

Api Microsip



Fecha de modificación:
19 de Diciembre 2016

Autor:
Carlos Enrique Loza

| | |
|-----------------------------|----------------------------------------------------------------------|
| Nombre del Podcast: | Podcast 2016-43 Api Microsip |
| Tema: | API Microsip |
| Objetivos: | Mostrar ejemplos de uso de la api para aplicaciones en Delphi y .Net |
| URL de los archivos: | |

Opciones de "A Cuadro": V=Video; M= Monitor; P=Powerpoint

| Desarrollo | | | |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------|----------|--------|
| Texto | Acción | A Cuadro | Escena |
| Que tal, bienvenidos al podcast técnico de la API Microsip. | A cuadro | V | 1 |
| La API por sus siglas en inglés, es una Interfaz para Programación de Aplicaciones, cuya función es permitir que distintas aplicaciones puedan trabajar entre sí. Nuestra API, permite a los desarrolladores de aplicaciones, poder obtener y guardar información a una base de datos Microsip. | A cuadro | V | 2 |
| Esta herramienta, que se puede descargar desde la sección de Versiones en nuestra página de soporte, cuenta con funciones para insertar documentos en los módulos de Compras, Cuentas por cobrar, Inventarios y Ventas, además de varias funciones para consultar, ejecutar procedimientos y guardar información. | Página de soporte | M | 3 |
| Incluye archivos con extensión .bas para programación en visual basic, archivos de extensión .cs para .Net y archivos .pas para desarrollos en Delphi. Si se utiliza cualquier otro lenguaje, todas las | Archivos de la API incluidos en el ZIP | M | 4 |

Api Microsip



Fecha de modificación:
19 de Diciembre 2016

Autor:
Carlos Enrique Loza

| | | | |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------|---|----|
| funciones están disponibles en el archivo ApiMicrosip.dll. | | | |
| Incluimos también las guías de referencia donde se explican todos los detalles técnicos de las funciones disponibles, sus parámetros necesarios y los valores de retorno para validar y obtener los resultados que estemos buscando. | Archivos de documentos incluidos en el zip de la API | M | 5 |
| Para ejemplificar el uso de la API, vamos a mostrar una aplicación dividida en dos secciones, en la primera, tenemos un ejemplo de conexión a base de datos donde simplemente solicitamos el nombre de usuario, la contraseña y la ubicación de la base de datos. | Abriendo la aplicación de prueba de la API y mostrar pestaña de conexión | M | 6 |
| En la segunda sección, tenemos los datos mínimos necesarios para poder crear una factura, como son la fecha, el cliente, descuento y una lista de artículos a incluir en la factura. | Mostrando aplicación de prueba, primer pestaña | M | 7 |
| Estos datos varían dependiendo del tipo de documento que se desee crear en Microsip, por lo que es importante revisar la documentación de la función a utilizar para conocer los parámetros necesarios para poder utilizarla. | Mostrando documento de texto donde se definen los parámetros | M | 8 |
| Una vez que tengamos el diseño de la aplicación, es necesario agregar los archivos con los recursos de la API dentro de nuestro proyecto. Para el caso de Delphi, vamos a agregar el archivo ApiMspBasicaExt.pas para tener disponibles las funciones de conexión y el archivo ApiMpVentasExt.pas para generar documentos en el módulo de Ventas. | Agregar archivos .pas al proyecto de delphi | M | 9 |
| En la prueba de conexión, vamos a | Mostrando código de | M | 10 |

Api Microsip



Fecha de modificación:
19 de Diciembre 2016

Autor:
Carlos Enrique Loza

| | | | |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------|---|----|
| establecer las variables correspondientes a una base de datos y a una transacción, que son las variables que se pueden utilizar para cualquiera de las funciones incluidas en la API. | aplicación en prueba de conexión | | |
| La función importante en este ejemplo, es "DBConnect", esta función intenta la conexión a la base de datos seleccionada, con el usuario y contraseña indicados en los cuadros de texto. | Mostrar dbconnect | M | 11 |
| En este punto, es importante mencionar que se debe de tomar en cuenta que la API está diseñada para manejar texto en formato ANSI, por lo que si el lenguaje de programación que estás utilizando, maneja textos en formato UNICODE, estos deben de convertirse a un formato que pueda reconocer la API. | A cuadro | V | 12 |
| En las siguientes líneas de código se hace una prueba para aplicar una transacción con la instrucción "commit" y finalmente nos desconectamos de la base de datos. En este código, podemos ver cómo se obtiene el número de retorno de cada función para verificar si se han presentado errores. | Mostrando funciones commit y disconnect | M | 13 |
| Cada uno de los valores que puede regresar una función, están indicados en las guías de referencia, esta documentación nos sirve para decidir qué hacer dependiendo del valor de retorno de la función que estemos utilizando. | Mostrar función de api message para ver los diferentes valores de retorno y su mensaje | M | 14 |
| Para el ejemplo de la creación de una factura, diseñamos una pantalla únicamente con los datos que vamos a necesitar para las funciones que vamos a | Monitor mostrando primer pestaña donde se captura la factura | M | 15 |

