# Programming Language Extension Using Structured Query Language for Database Access

Chapman Eze Nnadozie

Abstract-Relational databases constitute a very vital tool for the effective management and administration of both personal and organizational data. Data access ranges from a single user database management software to a more complex distributed server system. This paper intends to appraise the use a programming language extension like structured query language (SQL) to establish links to a relational database (Microsoft Access 2013) using Visual C++ 9 programming language environment. The methodology used involves the creation of tables to form a database using Microsoft Access 2013, which is Object Linking and Embedding (OLE) database compliant. The SQL command is used to query the tables in the database for easy extraction of expected records inside the visual C++ environment. The findings of this paper reveal that records can easily be accessed and manipulated to filter exactly what the user wants, such as retrieval of records with specified criteria, updating of records, and deletion of part or the whole records in a table.

*Keywords*—Data access, database, database management system, OLE, programming language, records, relational database, software, SQL, table.

#### I. INTRODUCTION

**E**VERY institution all over the world makes use of one or more databases that is created using Database Management System (DBMS) software to meet individual or institutional needs. Software such as Microsoft Excel, Microsoft Access, and Microsoft Visual FoxPro are used to create a single user application, while software such as Oracle and SQL Server are used if the user is building applications that need to access large, shared databases. All DBMS software supports the use of SQL as a common standard for manipulation of data from database tables [1].

This paper will be focusing on the use of SQL to maintain, manipulate and access databases from Visual C++ via OLE to meet specific needs. OLE, just like Open Database Connectivity (ODBC), is an Application Programming Interface (API) that is capable of linking any database application that is OLE-compliant to its appropriate DBMS software through its driver. OLE is implemented as a standard SQL-based interface which is a vital part of the Windows Operating System.

#### II. OBJECTIVES

This study intends:-

To find out a common interface through which any database can be accessed.

Chapman Eze Nnadozie is with the Department of Computer Science, Abubakar Tatari Ali Polytechnic, Bauchi, Nigeria (e-mail: chapsuccess@gmail.com).

- To find out how a database connection can be established in Visual C++ programming language environment.
- To determine a standard programming language extension with which queries can be issued in order to access and manipulate database tables from Visual C++.

#### **III. LITERATURE REVIEW**

A database can be seen as an integrated collection of logically related records or objects. Databases, consisting of tables that contain fields of data, are necessary in every organization for its effective management. Therefore, business applications need to maintain, manipulate, and access records stored in database tables. Each field in a database contains specific data, while a collection of related fields make up a record. DBMS software is used to maintain a database [2].

The older versions of windows operating system has the ODBC interface implemented as a standard SQL-based interface with the ability of handling any form of databases [3]. ODBC API architecture is such that its manager has the driver for all DBMS software. This architecture enables the DBMS vendors to focus on the creation of specific driver for their respective software without worrying about the entire path to an application [4]. Newer computers with latter versions of windows operating system like Windows 7, Windows 8 and Windows 10 are using OLE to connect to OLE databases.

Fig. 1 shows that an ODBC application makes database calls to the ODBC Driver Manager which in turn translates it to the appropriate database driver to establish a database connection. In other words, what the ODBC manager does is to link you to the appropriate driver for the data source you want through which your database can be accessed for maintenance, editing and querying as the case may require using SQL statements [2].

SQL is a standard relational database language for accessing and manipulating databases. This implies that SQL statements are used to query database tables to extract or change the information contained therein. With SQL commands, data can easily be extracted from any type of DBMS. The following are the basic types of SQL statements: SELECT, INSERT, DELETE and UPDATE [5].

