

Getting_contact_info_from_default_database

The code example below gets contact info from default contact database according to application needs. In the example we get general mobile phone number, home mobile phone number and work mobile phone number from default contact database. The code logic explained below. First of all we should open default database.

Header required:

```
#include <cntdb.h> // CContactDatabase
#include <cntitem.h> //CContactItem,CContactItemFieldSet
```

Library needed:

```
LIBRARY cntmodel.lib // CContactDatabase, CContactItem,CContactItemFieldSet
```

Source code:

```
CContactDatabase* contact_db;
TBuf<60> dbFile;
//gets default contact database file
CContactDatabase::GetDefaultNameL( dbFile );
//opens default contact database file
TRAPD(err, contact_db= CContactDatabase::OpenL(dbFile));
```

Because it is possible to have few the same type contact info we iterate through contacts first.

```
//create iterarot for contact_db
TContactIter contact_iterator(*contact_db);
TContactItemId contact_id;
//go through contacts while there any info
while( ( contact_id = contact_iterator.NextL() ) != KNullContactId)
```

Then we iterate through fields itself.

```
CContactItem* contact_item = contact_db->ReadContactL(contact_id);
CContactItemFieldSet &field_set = contact_item->CardFields();

//go through all the fields of contact and extract info according to //application needs
for (TInt i = 0 ; i < field_set.Count() ; i++)
```

After that we extract necessary information and process it according to application logic.

```
if(field.ContentType().ContainsFieldType(KUidContactFieldVCardMapCELL)&& field.ContentType().Cont
{
// This is work mobile processing. It will get all work mobiles in the //contact list
    WnnbHil&0',1);
    Wmpyifield.TextStorage()->Text());
    WmobTeeminate();
//process work mobile according to application logic
continue;
}
```

Note: It is very important to get work and home mobile first before general one! Otherwise it will mix it. This rule applies to all contact info.

The complete function code below:

```
//open current contacts from default database
```

Getting_contact_info_from_default_database

```

void CContacts::OpenContacts()
{
    CContactDatabase db;
    <60>TdbFile;
//gets default contact database file
    CContactDatabase defaultNameL( dbFile );
//opens default contact database file
    (err = db.Open(dbFile));
    //process errors if any
    if ( err == KErrNotFound )
    {
        // "does not exist"
    }
    else if( err == KErrLocked )
    {
        // "locked"
    }
    else if( err == KErrBadName )
    {
        // "bad name"
    }
    else if( err == KErrDiskFull )
    {
        // "disk full"
    }
    else if (err != KErrNone)
    {
        // "error opening default contact db"
    }

    //create iterator for contact_db
    TContactIter contact_iterator(contact_db);
    TContactItemId contact_id
//general mobile, home mobile and work mobile
    <100>Tmobile(0), hmobile(0), wmobile(0);

    //go through contacts while there any info
    while( ( contact_id = contact_iterator.NextL() ) != KNullContactId)
    {
        * contact_item = contact_db->ReadContactL(contact_id);
        ::PushInopsack(item);

        CContactItem field(contact_item->CardFields());
//go through all the fields of contact and extract info according to application needs
        for (TInt i = 0 ; i < field_set.Count() ; i++)
        {
            & field_contact_item_field
//work mobile
            if (field.ContentType().ContainsFieldType(KUIdContactFieldVCardMapCELL)&& field.ContentType().Con
            {
                // This is work mobile processing. It will get all work mobiles in the contact list
                Fill('\0',1);          wmobile.
                Copy(field.TextStorage()wmobile);
                ZeroTerminate();      wmobile.
//process work mobile according to application logic
                continue;
            }
            else if (field.ContentType().ContainsFieldType(KUIdContactFieldVCardMapCELL)&& field.ContentType
            // This is home mobile processing. It will get all home mobiles in the contact list
                Fill('\0',1);          hmobile.
                Copy(field.TextStorage()hmobile);
                ZeroTerminate();      hmobile.
//process home mobile according to application logic
                continue;
            }
        }
    }
}

```

Getting_contact_info_from_default_database

```
}else if (field.ContentType().ContainsFieldType(KUIdContactFieldVCardMapCELL)) {
// This is general mobile processing. It will get all general mobiles in the contact list
    Fill('\0',1);          mobile.
    Copy(field.TextStorage()->Text());
    ZeroTerminate();      mobile.
//process general mobile according to application logic
continue;
}
//TO DO add other KUIdContact fields according to application logic
}
    ->CloseContactItem(contact_item->Id());
    ::PopAndDestroy();
}
}
```

Similar Link:

- [Getting Contact Database Change Event](#)
- [Adding a Contact Item](#)
- [How to use contact database](#)
- [Reading and Editing a Contact Item](#)