

Actividad | 2 | Conexión y Tablas

Lenguaje de Programación II

Ingeniería en Desarrollo de Software



TUTOR: Félix Acosta Hernández

ALUMNO: Juan Carlos Legorreta Ramos

FECHA: 06-12-2024

Índice

Introducción.....	1
Descripción.....	1
Justificación.....	2
Desarrollo.....	2
Conexión.....	3
Tablas.....	6
Codigo.....	7
Conclusión.....	10
Referencias.....	11

Introducción

Dentro de esta segunda actividad se estará mostrando la creación y gestión de la base de datos de una empresa la cual con ayuda de la herramienta de gestión de bases de datos siendo SQLServer En la cual con la ayuda de la actividad pasada se implementarán varias tablas que se dan claves como lo son el centro de trabajo empleados y directivos las cuales tendrán la función de almacenar información de suma importancia para la empresa como información personal detalles de los senderos y los directivos responsables de supervisión. De igual manera se está especificando que hay que crear una estructura ésta incluirá claves primarias y foráneas lo que permitirá que se asegure la integridad de los datos relacionados entre varias tablas y así hola una consulta eficiente para poder acceder a la información ya almacenada ya que es de suma importancia que una base de datos esté bien estructurada para un entorno empresarial permitiendo hacer una gestión organizada en forma precisa.

Igual nos enfocaremos en una conexión a la base de datos utilizando visual studio este tipo de conexión nos permiten que las aplicaciones puedan interactuar entre sí utilizando esta base evitando así consultas y obteniendo información en tiempo real.

Descripción

Como ya mencionábamos estaremos quedando una base de datos para poder gestionar información relacionado a los empleados y centros de trabajo de una empresa estaremos trabajando en 3 tablas principales siendo centro de trabajo empleados y directivos.

Otro punto en el cual se centra es la utilización de la herramienta visual studio siendo éste un entorno de desarrollo para crear un programa en el lenguaje C++ que a su vez tendría conexión directa con la base de datos SQLServer de la actividad anterior, Adicional a esto estaremos

