

Tax Calculator SDK User Manual

1.0 About the Tax Calculator SDK

The Tax Calculator SDK is used to integrate the functionality provided by the Tax Calculator products into 3rd-party applications. This is intended to be used by developers only – as this product is used for programming!

If you've downloaded the SDK, you'll see a .zip archive containing a file named TESDKW.dll – this is library to integrate into your program. It is written using .NET 2.0 and queries our servers via the internet – so in order to function, it needs both .NET 2.0 installed, and have an internet connection. The SDK is exposed to COM, so even if your application is not written in a .NET language, you can still utilize it as COM.

2.0 Using the Tax Calculator SDK

A quick view of the TESDKW class (in a tool such as MS Object Browser) gives us a list of available members:

```
public string AuthorizationCode { set; get; }  
    Member of TESDK.TESDKW
```

```
public void GetRateEffectiveDates(string State, string RateCode, System.DateTime InvoiceDate, out  
System.DateTime BeginDate, out System.DateTime EndDate)  
    Member of TESDK.TESDKW
```

```
public string GetStatesList()  
    Member of TESDK.TESDKW
```

```
public void SearchAddress(string StreetAddr, string State, string City, System.DateTime InvoiceDate, out string  
RateCode, out decimal Rate, out string Message)  
    Member of TESDK.TESDKW
```

```
public void SearchAddress(string StreetAddr, string State, string Zip, string Zip4, System.DateTime InvoiceDate, out  
string RateCode, out decimal Rate, out string Message)  
    Member of TESDK.TESDKW
```

```
public TESDKW()  
    Member of TESDK.TESDKW
```

AuthorizationCode is the code sent to you at the time of purchase. If you're simply running the demo, please use the value of "DEMO". As with all of our SST Solutions, the DEMO code will return only one rate.

GetRateEffectiveDates() outputs the date range in which the rate code passed for the given invoice date is valid. This is useful for updating any codes in your system to ensure they're up-to-date.

GetStatesList() outputs a comma-delimited list of 2-character state identifiers available to you.

SearchAddress() outputs the rate code, rate, and special message for the given information. This method is overloaded to allow search based on Zip or City Name. It should be noted that Zip (plus 4 if known) will typically provide a more accurate result in a much quicker time than City Name.

TESDKW() – the constructor for the TESDKW object.

3.0 Examples

C# Console Application

```
using System;
using System.Collections.Generic;
using System.Text;
using TESDK;

namespace ConsoleApplication1
{
    class Program
    {
        static void Main(string[] args)
        {
            string RateCode = "";
            decimal Rate = 0;
            string Message = "";

            TESDKW objTaxEngine = new TESDKW();
            objTaxEngine.AuthorizationCode = "DEMO";
            objTaxEngine.SearchAddress
            (
                "210 E 13th St",
                "WA",
                "98660",
                "",
                DateTime.Now,
                out RateCode,
                out Rate,
                out Message
            );

            Console.WriteLine
            (
                RateCode + "\n" +
                Rate.ToString() + "\n" +
                Message
            );
            Console.ReadKey();
        }
    }
}
```

Visual Basic 6.0

The first thing that needs to happen for VB (and other non-.NET languages) is to register the dll for use. You do this using the REGASM command-line tool provided with .NET – an example: regasm c:\temp\tesdkw.dll /tlb:c:\temp\tesdkw.tlb

Once the library is registered, you can add it as a reference to your project. The following code example will run a query and display a message box with the results.

```
Dim objTaxEngine As TESDKW.TESDKW
Dim RateCode As String
Dim Rate As Currency
Dim Message As String

Set objTaxEngine = New TESDKW.TESDKW

objTaxEngine.AuthorizationCode = "DEMO"

Call objTaxEngine.SearchAddress _
    ( _
    "6003 NE 405th St", _
    "WA", _
    "98660", _
    "", _
    DateTime.Now, _
    RateCode, _
    Rate, _
    Message _
    )

MsgBox (RateCode & vbNewLine & Rate & vbNewLine & Message)
```