMSI_in_3rd_edition_-Synchronously

```
{
if(iStatus==KErrNone)
{
    = iSubscriberIMSIV1.iSubscriberId;
    CAStopScheduler
}
}

void CIMSIApp::DoCancel()
{
    iTelephonyAsync(CTelephony::EGetSubscriberIdCancel);
}
}
```

Now do the following steps:

- 1) Create a sample "Hello World" type application.
- 2) Put **IMSIApp.h** file in your application's "inc" folder.
- 3) Put **IMSIApp.cpp** file in your application's "src" folder.
- 4) Open .mmp file and add entry **SOURCE IMSIApp.cpp**.
- 5) Open .mmp file and add entry **LIBRARY etel3rdparty.lib**.
- 7) Now you need to include following header in your class to read IMSI:
#include "IMSIApp.h". Let's say in for e.g.: **"CYrAppUi.h"**
- 8) You can then call the static function from any of your Commands.
For ex: in your **CYrAppUi.h** you need to include **"IMSIApp.h"**
And Call it like:

```
case EReadIMSICommand1:
{
    TBuf<25> iIMSI;
    CIMSIApp::GetIMSI(iIMSI);
    CAknInformationNote* informationNote = new (ELeave) CAknInformationNote;
    informationNote->ExecuteLD(iIMSI);
}
}
```

- 9) Open .mmp file and add entry **CAPABILITY ReadDeviceData**.
- 10) You need to sign resulted **.Sis** file with your developer certificate to get it installed on your phone.

You can find a working application which can be built with Carbide.c++
Here it goes: [[Read IMSI](#)]

See Also

- [IMSI on Wikipedia](#)
- [Online tool to analyze IMSI](#)
- [DevInfo - Get the IMEI, IMSI, CellId etc., synchronously on 3.x devices.](#)