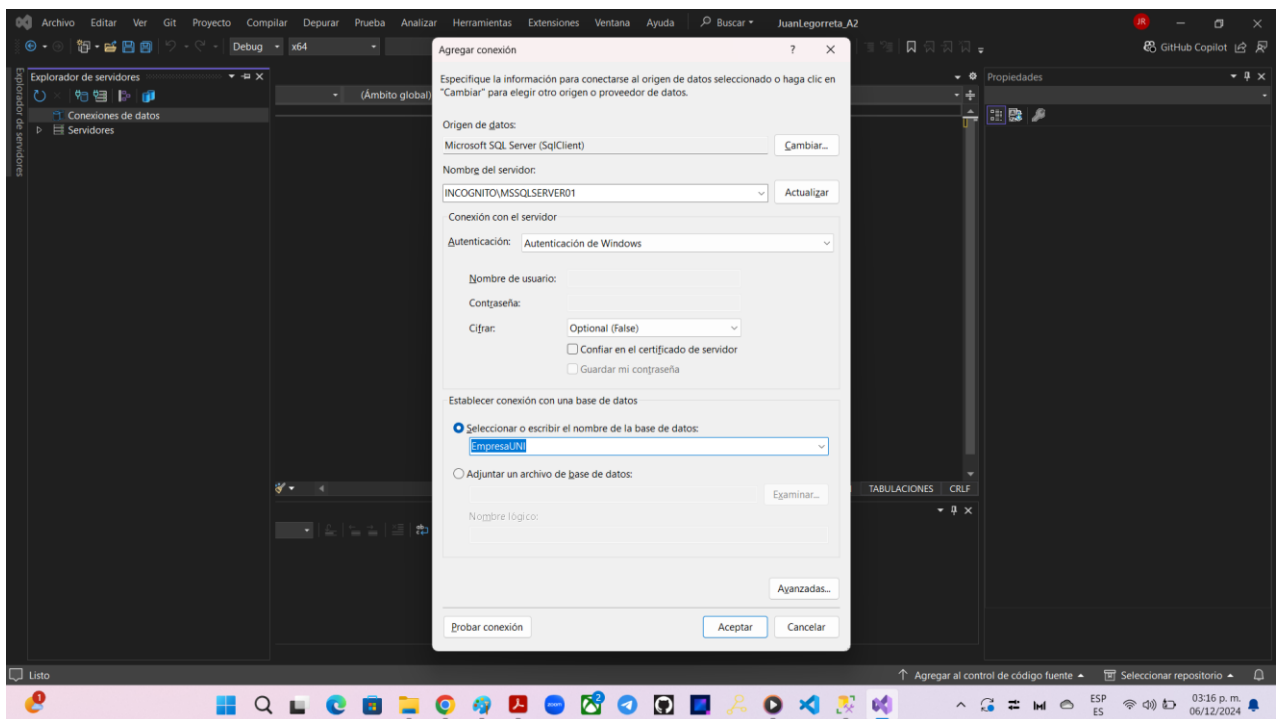
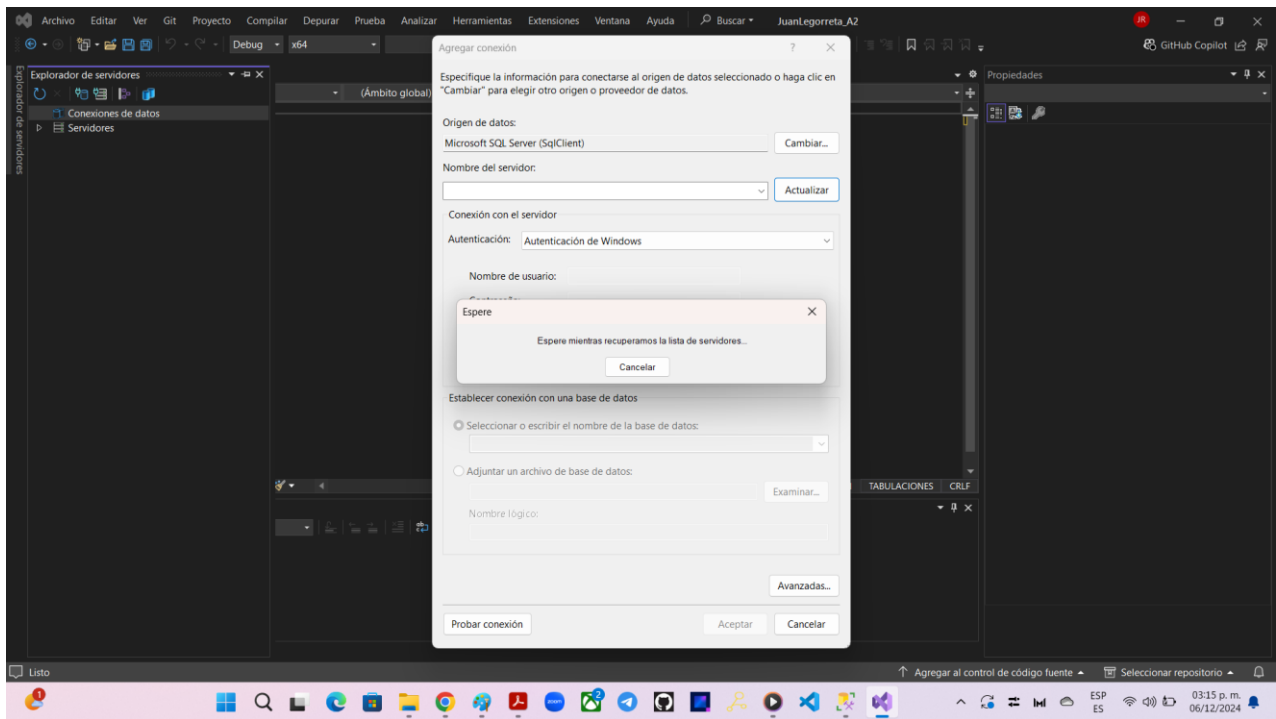
utilizando una conexión ODBC la cual tendrá una función muy importante ya que nos permitirá que la aplicación puede interactuar con esta base de datos lo que nos permitirá realizar operaciones básicas, hoy es por esto que el integrada del lenguaje de programación C++ permite desarrollar aplicaciones que permitan gestionar información de manera más eficiente y también automatizada esto especialmente como lo vimos va la gestión de datos en las empresas.

