

## تحليل وتصميم نظام محاسبي وتصميمه بوساطة "دورة حياة بناء النظام"

مضر زيد توفيق الدباغ

قسم علوم الحاسوب - كلية علوم الحاسوب والرياضيات

جامعة الموصل

[mudhardz@yahoo.com](mailto:mudhardz@yahoo.com)

### المستخلص

تعد عملية تحليل قاعدة البيانات وتصميمها من الأمور المهمة في مجال قواعد البيانات، إذ إنها تمثل العمود الفقري لأي قاعدة بيانات. وهذا البحث يمثل عملية تحليل نظام محاسبي للسيطرة على الديون والمخزن لمحل تجاري. لقد تم تطبيق هذا النظام في عدة محلات وشركات تجارية. يعتمد هذا النظام طريقة "دورة حياة بناء النظام" (System Development Life Cycle (SDLC)) المتبع مع الأنظمة غير المعقّدة.

يتناول البحث تحليل وتصميم نظام في مجال الحاسوبات بالاعتماد على قاعدة البيانات. ومن ثم تنفيذ هذا التصميم باستخدام أحد نظم إدارة قاعدة البيانات (Database Management Systems (DBMS)) وفيه يمكن التعرف على كيفية توظيف أدوات تحليل النظم وقاعدة البيانات في تحليل وتصميم نظام متكامل، في هذا البحث تم استخدام نظام "FoxPro 2.6 for Windows" لإمكاناته الجيدة في تمثيل قاعدة البيانات في بيئه الويندوز.

### Analysis and Design Accounting System by "System Development Life Cycle (SDLC)"

Midhar Z. Tawfeq

C.S. Dept. - College of C.S. and Math.

Mosul University

### Abstract

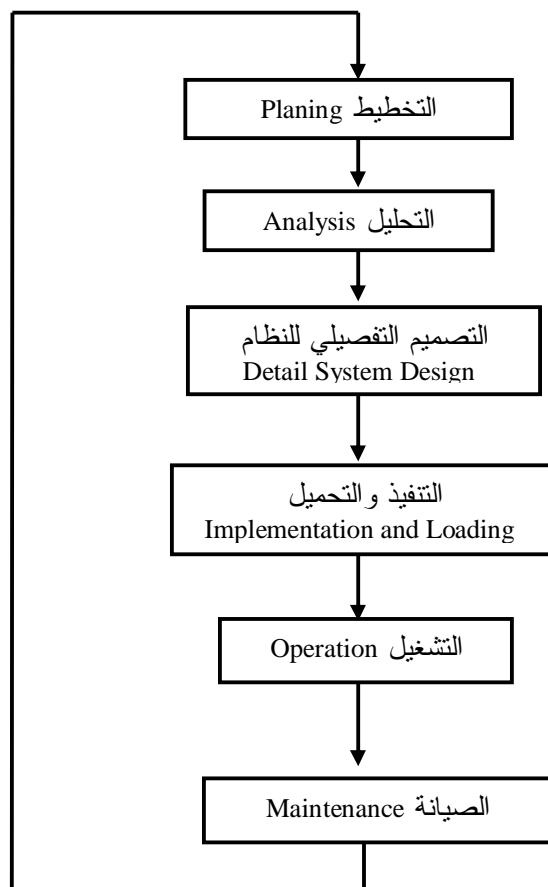
The operation database analysis and design is very important in the field of database. The current research represents the analysis and the design of accounting system in order to control the debts and stock of business store. This system applied in several stores and companies. The method "System Development Life Cycle" (SDLC) applied for analyzing and design the system used as simple systems.

This is treated to the analysis and the design of database system. This research tries to give an idea about the method of designing the accounting system. As well as, the implementation of this design by using one of the Database Management Systems (DBMS). In this research, the system used was "FoxPro 2.6 for windows" because of its ability to represent database in Windows ecology.

### ١. التمهيد

للبدء بتصميم نظام قاعدة بيانات يجب أولاً تحديد الخطة العامة لهذه العملية متمثلة بالخطوات الرئيسية ضمن طريقة "دورة حياة تصميم النظام System Development Life Cycle (SDL<sup>(١)</sup>)". كما في الشكل ١.

