

**Note!**

This API is not part of the public SDK. It can be found in the [SDK API Plug-in](#).

The Logs Timers plug-in API provides access to the timers/counters related to Logs processing. It provides access to the call log central repository which we can query using the keys for the information like number of new missed calls, call duration of last call, total dialled calls and received calls, also whether the settings of logs: "Log duration" and "Show call duration" are enabled or disabled.

Example code

// Headers Used

```
#include <centralrepository.h>
#include <LogsInternalCRKeys.h>
```

// Library used : centralrepository.lib

The following code displays "You have XX missed calls" :

```
// We need to Query Central Repository.
CRepository* cRepository = CRepository::NewLC( KCRUidLogs);

// Get the integer number
TInt num;
User::LeaveIfError( cRepository->Get( KLogsNewMissedCalls, num ) );

// To display message

CAknInformationNote* informationNote = new (ELeave) CAknInformationNote;
TBuf<100> msg;
msg.Copy(_L("You have "));
msg.AppendNum(num);
msg.Append(_L(" missed calls"));

informationNote->ExecuteLD(msg);

CleanupStack::PopAndDestroy();
```

The following code displays the call duration of last call in seconds :

```
// We need to Query Central Repository.
CRepository* cRepository = CRepository::NewLC( KCRUidLogs);

// Get the time in seconds
TInt time;
User::LeaveIfError( cRepository->Get( KLogsActualLastCallTimer, time ) );

// To display message
```

Logs_Timers_API

```
CAknInformationNote* informationNote = new (ELeave) CAknInformationNote;
TBuf<100> msg;
msg.AppendNum(time);
msg.Append(_L(" seconds"));

informationNote->ExecuteLD(msg);

CleanupStack::PopAndDestroy();
```

Note: The last call timer updates every time a call is ended.

Above code displays call duration of last call irrespective of call is on line1 or line2, dialled call or recieved call.

To get call duration of last call on line1 replace KLogsActualLastCallTimer with KLogsLastCallTimerLine1, or with KLogsLastCallTimerLine2 for last call on line2.

The following code displays the total call duration of dialled calls on line1 in seconds :

```
// We need to Query Central Repository.
CRepository* cRepository = CRepository::NewLC( KCRUidLogs);

// Get the time in seconds
TInt time;
User::LeaveIfError( cRepository->Get( KLogsDialledCallsTimerLine1, time ) );

// To display message

CAknInformationNote* informationNote = new (ELeave) CAknInformationNote;
TBuf<100> msg;
msg.AppendNum(time);
msg.Append(_L(" seconds"));

informationNote->ExecuteLD(msg);

CleanupStack::PopAndDestroy();
```

Note: The dialled call timer of line1 increments everytime when call on line1 is ended.

Replace KLogsDialledCallsTimerLine1 with KLogsDialledCallsTimerLine2 to get total dialled calls duration on line2.

The following code displays the total call duration of recieved calls on line1 in seconds :

```
// We need to Query Central Repository.
CRepository* cRepository = CRepository::NewLC( KCRUidLogs);

// Get the time in seconds
TInt time;
User::LeaveIfError( cRepository->Get( KLogsReceivedCallsTimerLine1, time ) );

// To display message

CAknInformationNote* informationNote = new (ELeave) CAknInformationNote;
TBuf<100> msg;
msg.AppendNum(time);
msg.Append(_L(" seconds"));
```

Logs_Timers_API

```
informationNote->ExecuteLD(msg);  
CleanupStack::PopAndDestroy();
```

Note: The received call timer of line1 increments everytime when call on line1 is ended.

Replace KLogsReceivedCallsTimerLine1 with KLogsReceivedCallsTimerLine2 to get total received calls duration on line2.

The following code gives information about whether the settings of logs: "Log duration" and "Show call duration" are enabled or disabled.

```
// We need to Query Central Repository.  
CRepository* cRepository = CRepository::NewLC(KCRUidLogs);  
  
// Get the integer value  
TInt value;  
User::LeaveIfError(cRepository->Get(KLogsLoggingEnabled, value));  
  
// To display message  
  
_LIT(message0, "Disabled");  
_LIT(message1, "Enabled");  
  
CAknInformationNote* informationNote = new (ELeave) CAknInformationNote;  
if (value==0)  
{  
    informationNote->ExecuteLD(message0); //disabled  
}  
else  
{  
    informationNote->ExecuteLD(message1); //enabled  
}  
CleanupStack::PopAndDestroy();
```

Note: Replace KLogsLoggingEnabled with KLogsShowCallDuration for checking "Show call duration" setting.

Example project

<http://wiki.forum.nokia.com/index.php/Image:LogsTimers.zip>