1) SELECT: This is the simplest form of query that selects rows and columns from one or more tables in a database. The basic form of a select statement is:

SELECT {DISTINCT} /Select - list/ FROM table\_list {WHERE condition } {ORDER BY field\_list {DESC }}

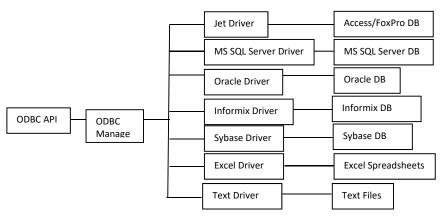


Fig. 1 ODBC architecture [2]

## THE WHERE CLAUSE

This is used to retrieve records that satisfy certain conditions. The WHERE clause condition can contain operators  $\langle , \rangle, =, \langle =, \rangle =, \langle \rangle$ , and Like. The Like operator is used for pattern matching with wildcard characters – percent (%) and underscore (\_).

## **ORDER CLAUSE**

This clause is used to sort the rows in a table. The named column becomes the basis for the sorting of the table either in ascending or descending order.

2) **INSERT**: This is used to add data into a table. The basic format is:

INSERT INTO table\_name (column\_lists) VALUES (value lists).

Table\_name refers to the name of the table. Column\_lists refer to the names of the columns you wish to add a new record to in your table. Value\_lists refer to the corresponding values you wish to insert into the table. Any column that is not specified is given a null value. However, column\_lists is omitted if you are adding values to all the columns in a table. Insert Into and Values are key words.

3) **DELETE:** This statement is used to remove an existing record in a table that meets the WHERE condition. It can also be used to delete an entire table from a database. Its format is:

DELETE FROM table\_name OR DELETE FROM table\_name WHERE search condition.

The table\_name is the name of the table in which you can delete rows. The search\_condition in the WHERE clause is the same as in the SELECT command.

4) UPDATE: This statement enables you to change a group of records in a table. It works directly on the source database table to update the data. Its format is as follows:

UPDATE table\_name

SET columnname1=value1, columnname2=value2,..., columnnameN=valueN

## WHERE condition.

Table\_name refers to the table to be updated. SET is a keyword. Columnname refers to the specific column you wish to replace its value with a new value. The where clause provides the condition that determines which rows to update.

## IV. METHODOLOGY

The researcher creates a database (StudResults) using the DBMS software Microsoft Access 2013. This database contains three relational database tables namely – DOT114 (with attributes AdmNo, LName, FName, ContAss, Exam, Total, and Grade), DOTSR (with attributes AdmNo, Dot111, Dot112, Dot113, Dot114, Dot115, Gns101, Gns111, and Gns121), and DOTGP (with attributes AdmNo, Tcur, Tcue, Csp, Cgpa, and Remarks). 'AdmNo' is the unique key for this relational database, and a one-to-one relationship exists between the tables. This database (StudResults) is then connected to within the Visual C++ programming language environment.

#### V. IMPLEMENTATION

The implementation is done in the visual C++ environment, and it entails the querying of the relational database earlier created using Microsoft access 2013 software to filter desired records. The programming language extension is to be used on the database is SQL which provides a standard means of maintaining, manipulating and accessing database tables. Using SQL, the researcher will be able to perform at least the following basic operations:

- > Retrieval of records from database tables.
- Insertion of new record(s) into a database table.
- > Updating of records in database tables.
- > Deleting of an existing record in a table.
- > Deleting of an entire table that is no longer needed.

#### VI. INPUT AND OUTPUT

Below are some screen shots of the inputs and their respective corresponding outputs in visual  $C^{++}$  9 environment.

## World Academy of Science, Engineering and Technology International Journal of Computer and Information Engineering Vol:12, No:8, 2018

olution Explorer 🛛 👻 🕂 🗙		(= 🛅 🛅 🚦									
	Que	ry1: Query(C:\\	StudResult	.cdb) Start Pag	e						
2		DOT114     All Colum     AdmNo     LName     FName     ContAss	ns) 🔺								
	۲.										
		Column	Alias	Table		Sort Type	Sort Order	Filter	Or	Or	0
	•	AdmNo		DOT114							
		LName		DOT114							
		FName		DOT114							
	SELECT FROM	AdmNo, LNa DOT114	ame, FName,	ContAss, Exam,	, Total, Gi	ade					
		AdmNo	LName	FNam	e	ContAss	Exam	Total	Grade		
	<u>۲</u>	27000	Umar	Shams	udeen	30	-40	70	Α		
		27001	Ezekiel	Lydia		18	22	40	D		
		27002	Bature	Christia	ana	12	13	25	F		
		27003	Sule	Abdul		24	26	50	с		