فمن المهم بعد دراسة كل الأدوات المستخدمة في مجال قواعد البيانات معرفة كيفية تشكيل هذه الأدوات ضمن سياق منظم، اذ يسهل استغلالها بشكل مفيد ومنتج للحصول على تصميم نظري كامل للنظام، ومن ثم تتنفيذ هذا التصميم على الحاسبة باستخدام نظام ادارة قاعدة البيانات Database Management System (DBMS). وتم في هذا البحث استخدام نظام Foxpro 2.6 for windows.



الشكل ١  
دورة حياة بناء النظام

## ٢. التخطيط Planning

قبل البدء بتصميم النظام الجديد يجب التخطيط لتصميم النظام المتمثل بالتقدير الأولي للنظام الحالي دراسة الجدوى للنظام المقترن<sup>(١)</sup>.

### ٢ - ١. التقدير الأولي Initial Assessment

النظام الحالي هو نظام يدوى يتمثل بسجل الديون، اذ يكون لكل زبون صفحة تتضمن مبالغ القوائم والإيصالات وبعد ذلك يضاف كل مبلغ إلى المجموع أو يطرح من المجموع للحصول على المجموع الجديد.

يلاحظ صعوبة كتابة سجل الديون بالإضافة إلى الوقت اللازم لملء السجل. فضلاً عن أن عند حدوث أي خطأ في الحساب يجب إعادة العملية من جديد. من نواقص النظام الحالي عدم وجود أي معالجة خاصة بالمخزن، اذ لا يمكن معرفة رصيد كل مادة والكمية المباعة والربح.

### ٢ - ٢. دراسة الجدوى Feasibility Study

تكليف النظم المقترن تتضمن ما يلي:

أ. جهاز حاسبة بمواصفات حدها الأدنى: معالج بينتيوم ٣ بسرعة ٥٠٠ ميكا هرتز، ذاكرة ٢٨ ميكا بايت، قرص صلب ١٠ كيكا بايت. سعر الجهاز كحد أدنى ٢٠٠ دولار أو أكثر بالاعتماد على المواصفات.

ب. جهاز طابعة نوع ليزر بسعر ١٣٠ دولار تقريباً.

ت. موظف يقوم بإدخال البيانات (القوائم والإيصالات) في حالة كون حجم البيانات كبير. هذا الموظف يتلقى راتباً يتراوح بين ١٠٠٠٠ ١٥٠٠٠ مائة الف دينار إلى مائة وخمسين الف دينار.

أما فيما يتعلق بفوائد النظام المقترن فهي تتلخص بال النقاط الآتية:

أ. سهولة الحصول على الديون بعد إدخال القوائم والإيصالات.

ب. لكل زبون كشف حساب يتم الحصول عليه بسهولة وبشكل آني.

ت. كذلك يتم تحديد لكل مادة: الرصيد المتبقى منها، كلفة القطعة الواحدة، القيمة الكلية للمادة، كلفة المخزن بالكامل. فضلاً عن الكمية المباعة من كل مادة مع سعر البيع وسعر الشراء والفرق بينهما (الربح).

ث. لكل مادة هناك كشف بالحركة التي أجريت على المادة من شراء وبيع ومقدار الربح المترتب على البيع ولا يزيد مدة زمنية يمكن تحديدها.

ج. يتم إعطاء الحركة النقدية مع رصيد الصندوق بعد كل حركة.

ح. يتم إعطاء المبيعات والمشتريات مع الربح المترتب عليها خلال مدة معينة.

خ. عند حدوث أي خطأ في القوائم أو الإيصالات يمكن إصلاح الخطأ بسهولة وتلقائياً يتم تحديث الديون والمخزن والكشف الخاص بكل زبون وكل مادة فضلاً عن تحديث رصيد الصندوق.

نلاحظ من هذه الفوائد المترتبة من استخدام النظام المقترن أنها أكثر بكثير من التكاليف، فبتجنب خطأ بسيط يحدث بسبب النظام الحالي يمكن توفير آلاف الدولارات.

