



We can receive events about contact database changes in our application, the **MContactDbObserver** base class must be inherited and the virtual function **HandleDatabaseEventL()** must be implemented. This function is called when a contact database event occurs.

If you are not familiar with Observer Pattern, please refer to [\[Observer Section\]](#) on Wiki.

Header required:

```
#include <CNTDBOBS.H> //MContactDbObserver
#include <cntdb.h> // CContactDatabase, CContactChangeNotifier
```

Library needed:

```
LIBRARY cntmodel.lib // CContactDatabase, CContactChangeNotifier
```

Source code:

```
class CContactDatabaseObserver : public CBase, MContactDbObserver
{

public:

    CContactDatabaseObserver();

    // we need to override the virtual function from the base class //

    virtual void HandleDatabaseEventL( TContactDbObserverEvent  aEvent );

};

// Handle contact database events //

void CContactDatabaseObserver::HandleDatabaseEventL(TContactDbObserverEvent
aEvent)
{
    switch(aEvent.iType)
    {
        //if any contact deleted in phonebook
        case EContactDbObserverEventContactDeleted:
            break;
        //event if any contact changed
        case EContactDbObserverEventContactChanged:
        {}
            break;
        //event if any new contact added to phonebook
        case EContactDbObserverEventContactAdded:
        {}
            break;
        //check TContactDbObserverEventType for more events.
    }
}
```

Getting_Contact_Database_Change_Event

```
//We must also create the CContactChangeNotifier object to register itself to  
//receive events.
```

```
CContactDatabase* ContactDatabase = CContactDatabase::OpenL();
```

```
// Here iContactDatabaseObserver is a CContactDatabaseObserver pointer //
```

```
CContactChangeNotifier* DatabaseNotifier =
```

```
CContactChangeNotifier::NewL(*ContactDatabase , this);// Changes by amitgupta on date 5/06/2008
```

Related Links:

- [Getting contact info from default database](#)