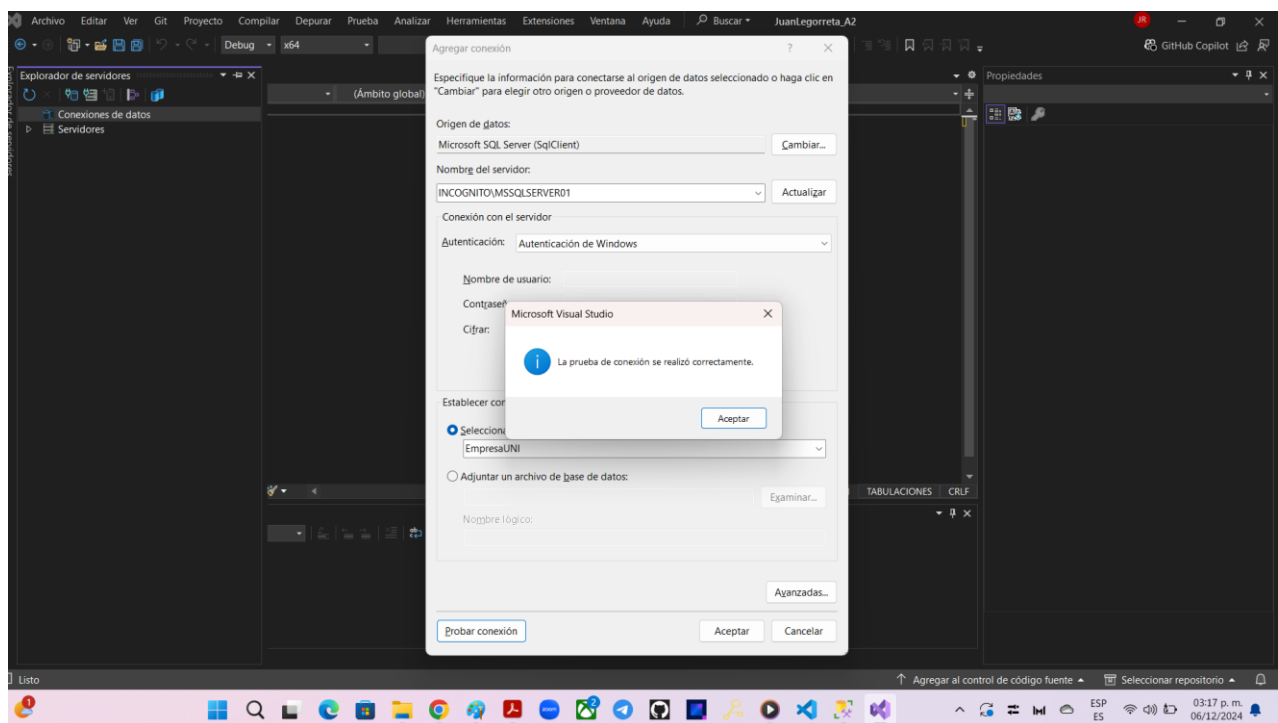
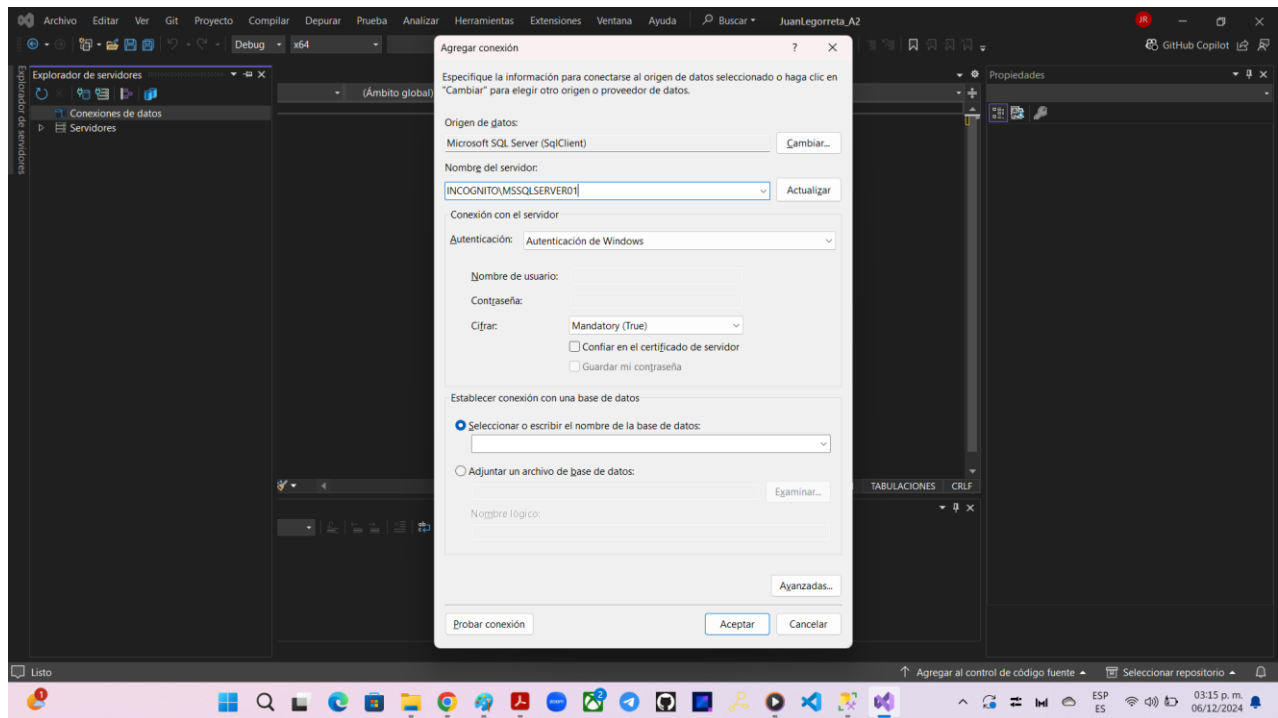
Justificación