### ٣. التحليل Analysis

بعد التخطيط لتصميم النظام والتأكد من قبول النظام والموافقة عليه من قبل إدارة الشركة أو صاحب المحل التجاري، يمكن البدء بتحليل النظام المقترن من خلال الخطوات الآتية<sup>(١)</sup>:

#### ٣ - ١. متطلبات المستخدم User Requirements

بعد إجراء مقابلة مع المستخدمين يتم تحديد كل ما يحتاج إليه المستخدم فيما يخص النظام. وهذه المتطلبات تمثل مخرجات النظام، وكما يأتي: الديون، المخزن، الصندوق، الزيتون، والمادة

أ. الديون تتضمن ما يأتي: اسم المدين أو الدائن مع الرصيد بالعملتين الدينار والدولار.  
ب. المخزن تتضمن ما يأتي: إسم المادة، المتبقي من المادة، تكلفة الوحدة، المبلغ الكلي، الكمية المباعة، سعر البيع، المبلغ الكلي، سعر الشراء، المبلغ الكلي، الربح الكلي، ربح القطعة الواحدة، والنسبة.

ت. الرصيد ويتضمن ما يأتي: المبلغ الوارد، المبلغ الصادر، نوع الحركة، رقم القائمة، تاريخ القائمة، والملحوظات.

ث. كشف الزيتون ويتضمن ما يأتي: المبلغ، نوع الحركة، رقم القائمة، تاريخ القائمة، الملاحظات، وإسم الحساب المقابل في الحركة. كما يضاف في الكشف التفصيلي: اسم المادة، العدد، السعر.

ج. كشف المادة ويتضمن ما يأتي: اسم الزيتون الذي قام بالشراء أو البيع، نوع الحركة شراء أم بيع، رقم القائمة، تاريخ القائمة، عدد القطع، سعر القطعة الواحدة، المبلغ، سعر الشراء، الربح، الملاحظات.

ح. المشتريات تتضمن ما يأتي: التاريخ، رقم القائمة، المادة، العدد، السعر، والمبلغ.

خ. المبيعات تتضمن ما يأتي: التاريخ، رقم القائمة، المادة، العدد، السعر، المبلغ، سعر الشراء، والربح.

#### ٢ - ٣. تقييم النظام الحالي Existing System Evaluation

يحتوي النظام الحالي على عدد من المشاكل فيما يأتي يتم ذكرها:

أ. طول الوقت وصعوبة حساب رصيد كل زبون. حيث يجب إضافة مبلغ قائمة البيع ومبلغ إيصال الدفع إلى رصيد الزيتون، وطرح مبلغ قائمة الشراء ومبلغ إيصال الاستلام من رصيد الزيتون.

ب. عند حدوث أي خطأ في الحساب بسبب خطأ في الجمع أو الطرح أو خطأ في ترحيل القوائم والإيصالات يجب حذف جميع العمليات بعد الخطأ وإعادة حسابها، وذلك لأن الخطأ سوف يؤثر على العمليات التي تليه في الكشف.

ت. لا يمكن حساب رصيد المادة إذ إن هذه العملية تكون صعبة ومجده، لذلك لا يتم إجراءها، ومن ثم لا يمكن اكتشاف أي نقص في أرصدة المواد عند إجراء الجرد.

ث. ليس هنالك أي سيطرة على حركة المواد ومعرفة المواد الأكثر مبيعاً لغرض التركيز عليها.

ج. ليس من الممكن حساب ربح كل مادة من خلال جرد مشتريات المادة ومبيعاتها وحساب الفرق بين المشتريات والمبيعات لهذه المادة، لأن هذه العملية أيضاً من الصعب إنجازها.

ح. حساب رصيد الصندوق يستغرق وقتاً طويلاً مع احتمالية وقوع أخطاء أثناء عملية التسجيل. فضلاً عن أن الإيصالات يجب أن ترحل إلى حساب الزبون وحساب الصندوق أيضاً في الوقت نفسه.

خ. عند إجراء أي تعديل على القوائم أو الإيصالات، من صعب بمكان تحديث الصفحة الخاصة بالزبون فيما يخص القائمة بالاجل والإيصال. وكذلك فيما يخص رصيد الصندوق. هذا فضلاً عن أن المواد لا يوجد أي كشف خاص بها.

كما لاحظنا فيما سبق المشاكل الكثيرة للنظام الحالي التي سوف يتم معالجتها من خلال النظام المقترن. وفيما يأتي التصميم المنطقي للنظام المقترن.

### ٣ - ٣. تصميم النظام المنطقي Logical System Design

بعد تحديد مشاكل النظام الحالي، يمكن البدء بتصميم النظام المقترن. إذ يجب تجنب جميع المشاكل المذكورة سابقاً، وإلا يعد النظام المقترن غير مفيد. بالاعتماد على متطلبات المستخدم (مخرجات النظام المقترن) <sup>(١)</sup>.

