

Error_codes

Reviewer Approved



Global error codes from `e32err.h`. Note that each constant is defined as an integer like this:

```
const TInt KErrNone;
```

Also note that all of these codes indicate an error in the environment, or in user input from which a program may recover.

Error code	Value	Description
<code>KErrNone</code>	0	No error.
<code>KErrNotFound</code>	-1	Item not found.
<code>KErrGeneral</code>	-2	An error that has no specific categorisation.
<code>KErrCancel</code>	-3	Indicates an operation that has been cancelled.
<code>KErrNoMemory</code>	-4	An attempt to allocate memory has failed.
<code>KErrNotSupported</code>	-5	Some functionality is not supported in a given context. There may be many reasons for this; for example, a device may not support some specific behaviour.
<code>KErrArgument</code>	-6	An argument is out of range. A system wide error code indicates an error in the environment, or in user input from which a program may recover.
<code>KErrTotalLossOfPrecision</code>	-7	A calculation has lost precision. This error arises when converting from an internal 96-bit real representation to a <code>TReal32</code> ; the exponent of the internal representation is so small that the 32-bit real cannot contain it.
<code>KErrBadHandle</code>	-8	An invalid handle has been passed. A function involving a resource owned by a server or the kernel has specified an invalid handle.
<code>KErrOverflow</code>	-9	Indicates an overflow in some operation. In the context of mathematical or time/date functions, indicates a calculation that has produced arithmetic overflow exceeding the bounds allowed by the representation. In the context of data transfer, indicates that a buffer has over-filled without being emptied soon enough.
<code>KErrUnderflow</code>	-10	Indicates an underflow in some operation. In the context of mathematical or time/date functions, indicates a calculation that has produced a result smaller than the smallest magnitude of a finite number allowed by the representation.

Error_codes

		In the context of data transfer, indicates that a buffer was under-filled when data was required.
KErrAlreadyExists	-11	An object already exists. An object of some name/type is to be created, but an object of that name/type already exists.
KErrPathNotFound	-12	In the context of file operations, a path was not found.
KErrDied	-13	A handle refers to a thread that has died.
KErrInUse	-14	A requested resource is already in exclusive use.
KErrServerTerminated	-15	Client/server send/receive operation cannot run, because the server has terminated.
KErrServerBusy	-16	A client/server send/receive operation cannot run, because the server is busy handling another request.
KErrCompletion	-17	Indicates that an operation is complete, successfully or otherwise. This code may be used to indicate that some follow on operation can take place. It does not necessarily indicate an error condition.
KErrNotReady	-18	Indicates that a device required by an i/o operation is not ready to start operations. A common reason for returning this code is because a device has not been initialised, or has no power.
KErrUnknown	-19	A device is of unknown type.
KErrCorrupt	-20	Indicates that some media is not formatted properly, or links between sections of it have been corrupted.
KErrAccessDenied	-21	Access to a file or device is denied, because the permissions on the file do not allow the requested operation to be performed or the device is in use.
KErrLocked	-22	An operation cannot be performed, because the part of the file to be read or written is locked.
KErrWrite	-23	During a file write operation, not all the data could be written.
KErrDisMounted	-24	A volume which was to be used for a file system operation has been dismounted.
KErrEof	-25	Indicates that end of file has been reached. Note that <code>RFile::Read()</code> is a higher-level interface. When the end of the file is reached, it returns zero bytes in the destination descriptor, and a <code>KErrNone</code> return value. <code>KErrEof</code> is not used for this purpose; other error conditions are returned only if some other error condition was indicated on the file.
KErrDiskFull	-26	A write operation cannot complete, because the disk is full.
KErrBadDriver	-27	A driver DLL is of the wrong type.

Error_codes

KErrBadName	-28	A file name or other object name does not conform to the required syntax.
KErrCommsLineFail	-29	A communication line has failed.
KErrCommsFrame	-30	A frame error has occurred in a communications operation.
KErrCommsOverrun	-31	An overrun has been detected by a communications driver.
KErrCommsParity	-32	A parity error has occurred in communications.
KErrTimedOut	-33	An operation has timed out.
KErrCouldNotConnect	-34	A session could not connect.
KErrCouldNotDisconnect	-35	A session could not disconnect.
KErrDisconnected	-36	A function could not be executed because the required session was disconnected.
KErrBadLibraryEntryPoint	-37	A library entry point was not of the required type.
KErrBadDescriptor	-38	A non-descriptor parameter was passed by a client interface, when a server expected a descriptor.
KErrAbort	-39	An operation has been aborted.
KErrTooBig	-40	A number was too big.
KErrDivideByZero	-41	A divide-by-zero operation has been attempted.
KErrBadPower	-42	Insufficient power was available to complete an operation.
KErrDirFull	-43	An operation on a directory has failed.
KErrHardwareNotAvailable	-44	An operation cannot be performed because the necessary hardware is not available.
KErrSessionClosed	-45	The completion status when an outstanding client/server message is completed because a shared session has been closed.
KErrPermissionDenied	-46	An operation cannot be performed due to a potential security violation.
KErrExtensionNotSupported	-47	A requested extension function is not supported by the object concerned.
KErrCommsBreak	-48	A break has occurred in a communications operation.

Resources

Download the error codes spreadsheet [here](#)