El implementar esta solución en la actividad es esencial para las empresas Ya que una base de datos nos proporciona una estructura que esté mejor organizada y optimizada para poder almacenar grandes volúmenes de información en una forma más eficiente ya que al utilizar la herramienta SQLServer como herramienta para la gestión de esta base de datos nos ayuda a que los datos sean almacenados de una forma más accesible segura y fácil de manejar a comparación de utilizar archivos o estructuras que no estén bien diseñadas. Así como integra el lenguaje de C++ con estas bases de datos a través de una conexión ODBC nos facilita que la aplicación pueda interactuar directamente con esta base de datos para realizar tareas como actualización eliminación y consulta de datos, y es de suma importancia de tener esta capacidad en los sistemas que requieren actualizaciones constantes o un acceso más rápido a la información esto en tiempo real como bien vimos para la gestión del personal de una empresa, sin olvidar qué estás aplicaciones al interactuar con esas bases de datos pueden automatizar las tareas repetitivas lo que minimiza demasiado el riesgo de un error humano y mejora la eficiencia operativa.

Desarrollo

Conexión





En las imágenes podemos observar la conexión de visual studio con la base de datos perteneciente a la herramienta SQLServer.

```
1 // JuanLegerreta_A2
2 #include <windows.h>
3 #include <reg.h>
4 #include <regapi.h>
5 #include <stdio.h>
6 #include <string.h>
7 #include <stdlib.h>
8 #include <conio.h>
9 #include <time.h>
10 #include <ctype.h>
11 #include <math.h>
12 #include <float.h>
13 #include <limits.h>
14 #include <locale.h>
15 #include <memory.h>
16 #include <stdarg.h>
17 #include <stdbool.h>
18 #include <stdint.h>
19 #include <inttypes.h>
20 #include <time_t.h>
21 #include <wchar_t.h>
22 #include <wctype.h>
23 #include <wctype.h>
24 #include <wctype.h>
25 #include <wctype.h>
26 #include <wctype.h>
27 #include <wctype.h>
28 #include <wctype.h>
29 #include <wctype.h>
30 #include <wctype.h>
31 #include <wctype.h>
32 #include <wctype.h>
33 #include <wctype.h>
34 #include <wctype.h>
35 #include <wctype.h>
36 #include <wctype.h>
37 #include <wctype.h>
38 #include <wctype.h>
39 #include <wctype.h>
40 #include <wctype.h>
41 #include <wctype.h>
42 #include <wctype.h>
43 #include <wctype.h>
44 #include <wctype.h>
45 #include <wctype.h>
46 #include <wctype.h>
47 #include <wctype.h>
48 #include <wctype.h>
49 #include <wctype.h>
50 #include <wctype.h>
51 #include <wctype.h>
52 #include <wctype.h>
53 #include <wctype.h>
54 #include <wctype.h>
55 #include <wctype.h>
56 #include <wctype.h>
57 #include <wctype.h>
58 #include <wctype.h>
59 #include <wctype.h>
60 #include <wctype.h>
61 #include <wctype.h>
62 #include <wctype.h>
63 #include <wctype.h>
64 #include <wctype.h>
65 #include <wctype.h>
66 #include <wctype.h>
67 #include <wctype.h>
68 #include <wctype.h>
69 #include <wctype.h>
70 #include <wctype.h>
71 #include <wctype.h>
72 #include <wctype.h>
73 #include <wctype.h>
74 #include <wctype.h>
75 #include <wctype.h>
76 #include <wctype.h>
77 #include <wctype.h>
78 #include <wctype.h>
79 #include <wctype.h>
80 #include <wctype.h>
81 #include <wctype.h>
82 #include <wctype.h>
83 #include <wctype.h>
84 #include <wctype.h>
85 #include <wctype.h>
86 #include <wctype.h>
87 #include <wctype.h>
88 #include <wctype.h>
89 #include <wctype.h>
90 #include <wctype.h>
91 #include <wctype.h>
92 #include <wctype.h>
93 #include <wctype.h>
94 #include <wctype.h>
95 #include <wctype.h>
96 #include <wctype.h>
97 #include <wctype.h>
98 #include <wctype.h>
99 #include <wctype.h>
100 #include <wctype.h>
```

Salida

Mostrar salida de: Compilación

Compilación iniciada a las 06:05 p. m.