Fig. 2 Sample from a SELECT command

ine can thew bata e	Query Designe	op\Journals\Stud r Tools Wine	dow Help						
🎦 • 🖾 • 💕 🔙 🍠   🐰	- 🗈 🖺 🛛	) • (H • (II •		*			- 2		- 🖂
🐨 💷 🕺 🔠 Change Ty	pe 🕶 📘 🔞	🕼 💼 🗐	÷						
olution Explorer 🛛 👻 🖣	1 🗙 🖉 Qu	uery1: Query(C:\.	\StudResult	cdb) Start Pa	ige				
3		DOTGP (All Colu AdmNo Tcur Tcue Csp							
	<	Column	Alias	Table	Outp	Sort Type	Sort Order	Filter	Or
	×	AdmNo		DOTGP	$\checkmark$				
		Tcur		DOTGP	$\leq$				
	SELE	Tcue	mNo Tour To	DOTGP	Permarke				
	SELEO FROM ORDE	CT TOP 5 Ad		ue, Csp, Cgpa,					
	FROM	CT TOP 5 Adi M DOTGP			Remarks	Csp	Сдра	Remar	ks
	FROM	CT TOP 5 Ada M DOTGP ER BY Cgpa DESC	:	cue, Csp, Cgpa,	Remarks	Csp 89	Cqpa 3.87	Remar PASSED	
	FROM	CT TOP 5 Adi M DOTGP ER BY Cgpa DESC AdmNo 27034 27029	Tcur           23           23	ue, Csp, Cgpa, Tcu	Remarks				
	FROM ORDE	CT TOP 5 Adi M DOTGP ER BY Cgpa DESC AdmNo 27034 27029 27015	Tcur 23 23 23 23	zue, Csp, Cgpa, Tcur 23 23 23	Remarks	89	3.87	PASSED	). ).
Soluti 🧟 Class 🚍 Pro	FROM ORDE	CT TOP 5 Adi M DOTGP ER BY Cgpa DESC AdmNo 27034 27029 27015	Tcur           23           23	zue, Csp, Cgpa, Tcur 23 23 23	Remarks	89 89	3.87	PASSED PASSED	). ).
•	FROM ORDE	CT TOP 5 Adi M DOTGP ER BY Cgpa DESC AdmNo 27034 27029 27015	Tcur 23 23 23 23	zue, Csp, Cgpa, Tcur 23 23 23	Remarks	89 89	3.87	PASSED PASSED	). ).
itput	FROM ORDE	CT TOP 5 Ad DOTGP ER BY Cgpa DESC AdmNo 27034 27029 27015 4 1 of	Tcur 23 23 23 23	tue, Csp, Cgpa, 23 23 23 23 3   ■	Remarks	89 89	3.87	PASSED PASSED	). ).
itput	FROM ORDE	CT TOP 5 Ad DOTGP ER BY Cgpa DESC AdmNo 27034 27029 27015 4 1 of	Tcur 23 23 23 55 <b>&gt; &gt;</b>	tue, Csp, Cgpa, 23 23 23 23 3   ■	Remarks	89 89	3.87	PASSED PASSED	). ).
itput iow output from:	ope	CT TOP 5 Adi M DOTGP ER BY Cgpa DESC 27034 27034 27015 1 ot	Tcur 23 23 23 55 <b>&gt; &gt;</b>	tue, Csp, Cgpa, 23 23 23 23 3   ■	Remarks	89 89	3.87	PASSED PASSED	). ).
Soluti 🐼 Class 🕞 Pro stput now output from: Code Definition Window 🏂 eady	ope	CT TOP 5 Adi M DOTGP ER BY Cgpa DESC 27034 27034 27015 1 ot	Tcur 23 23 23 55 <b>&gt; &gt;</b>	tue, Csp, Cgpa, 23 23 23 23 3   ■	Remarks	89 89	3.87	PASSED PASSED	). ).