قبل البدء بالتصميم المنطقي للنظام يجب تحديد نظام إدارة قاعدة البيانات المستخدم. وفي هذا البحث تم استخدام نظام "FoxPro 2.6 for Windows" في تمثيل قاعدة البيانات ذات الحجم الكبير، والفصل بين البيانات والتطبيق ومن ثم يمكن تكوين نسخة احتياطية من البيانات.

وبحسب مفهوم دورة حياة بناء النظام هناك عدة مستويات او مراحل يجب المرور عليها وهي <sup>(٢)</sup>:

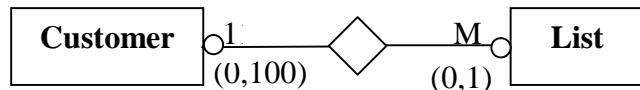
#### ٣ - ٣ - ١. المستوى المفاهيمي Conceptual Level

يتمثل هذا المستوى بمخطط الكينونة والعلاقة Entity-Relationship Diagram (ERD) من خلال المتطلبات المذكورة سابقاً نقوم بتحديد الكينونات الرئيسية التي يجب توافرها لغرض الحصول على هذه المتطلبات (المخرجات). وهذا عرض لهذه الكينونات:

أ. List: تمثل القوائم والإيصالات. تستخدم في حساب الديون والصندوق المخزن والمبيعات والأرباح.

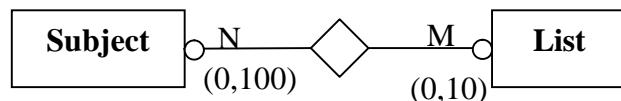
ب. Customer: تمثل الزبائن. تستخدم لغرض حساب الديون.  
ت. Subject: تمثل المواد. تستخدم لغرض حساب المخزن والمبيعات والأرباح.  
بعد تحديد الكينونات الرئيسية يجب وضع القواعد التي يتم من خلالها الربط بين هذه الكينونات ضمن ERD، الذي يتضمن: الترابط Connectivity وهي تكون (1:1, 1:M, M:1, M:N)، درجة العلاقة Cardinality التي تمثل الحد الأدنى والأعلى من عدد القيود التي ترتبط بالكينونة الآخر، ومشاركة العلاقة Relationship Participation التي تحدد نوع العلاقة هل هي اختيارية Optional (دائرة) أم إجبارية Mandatory (بدون دائرة) <sup>(٤)</sup>. وكما يأتي:

**القاعدة ١:** لكل قيد من List قيد واحد من Customer، وللقيد الواحد من Customer أكثر من قيد من List كما في الشكل ٢.



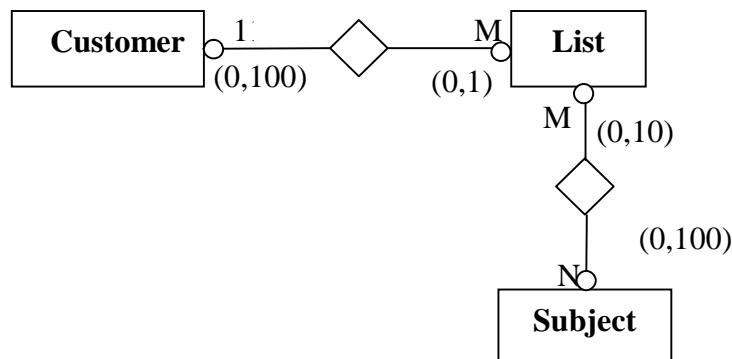
الشكل ٢ ERD للقاعدة ١

**القاعدة ٢:** لكل قيد من List أكثر من قيد من Subject، وللقيد الواحد من Subject أكثر من قيد من List كما في الشكل ٣.



الشكل ٣ ERD للقاعدة ٢

الشكل ٤ يمثل ERD للمستوى المفاهيمي.



الشكل ٤ ERD للمستوى المفاهيمي

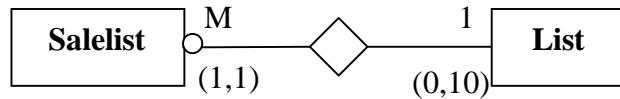
### ٣-٣-٣ المستوى الداخلي Internal Level

في هذا المستوى يتم تحويل الـ ERD إلى مجموعة من الجداول. لقيام بذلك يجب تحويل أي علاقة من نوع M:N إلى نوع 1:M، لأن هذه العلاقة لا يمكن تمثيلها ضمن جدول.