1>----- Operación Compilar iniciada: Proyecto: JuanLegerreta_A2, configuración: Debug x64 -----

1>JuanLegerreta_A2.cpp

1>JuanLegerreta_A2.vcxproj -> C:\Users\borre\Desktop\project\JuanLegerreta_A2\x64\Debug\JuanLegerreta_A2.exe

----- Compilación: 1 correcto, 0 erróneo, 0 actualizado, 0 omitido -----

----- Compilar completado a las 06:05 p. m. y tardó 02.754 segundos -----

Lista de errores: Salida

Compilación correcta

```
1 // JuanLegerreta_A2
2 #include <windows.h>
3 #include <reg.h>
4 #include <regapi.h>
5 #include <stdio.h>
6 #include <string.h>
7 #include <stdlib.h>
8 #include <conio.h>
9 #include <time.h>
10 #include <ctype.h>
11 #include <math.h>
12 #include <float.h>
13 #include <limits.h>
14 #include <locale.h>
15 #include <memory.h>
16 #include <stdarg.h>
17 #include <stdbool.h>
18 #include <stdint.h>
19 #include <inttypes.h>
20 #include <time_t.h>
21 #include <wchar_t.h>
22 #include <wctype.h>
23 #include <wctype.h>
24 #include <wctype.h>
25 #include <wctype.h>
26 #include <wctype.h>
27 #include <wctype.h>
28 #include <wctype.h>
29 #include <wctype.h>
30 #include <wctype.h>
31 #include <wctype.h>
32 #include <wctype.h>
33 #include <wctype.h>
34 #include <wctype.h>
35 #include <wctype.h>
36 #include <wctype.h>
37 #include <wctype.h>
38 #include <wctype.h>
39 #include <wctype.h>
40 #include <wctype.h>
41 #include <wctype.h>
42 #include <wctype.h>
43 #include <wctype.h>
44 #include <wctype.h>
45 #include <wctype.h>
46 #include <wctype.h>
47 #include <wctype.h>
48 #include <wctype.h>
49 #include <wctype.h>
50 #include <wctype.h>
51 #include <wctype.h>
52 #include <wctype.h>
53 #include <wctype.h>
54 #include <wctype.h>
55 #include <wctype.h>
56 #include <wctype.h>
57 #include <wctype.h>
58 #include <wctype.h>
59 #include <wctype.h>
60 #include <wctype.h>
61 #include <wctype.h>
62 #include <wctype.h>
63 #include <wctype.h>
64 #include <wctype.h>
65 #include <wctype.h>
66 #include <wctype.h>
67 #include <wctype.h>
68 #include <wctype.h>
69 #include <wctype.h>
70 #include <wctype.h>
71 #include <wctype.h>
72 #include <wctype.h>
73 #include <wctype.h>
74 #include <wctype.h>
75 #include <wctype.h>
76 #include <wctype.h>
77 #include <wctype.h>
78 #include <wctype.h>
79 #include <wctype.h>
80 #include <wctype.h>
81 #include <wctype.h>
82 #include <wctype.h>
83 #include <wctype.h>
84 #include <wctype.h>
85 #include <wctype.h>
86 #include <wctype.h>
87 #include <wctype.h>
88 #include <wctype.h>
89 #include <wctype.h>
90 #include <wctype.h>
91 #include <wctype.h>
92 #include <wctype.h>
93 #include <wctype.h>
94 #include <wctype.h>
95 #include <wctype.h>
96 #include <wctype.h>
97 #include <wctype.h>
98 #include <wctype.h>
99 #include <wctype.h>
100 #include <wctype.h>
```

Conectado a la base de datos exitosamente.

C:\Users\borre\Desktop\project\JuanLegerreta_A2\x64\Debug\JuanLegerreta_A2.exe (proceso 15468) se cerró con el código 0 (0x0).

Presione cualquier tecla para cerrar esta ventana. . .

Salida

Mostrar salida de: Compilación

Compilación iniciada a las 06:05 p. m.

1>----- Operación Compilar iniciada: Proyecto: JuanLegerreta_A2, configuración: Debug x64 -----

1>JuanLegerreta_A2.cpp

1>JuanLegerreta_A2.vcxproj -> C:\Users\borre\Desktop\project\JuanLegerreta_A2\x64\Debug\JuanLegerreta_A2.exe

----- Compilación: 1 correcto, 0 erróneo, 0 actualizado, 0 omitido -----

----- Compilar completado a las 06:06 p. m. y tardó 01.274 segundos -----

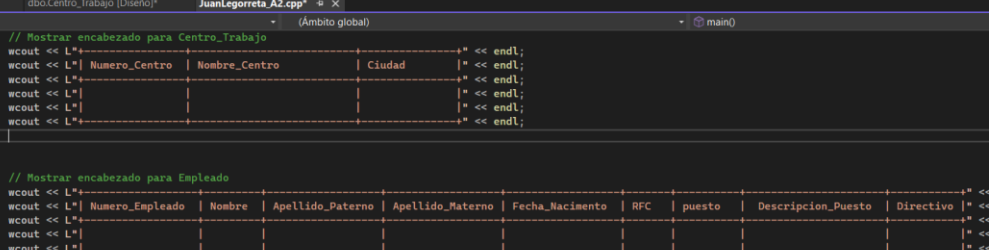
Lista de errores: Salida

Compilación correcta

En esas imágenes se ve el proceso de conexión de visual studio con la base de datos para eso después de las librerías se establece una conexión con la base utilizando ODBC para crear un gestor de un tono seguido de configurar la versión de éste para poder proseguir con una gestión de

conexión para establecer la conexión con el servidor de la base de datos Para poder proseguir se intenta conectar con el servidor utilizando los credenciales proporcionadas y así me edifica que la conexión fue José exitosa o haya fallado.