Fig. 3 Sample from another SELECT command

## World Academy of Science, Engineering and Technology International Journal of Computer and Information Engineering Vol:12, No:8, 2018

😫 Query3: Query(C:\Users\Chapma	n\Desktop\Jo	ournals\StudResu	lts.accdb) - Visu	al C++ 2008 Express E	dition		
File Edit View Data Query	Designer 1	Fools Window	Help				
i 🛅 • 🛅 • 💕 🗐 🥔 i 👗 🗈	🔁 🔊 🗸	(* • 📮 • 🖏		-	- 2	<u>*</u>	-
🐨 📃 🕺 🛃 Change Type 🕶	୍ତ୍ତ 😵 👔	1 1 個 📮					
Solution Explorer 🛛 🔻 🕂 🗙	Query	3: Query(C:\\St	udResultcdb)	Query2: Query(C:\	\MScSP1.accdb)	Query1: Query(C:\	\StudResultcdl
		DOT114  * (All Columns  AdmNo LName FName ContAss	<u>_</u> ;) <u>_</u>				
	<						
		Column	New Value				
		AdmNo	27054				
		LName	Abel				
		FName	Daniel				
	VALUES		lame, FName, Co Daniel, 30, 45, 66,	ontAss, Exam, Total, ( A)	Grade)		
🟹 Soluti 🐼 Class 📑 Prope	N 4	0 of 0	▶ ₩ ▶=  @	)			
Output							
Show output from:		- 8 4	) B,   T,   Z				
Code Definition Window 20 Call	Browser 🖃	Output		-			
Ready							
		~ _		1117	-	And the second second second	
			B. N		**		

Fig. 4 Sample from an INSERT command

Query3: Query(C:\Users\Chapmar				sual C++	2008 Express E	dition				
File Edit View Data Query I	-		-						- 5 - 3	3D 🗖
🖥 • 🕅 • 📂 💭 🕼 🗍 🐰 🖻				*					- 🗟 🚰 过	× 🗗
🐨 🔜 🕺 Change Type 🕶										_
lution Explorer 💿 🔻 🕂 🗙	Qu	ery3: Query(C:\\	StudResultcdl	) Query	/2: Query(C:\\	MScSP1.accd	b) Query1: Que	ery(C:\\StudResu	ltcdb) Start Page	
1		DOT114								
		* (All Colum	ins)							
		AdmNo								
		LName FName								
		ContAss	-							
		,	_							
	<									
	<u> </u>	Column	Table	Set	New Value	Filter	Or	Or	Or	
	Þ	AdmNo	DOT114		27054	Thee	01	01	01	
		LName	DOT114		Abel					
		FName	DOT114		Daniel					
	UPDA		1							
	SET	AdmNo = 2	7054, LName = /	Abel, FNa	ime = Daniel, C	ontAss = 30, E	xam = 45, Total =	= 66, Grade = A		
				<u> </u>						
Soluti 💽 Class 🔚 Prope		4 0 of 0								
utput										
how output from:		- 🖓	a 🕒 🔍	41						
Code Definition Window 🔚 Call B	Browser	Output								
eady										
	-					Contraction of the	-			-
P 🗆 🤶		â 🕘	5 B. 😹 🚺		w	**				

Fig. 5 Sample from an UPDATE command

## World Academy of Science, Engineering and Technology International Journal of Computer and Information Engineering Vol:12, No:8, 2018