**القاعدة ٢** تحتوي على علاقة M:N التي يجب تحويلها إلى علاقاتين من نوع 1:M.<sup>(٤)</sup> فيما يأتي عملية التحويل:

**القاعدة ٣:** لكل قيد من List أكثر من قيد من Salelist ولكل قيد من Salelist قيد واحد من List كما في الشكل ٥. تمثل فقرات قوائم البيع.

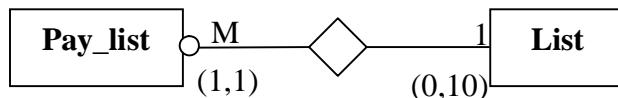
---



**الشكل ٥ ERD للقاعدة ٣**

**القاعدة ٤:** لكل قيد من List أكثر من قيد من Pay\_list ولكل قيد من Pay\_list قيد واحد من List كما في الشكل ٦. تمثل فقرات قوائم الشراء.

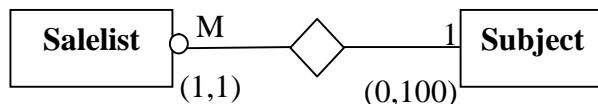
---



**الشكل ٦ ERD للقاعدة ٤**

**القاعدة ٥:** لكل قيد من Salelist أكثر من قيد من Subject ولكل قيد من Salelist قيد واحد من Subject كما في الشكل ٧.

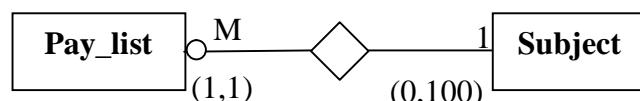
---



**الشكل ٧ ERD للقاعدة ٥**

**القاعدة ٦:** لكل قيد من Subject أكثر من قيد من Pay\_list ولكل قيد من Pay\_list قيد واحد من Subject كما في الشكل ٨.

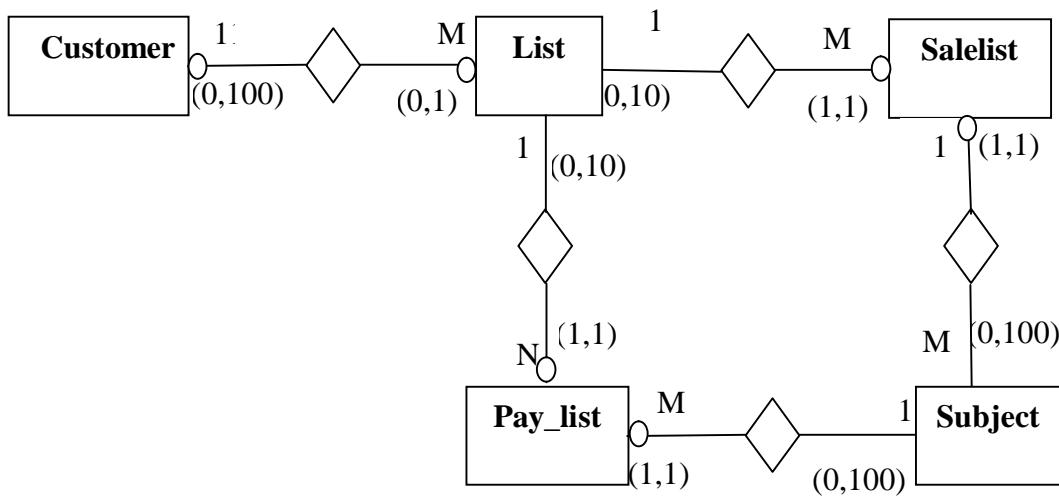
---



**الشكل ٨ ERD للقاعدة ٦**

**الشكل ٩** يمثل ERD النهائي للمستوى الداخلي.

---



الشكل ٩ المستوى الداخلي لقواعد

بعد الانتهاء من قواعد المستوى الداخلي يجب كتابة قاموس البيانات Data Dictionary وكما في الجدول ١.