Tablas



```
25 // Mostrar encabezado para Centro_Trabajo
26 wcout << L"*" << endl;
27 wcout << L"*| Numero_Centro | Nombre_Centro | Ciudad |" << endl;
28 wcout << L"*" << endl;
29 wcout << L"*| | | |" << endl;
30 wcout << L"*" << endl;
31 wcout << L"*" << endl;
32
33
34
35 // Mostrar encabezado para Empleado
36 wcout << L"*" << endl;
37 wcout << L"*| Numero_Empleado | Nombre | Apellido_Paterno | Apellido_Materno | Fecha_Nacimiento | RFC | puesto | Descripción_Puesto | Directivo |" << endl;
38 wcout << L"*" << endl;
39 wcout << L"*| | | | | | | | | |" << endl;
40 wcout << L"*" << endl;
41 wcout << L"*" << endl;
42
43 // Mostrar encabezado para Directivo
44 wcout << L"*" << endl;
45 wcout << L"*| Numero_Empleado | Centro_Supervisado | Combustible |" << endl;
46 wcout << L"*" << endl;
47 wcout << L"*| | | |" << endl;
48 wcout << L"*" << endl;
49 wcout << L"*" << endl;
50 wcout << L"*" << endl;
51
```

100 % No se encontraron problemas. Línea: 32 Carácter: 9 SPC CRLF

Salida

Mostrar salida de: Compilación

Compilación iniciada a las 06:06 p. m....

1>----- Operación Compilador Iniciada: Proyecto: JuanLegerreta_A2, configuración: Debug x64 -----

1>JuanLegerreta_A2.cpp

1>JuanLegerreta_A2.vcxproj -> C:\Users\borre\Desktop\project\JuanLegerreta_A2\x64\Debug\JuanLegerreta_A2.exe