🔩 Query3: Query(C:\Users\Chapman	\Deskto	p\Journals\StudRes	ults.accdb) - Vi	isual C++	2008 Express Ec	lition			
File Edit View Data Query D	esigner	Tools Window	/ Help						
🛅 • 🛅 • 📂 🔒 🥔   🐰 🐚	11 9	• (* • 📮 • 🖾		Ŧ		-	2	- 🗟 🕾 😒	. 🗆 🔹
😨 📃 🕺 🛃 Change Type 🕶	! 🕺	🕼 📩 📜							
Solution Explorer 🛛 🔻 🕂 🗙	Que	ery3: Query(C:\\S	tudResultcd	b) Query	/2: Query(C:\\	MScSP1.accdb)	Query1: Query(C:\\Stu	udResultcdb) Start Page	
		DOT114     * (All Column     AdmNo     LName     FName     ContAss	_						
	<	-	Table	6.4	New Value		0.5 5 K.	× 0r	
	Þ	Column	DOT114	Set 🗸	'Muhamm	Visual C++ 200	8 Express Edition	× 0r	
		LINAITIE	001114		wunamm	2	ows affected by last query		
						🕕 "	ows an ected by last query		
	UPDA' SET WHER	LName = 'M					OK Help		
रू Soluti 🐼 Class 📑 Prope	4	0 of 0							
Output									
Show output from:			£1 B≱   III	<b>P</b>					
Code Definition Window 📜 Call B	rowser	Output							
Ready									
H 2 🗆 🤤		â 👂	B. 🛛		w	**			

Fig. 6 Sample from another UPDATE command

Query1: Query(C:\Users\Chapman	\Desktop	Journals\StudRes	ults.accdb) - V	isual C++ 2008 Exp	oress Ed	ition	
File Edit View Data Query D	esigner	Tools Window	Help				
🛅 • 🛅 • 💕 🔒 🗿   🐰 🖬 🛛	2 9	- (		*		-	- 🗠 😁 🐋 🏷 💽 - 📮
🕲 🏢 🕺 👪 Change Type 🕶	? 💀	()=   🎁 🛅 📮					
Solution Explorer 🛛 🔻 🕂 🗙	Que	ry1: Query(C:\\S	tudResultcd	b) Start Page			
<u>6</u>		DOT114 ★!* (All Column AdmNo LName FName ContAss	5) ▲ ▼				
	<				-		
		Column	Table	Filter	Or	Visual C++ 2008 Express Edition	×
	•	LName	DOT114	= 'Yohanna'			
						1 row affected by last query	
	DELETI	E FROM DOT114 E (LName = 'Yo	hanna')			OK Help	]
🜄 Soluti 🐼 Class 📃 Prope	<b>M</b> 4	0 of 0					
Output							
Show output from:		- 3 4	2 By   = ;	<b>P</b>			
Code Definition Window 📜 Call B	rowser						
Ready							
		<b>(</b>	B. 🥂	W		14.	State and a state of the second state of the s



## VII. SUMMARY AND CONCLUSION

Typically, a DBMS is designed to create and manage a database. A database, which is a collection of related records, is composed of several fields called data members in C++ [6]. ODBC API, which is now being replaced by OLE API, has a standard SQL-based interface to work with any type of database thereby making database access a renowned reality. This is because you can create a database using any DBMS software, and administer it from a programming language such as Visual C++ to extract records from the database. Having established a connection to the database, the basic SQL commands can be used to query the database tables to obtain the desired results.

#### REFERENCES

- Williams, M., Bennett, D. et al. (2002). Microsoft Visual C++ 6 Unleashed. U.S.A.: Sams Publishing.
- O'Leary, T. J. and O'Leary, L. I. (2007). Computing Essentials. New York: The McGraw-Hill Companies.
   Chapman, D. (2001). Sams Teach Yourself Visual C++ 6 in 21 Days.
- [4] Lester, C. S. et al. (1999). Database Programming with Visual J++.
- [4] Lester, C. S. et al. (1999). Database Programming with Visual 314. U.S.A.: Que Corporation.
   [5] Din, Akeel I. (2007). Structured Query Language (SQL) A practical
- [5] Din, Aker I. (2007). Structured Query Language (SQL) A practical approach.
  [6] Dictal P. L. and Dictal H. M. (2010). Cliff Have to Decompose Sinth
- [6] Dietel, P. J. and Dietel, H. M. (2010). C++ How to Program –Sixth Edition. New Jersey: Pearson Prentice Hall.