### الجدول ١١ قاموس البيانات للجدول List

المرجع	المفتاح	الوصف	النوع	اسم الحقل	اسم الجدول
	أساسي	رمز القائمة	N(10)	Code	List
		رقم القائمة	N(10)	List_code	
Customer	أجنبي	رمز الزيون	N(8)	Customer	
		تاريخ القائمة	D(8)	List_date	
		وقت القائمة	C(8)	TIME	
		وقت القائمة للعرض	C(9)	OUTTIME	
		نوع القائمة	N(14)	LIST_TYPE	
		العملة للمبلغ الأول	L(1)	CURRENCY1	
		المبلغ الأول	N(14,2)	MONY1	
		نوع الدفع	L(1)	PAYING	
		العملة للمبلغ الثاني	L(1)	CURRENCY2	
		المبلغ الثاني	N(14,2)	MONY2	
Customer	أجنبي	الحساب المقابل	N(6)	RECEIVER	

يتبع ←

ما قبله ←

		الملحوظات	C(60)	NOTE	
		سعر الدولار	N(4)	DOLARPRICE	
			156	المجموع	

### الجدول ١ ب

#### ب قاموس البيانات للجدول SaleList

المرجع	المفتاح	الوصف	النوع	اسم الحقل	اسم الجدول
List	أجنبي	رمز القائمة	N(10)	List_code	SaleList
Subject	أجنبي	رمز المادة	N(6)	Subject	
		الكمية	N(10,3)	Number	
		السعر	N(14,3)	Unitprice	
			40	المجموع	

### الجدول ١ ت

#### قاموس البيانات للجدول Pay\_List

المرجع	المفتاح	الوصف	النوع	اسم الحقل	اسم الجدول
List	أجنبي	رمز القائمة	N(10)	List_code	Pay_List
Subject	أجنبي	رمز المادة	N(6)	Subject	
		الكمية	N(10,3)	Number	
		السعر	N(14,3)	Unitprice	
			40	المجموع	

### الجدول ١ ث

#### قاموس البيانات للجدول Customer

المرجع	المفتاح	الوصف	النوع	اسم الحقل	اسم الجدول
	أساسي	رمز الزبون أو الحساب	N(8)	Code	Customer
		اسم الزبون	C(50)	Name	
		الرصيد القديم من الدينار	N(10)	Orgin1	
		الرصيد القديم من الدولار	N(11,3)	Orgin2	
		قابل للمسح ام لا	L(1)	Constant	
		نوع الزبون	N(1)	Important	
			81	المجموع	

### الجدول ١ جـ قاموس البيانات للجدول Subject

اسم الجدول	اسم الحقل	النوع	الوصف	المفتاح	المرجع
Subject	Code	N(6)	رمز المادة	اساسي	
	Name	C(40)	اسم المادة		
	Item_no	C(10)	PN للمادة		
	Number	N(10,3)	الكمية القديمة من المادة		
	Unitcost	N(14,3)	كلفة الكمية القديمة من المادة		
	Type	L(1)	نوع العملة (دينار / دولار)		
	Constant	L(1)	قابل للمسح ام لا		
	المجموع	82			

### ٣-٣-٣ المستوى الخارجي External Level

في هذا المستوى يتم تحويل متطلبات المستخدم (المذكورة في الفقرة ٣-١) إلى مجموعة من عروض خارجية External Views، كل عرض يتكون من جزعين. الأول هو مخطط ERD يحتوي على الكيغونات المستخدمة في العرض. أما الثاني فهو مجموعة استعلامات Queries، وكل استعلام يمثل ايعاز SQL. وقد تم استخدام الاستعلامات التالية (٤):

- أ. **الديون Payment:** للحصول على الديون نحتاج الى الكيغونات Cusomter List في الشكل ٤.
- **الاستعلام ١ Payment 1:** في هذا الاستعلام يتم حساب رصيد الحساب بالدينار باستعمال الدالة Balance1 وبالدولار باستعمال الدالة Balance2 مع اسم الزبون. يدخل في حساب الرصيد مبلغ البيع والشراء، وكذلك مبلغ وصل الاستلام ووصل الدفع، فضلاً عن ارباح قوائم البيع.
- ب. **كشف الزيون Customer:** للحصول على كشف زبون نحتاج الى الكيغونات Cusomter List في الشكل ٤.
- **الاستعلام ٢ Amount1:** لغرض ربط الكيغونة list مع الكيغونة Customer M.customer Receiver. اذا كان الحقل Customer والحقن Receiver يساوي (الحساب المطلوب كشفه). الجدول type1 لغرض تكرار List. كما في الشكل ١٠ (٣).

```

SELECT list_fields,;
IIF(type1.i=1, customer,00000000) AS customer,;
IIF(type1.i=2 OR receiver!=customer,;
    receiver,00000000) AS receiver,;
customer.name AS customername ;
FROM list,customer1 customer,type1 ;
WHERE (list.customer = m.customer OR ;
    list.receiver = m.customer) AND ;
    (type1.i=1 OR list.receiver = list.customer ) AND ;
    customer.code=receiver1(list.code) AND ;
    list.list_date<=m.edate ;
INTO CURSOR amount1;
ORDER BY list.list_date, list.time