----- Compilación: 1 correcto, 0 erróneo, 0 actualizado, 0 omitido -----

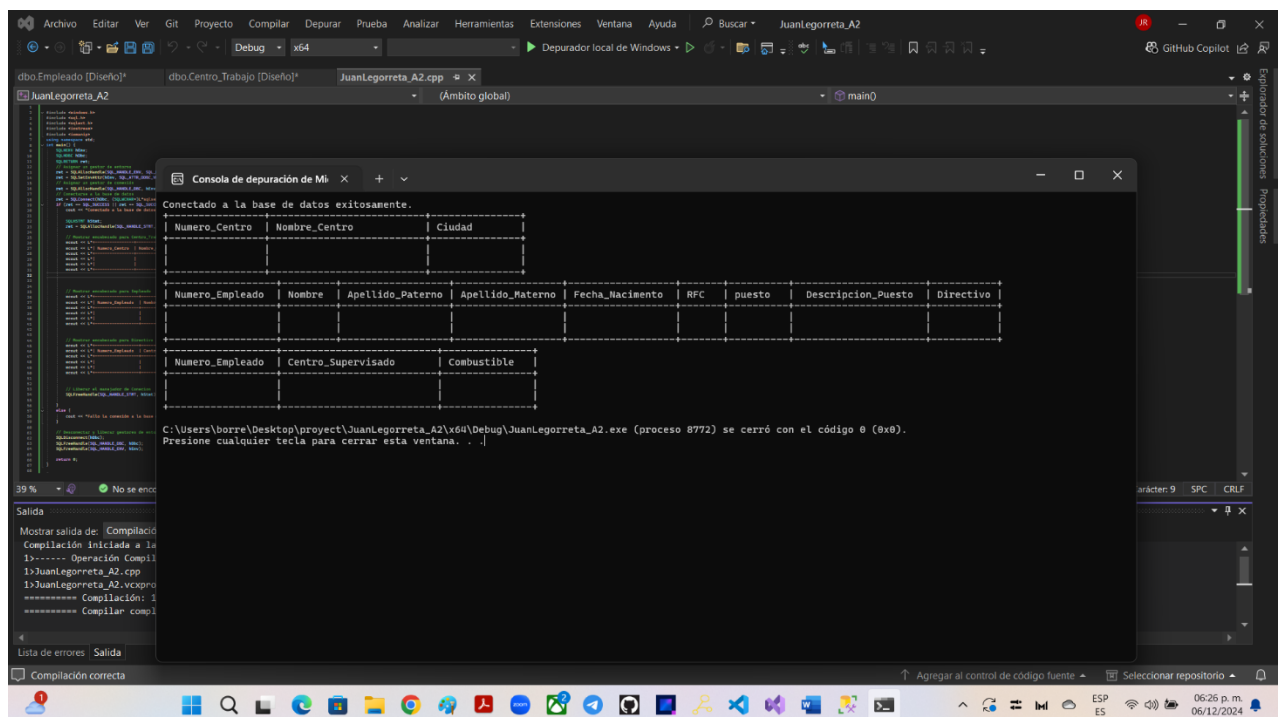
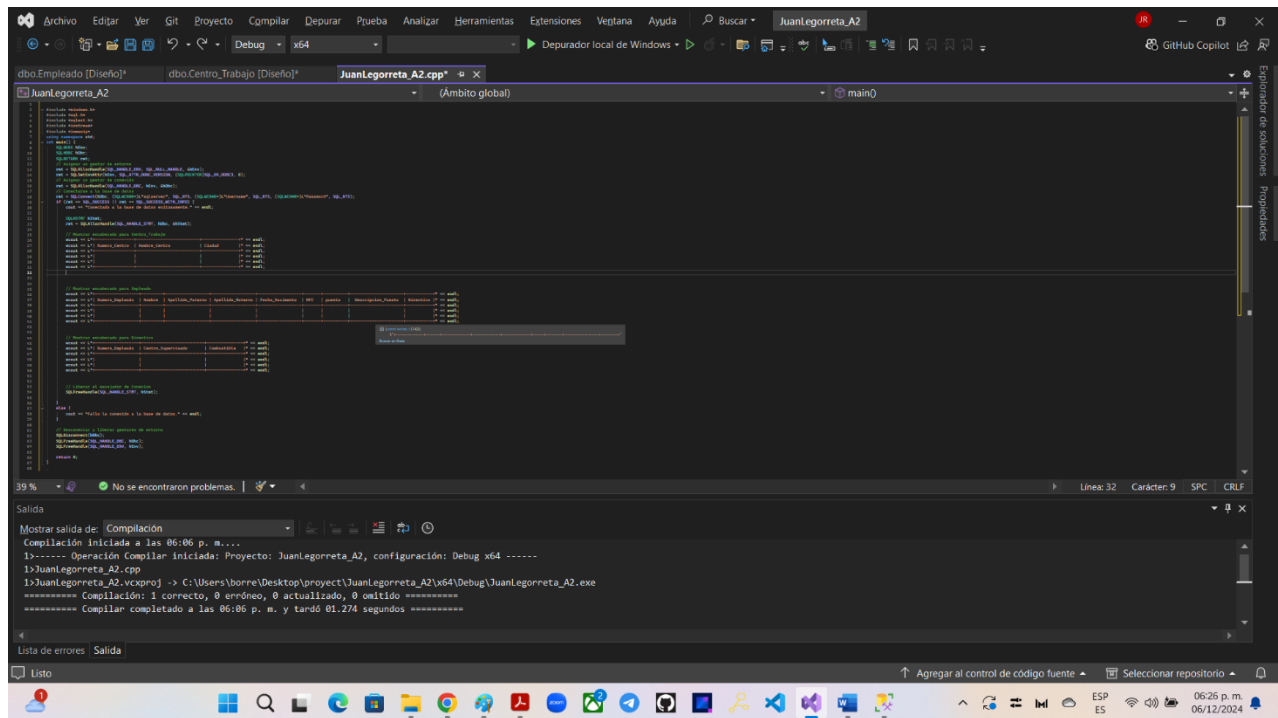
----- Compilar completado a las 06:06 p. m. y tardó 01.274 segundos -----

Lista de errores: Salida

Listo

En esta imagen se muestra el código de los encabezados de las tablas siendo centro de trabajo empleado y directivo usando el comando wcout ya que este código nos permitirá imprimir los encabezados y como las tablas están vacías no se verán los registros debajo de los encabezados.

Código



Dentro de estas a mi imagines se nos muestra el mensaje desconectado a la base de datos exitosamente lo que nos dice que la conexión fue correcta posteriormente simplemente a las tablas vacías de centros de trabajo empleados y directivos.

En el código podemos ver que exclusiva la función `SQLConnect` para poder conectarse al servidor de la base de datos de sql server con las credenciales proporcionadas si la conexión es correcta sociedad con la creación de las tablas.