Api Microsip



Fecha de modificación:
19 de Diciembre 2016

Autor:
Carlos Enrique Loza

| | | | |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------|---|----|
| utilizar, los cuales, para el caso de la factura, son la fecha, el cliente, descuento extra y una tabla, donde es posible capturar el artículo, las unidades, el precio, descuento y las notas, que correspondan a cada una de las partidas que vamos a incluir en la factura. | | | |
| Como funcionalidad adicional, tenemos el ejemplo de la búsqueda de clientes y artículos a través de una consulta SQL, utilizando siempre las funciones disponibles en la API. Para realizar la consulta, es necesario primero establecer una transacción, definir un manejador para la consulta y una vez iniciada la transacción, con la función SQLQry, establecemos la consulta que se va a ejecutar. | Mostrar función para buscar con SQL | M | 16 |
| En caso de que nuestra consulta tenga parámetros, será necesario establecer los valores de estos parámetros con la función SQLSetparam. | Mostrar set param en pantalla | M | 17 |
| Cuando la consulta esté lista, la ejecutamos con la instrucción SQLExecQuery y obtenemos los valores que estamos buscando con la función SQLGetField. | Mostrar exec y getfield | M | 18 |
| Tanto la función SQLSetParam, como la función SQLGetField, se pueden manejar con diferentes tipos de datos, por lo que es importante revisar la documentación de la API para conocer con qué tipo de dato podemos obtener los resultados de la consulta. | Mostrar documento de guía con las diferentes opciones de getfieldas | M | 19 |
| Como último paso, simplemente cerramos la consulta que acabamos de realizar con | Mostrar código sqlclose | M | 20 |

Api Microsip



El Software Confiable de México

Fecha de modificación:
19 de Diciembre 2016

Autor:
Carlos Enrique Loza

| | | | |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------|---|----|
| la función SQLClose. | | | |
| Una vez que hemos capturado los datos necesarios para la factura, | Mostrar datos ya capturados en la primer pestaña de la aplicación | M | 21 |
| Primero creamos el encabezado de la factura con la función NuevaFactura, muchos de los parámetros necesarios para esta función, se pueden establecer con valores predeterminados, o dicho de otra forma, con valores que el sistema toma de forma automática. | Mostrar código de nueva factura | M | 22 |
| Todas las opciones de estos valores automáticos, están definidas en las guías de referencias de cada función. | A cuadro | V | 23 |
| En el siguiente paso, vamos a recorrer la tabla donde capturamos los artículos, para ir agregando partidas en la factura utilizando la función RenglonFactura. | Mostrar función renglón factura | M | 24 |
| Finalmente, con la función AplicaFactura, estos datos se van a crear en Microsip, por lo que es muy importante revisar el código de retorno de esta función para saber si la aplicación se realizó correctamente. | Mostrar código de función aplicafactura | M | 25 |
| Si todo salió bien, el documento generado desde nuestra aplicación, se mostrará en Microsip como cualquier otro documento. | Mostrar documento generado en microsip | M | 26 |
| Por último, debemos recordar que al finalizar nuestro trabajo con la API de Microsip, es muy importante liberar los recursos del equipo utilizando la función LiberarRecursos. | Mostrar código para liberarrecursos | M | 27 |
| Los archivos de estos ejemplos, están disponibles para descargar en nuestra | A cuadro | V | 28 |

Api Microsip



Fecha de modificación:
19 de Diciembre 2016

Autor:
Carlos Enrique Loza

página de soporte, esperamos que sean de utilidad. Muchas gracias por su atención y recuerden seguirnos en nuestras redes sociales Facebook, twitter y youtube. Nos vemos en el siguiente podcast técnico de Microsip.