```

### الشكل ١٠ الاستعلام ٢ Amount1

- الاستعلام ٣ Amount21: لاضافة الرصيد القديم للزبون بالدينار والدولار الى القوائم المتمثلة بالاستعلام ٢ عن طريق UNION.
- الاستعلام ٤ Amount2: يقوم بجمع القوائم التي تسبق تاريخ بدء الكشف بوصفه الرصيد القديم وعرض جميع القيود التي تلي تاريخ بدء الكشف التي تمثل تفاصيل الكشف باستخدام UNION.
- الاستعلام ٥ Amount3: للتأكد من أن المبلغ الأول (المدين) يضم الأرقام الموجبة فقط، والمبلغ الثاني (الدائن) يضم الأرقام السالبة فقط.
- الاستعلام ٦ Amount: يمثل الصورة النهائية للكشف التي تضم عملة المدين (currency1)، مبلغ المدين (mony1)، عملة الدائن (currency2)، مبلغ الدائن (mony2)، تاريخ القائمة (list\_date)، نوع القائمة (type)، رقم القائمة (list\_code)، وفي النهاية الرصيد بالدينار (total1) وبالدولار (total2) الذي يتم حسابه في هذا الاستعلام من خلال ربط الاستعلام السابق مع نفسه بالاعتماد على تاريخ القائمة ورقم القائمة كما في الشكل ١١.

```

SELECT amount3_fields,;
    round(SUM(IIF(amount4.currency1="Id",;
                    amount4.mony1,000000000.00000)+;
                    IIF(amount4.currency2="Id",;
                    amount4.mony2,000000000.00000)),0);
                    AS total1,;
    round(SUM(IIF(amount4.currency1=" $",;
                    amount4.mony1,000000000.00000)+;
                    IIF(amount4.currency2=" $",;
                    amount4.mony2,000000000.00000)),;
                    pricedec) AS total2;
FROM amount3, amount3 amount4;
WHERE (amount3.list_date>amount4.list_date;
       OR amount3.list_date=amount4.list_date;
       AND amount3.code>=amount4.code);
GROUP BY amount3.code;
ORDER BY list_date, code, type;
INTO CURSOR amount

```

الشكل ١١ الاستعلام ٦ Amount

ت. كشف تفصيلي للزبون CustDtl: للحصول على كشف تفصيلي للزبون نحتاج إلى جميع كيونات قاعدة البيانات كما في الشكل ٩.

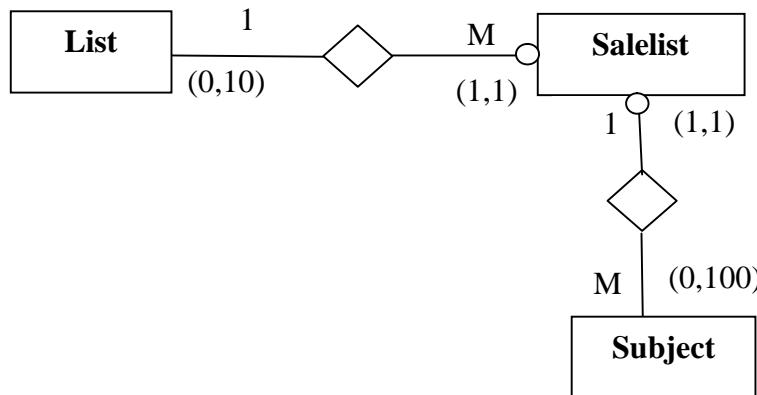
الاستعلام ٧: لغرض حساب نسبة الخصم او الزيادة في المبلغ لكل قائمة. تم استخدام الكيونة List والキーونة Salelist في الجزء الاول من الاستعلام الخاص بالمبيعات. أما الجزء الثاني فهو مشابه للأول مع اختلاف استعمال الكيونة UNION بدلاً من Salelist Pay\_list. باستخدام قمنا بالربط بين الجزئين بحيث نحصل على ناتج يمثل القوائم مع نسبة الخصم او الزيادة في كل قائمة.