A continuación os traemos el código completo que se utilizó en la actividad:

```
#include <windows.h>
#include <sql.h>
#include <sqlext.h>
#include <iostream>
#include <iomanip>
using namespace std;
int main() {
    SQLHENV hEnv;
    SQLHDBC hDbc;
    SQLRETURN ret;

    // Asignar un gestor de entorno
    ret = SQLAllocHandle(SQL_HANDLE_ENV, SQL_NULL_HANDLE, &hEnv);
    ret = SQLSetEnvAttr(hEnv, SQL_ATTR_ODBC_VERSION, (SQLPOINTER)SQL_OV_ODBC3, 0);

    // Asignar un gestor de conexión
    ret = SQLAllocHandle(SQL_HANDLE_DBC, hEnv, &hDbc);

    // Conectarse a la base de datos
    ret = SQLConnect(hDbc, (SQLWCHAR*)L"sqlserver", SQL_NTS, (SQLWCHAR*)L"Username", SQL_NTS, (SQLWCHAR*)L"Password",
SQL_NTS);

    if (ret == SQL_SUCCESS || ret == SQL_SUCCESS_WITH_INFO) {
        cout << "Conectado a la base de datos exitosamente." << endl;

        SQLHSTMT hStmt;
        ret = SQLAllocHandle(SQL_HANDLE_STMT, hDbc, &hStmt);

        // Mostrar encabezado para Centro_Trabajo
        wcout << L"+-----+-----+-----+" << endl;
        wcout << L"| Numero_Centro | Nombre_Centro | Ciudad |" << endl;
        wcout << L"+-----+-----+-----+" << endl;
        wcout << L"| | |" << endl;
```

```

wcout << L" | | |" << endl;
wcout << L"+-----+-----+-----+" << endl;

// Mostrar encabezado para Empleado
wcout << L"+-----+-----+-----+-----+-----+-----+-----+" <<
endl;
wcout << L"| Numero_Empleado | Nombre | Apellido_Paterno | Apellido_Materno | Fecha_Nacimiento | RFC | puesto |
Descripcion_Puesto | Directivo |" << endl;
wcout << L"+-----+-----+-----+-----+-----+-----+-----+" <<
endl;
wcout << L" | | | | | | | |" << endl;
wcout << L" | | | | | | | |" << endl;
wcout << L"+-----+-----+-----+-----+-----+-----+-----+" <<
endl;

// Mostrar encabezado para Directivo
wcout << L"+-----+-----+-----+" << endl;
wcout << L"| Numero_Empleado | Centro_Supervisado | Combustible |" << endl;
wcout << L"+-----+-----+-----+" << endl;
wcout << L" | | |" << endl;
wcout << L" | | |" << endl;
wcout << L"+-----+-----+-----+" << endl;

// Liberar el manejador de Conexion
SQLFreeHandle(SQL_HANDLE_STMT, hStmt);

}
else {
    cout << "Fallo la conexión a la base de datos." << endl;
}

// Desconectar y liberar gestores de entorno
SQLDisconnect(hDbc);
SQLFreeHandle(SQL_HANDLE_DBC, hDbc);
SQLFreeHandle(SQL_HANDLE_ENV, hEnv);

```

```
return 0;  
}
```

Conclusión

En la actividad presente llega a tener demasiada relevancia tanto en el ámbito laboral como en la vida cotidiana ya que la creación y gestión de una base de datos utilizando el lenguaje C++ y SQLServer Ya que en el ámbito profesional la habilidad para poder gestionar los datos de una forma eficiente son fundamentales para el desarrollo de aplicaciones e incluso sistemas empresariales con la actividad nos da una comprensión más profunda de cómo se establece una conexión entre aplicaciones y una base de datos para poder así interactuar de una manera más eficiente con los datos almacenados hola esto ya que también en el mundo actual donde muchas organizaciones gestionan grandes cantidades de datos contar con una base bien diseñada y gestionada es clave para el éxito en cualquier empresa es por eso que esta habilidad se convierte muy útil en el campo laboral para los que te bajan en desarrollo de software administración de empresas como análisis de datos y gestión de las mismas bases todo esto al poder crear mantener consulta estos datos permiten optimizar procesos y garantizar la integridad de los mismos. Al mismo modo afecta en la vida cotidiana el poder entender cómo es que funciona una base de datos nos puede ayudar a desarrollar aplicaciones personales como en el caso de gestión financiera o cualquier otra forma donde necesitamos almacenar datos y organizarlos de una forma más avanzada.

Entendiendo esto que una base de datos son componentes que son muy fundamentales en las estructuras tecnológicas de hoy en día y el poder entender la gestión y creación se vuelve una habilidad muy valiosa en el mundo digital en el que actualmente vivimos.

Referencias

Luisito Coding. (2017, 2 julio). *Como conectar Visual Studio con SQL Server* [Video]. YouTube.

<https://www.youtube.com/watch?v=CmWleAx4CFw>