

## GETTING STARTED

### Installation

Included with the single DLL file **QuickPDFDLL0723.dll** are various header/import files. They provide an easier way to interface with Quick PDF Library. Technical details of the interface are provided here.

### Initializing/releasing the library

All functions in the DLL use the **stdcall** convention. The **QuickPDFCreateLibrary** function must be called to initialize the library. An InstanceID will be returned which must be passed as the first parameter to all the other functions.

When you are finished with the library, call **QuickPDFReleaseLibrary** to release all allocated memory.

### Unlocking the library

Once you have an InstanceID, you should call the **QuickPDFUnlockKey** function, passing it your license key, to unlock the library.

```
int InstanceID;
InstanceID = QuickPDFCreateLibrary();
if (QuickPDFUnlockKey(InstanceID, "your license key") == 1) {
    QuickPDFDrawText(InstanceID, 100, 500, "Hello world");
    QuickPDFSaveToFile(InstanceID, "C:\\Docs\\HelloFromDLL.pdf");
}
QuickPDFReleaseLibrary(InstanceID);
```

## Sending strings to Quick PDF Library

All Quick PDF Library string parameters are defined as PChars, which are pointers to 8-bit null-terminated strings.

If you need to send binary data to Quick PDF Library that may contain null characters, you can ask Quick PDF Library to create a temporary buffer of a certain size.

Use the **QuickPDFCreateBuffer** and **QuickPDFAddToBuffer** functions to create the buffer and fill it with data. The value returned by the **QuickPDFCreateBuffer** function can then be used for any Quick PDF Library string parameter:

```
char * Buffer;
char * Content = ...; // pointer to the data
Buffer = QuickPDFCreateBuffer(10000);
QuickPDFAddToBuffer(InstanceID, Buffer, Content, 10000);
QuickPDFStoreCustomDataFromString(InstanceID,
    "MyData", Buffer, 1, 0);
QuickPDFReleaseBuffer(InstanceID, Buffer);
```

## Receiving strings from Quick PDF Library

Functions that return string data will return a PChar, a pointer to a null terminated string. The memory for this string is contained within the Quick PDF Library instance.

The data in the string should be copied out immediately as the same memory will be used for subsequent calls to the DLL.

Some functions may return data that contains null characters.

To get the length of the returned string, use the **QuickPDFStringResultLength** function:

```
char * Content;
int ContentLength;
Content = QuickPDFRetrieveCustomDataToString(InstanceID,
    "MyData", 1);
ContentLength = QuickPDFStringResultLength(InstanceID);
// copy the data in Content now
```

## Product website

Please visit the Quick PDF Library website for news and information:

**<http://www.quickpdflibrary.com/>**