الاستعلام ٨: يقوم باسترجاع تفاصيل القائمة (شراء او بيع) بالإضافة الى حساب السعر الحقيقي بعد تطبيق النسبة المأخذة من الاستعلام ٧ Listratio.

الاستعلام ٩: Amount4 يقوم بالربط بين الاستعلام ١١ amount3 والاستعلام list1١٥ الذي يضم تفاصيل القائمة.

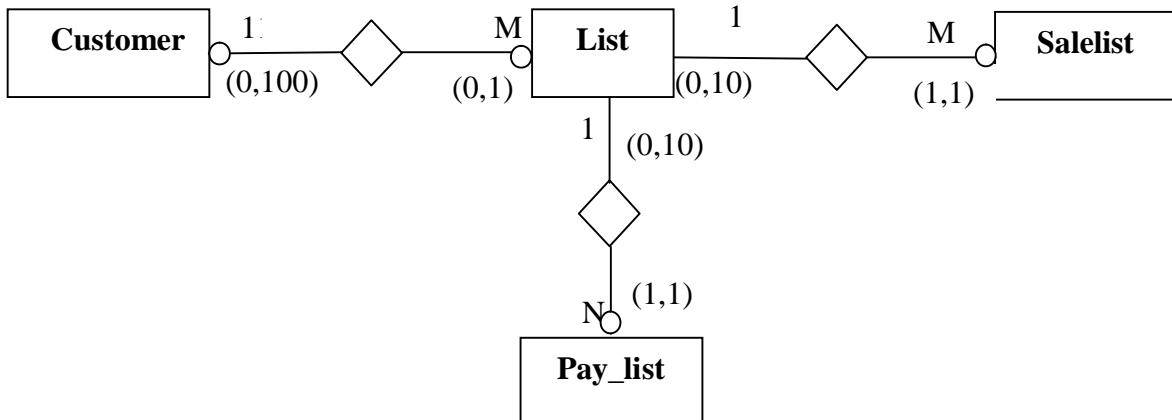
الاستعلام ١٠: Amount يمثل الصورة النهائية للكشف التفصيلي كما في الاستعلام ٦ مع اضافة: رقم المادة (subject)، الكمية (number)، سعر القطعة الواحدة (unitpirce)، العملة (currecny3)، اسم المادة (subjectname)، ورمز المادة (item\_no). في هذا الاستعلام تم استخدام GROUP BY و SUM .

ث. المخزن ViwStock: يتضمن هذا العرض أسماء المواد مع الرصيد، المبيعات، والأرباح لكل مادة. في هذا العرض يتم استخدام الكيونات List، Salelist، Subject و كما في الشكل ١٢ .



الشكل ١٢ ERD العرض ViewStock

- الاستعلام ١١: حساب مجموع المبيعات من كل مادة بالدينار والدولار. يستخدم في هذا الاستعلام **SUM** مع **GROUP BY**, فضلاً عن **UNION**.  
المتغيران  $m.edate$  و  $m.bdate$  تمثل المدة الزمنية للتقرير.
- الاستعلام ١٢: Stock1: حساب الرصيد ( $epiece$ ) وكلفة القطعة الواحدة ( $eunitcost$ ) واجمالي الكلفة بالدينار ( $etotalid$ ) والدولار ( $etotalud$ ) لكل مادة، فضلاً عن المبيعات التي تم حسابها في الاستعلام ١٨. ذلك يتم حساب كلفة الكميات المباعة من المادة ( $tcostid, tcostud$ ).
- الدالة  $piece$  لحساب الرصيد مع الكلفة. الدالة  $subcost$  لحساب كلفة المبيعات في هذا العرض يتم استخدام البرمجة الاجرامية (Procedural Language) لحساب الرصيد والكلفة، بخلاف العروض السابقة، اذ لا يمكن حسابها باستخدام لغة استرجاع البيانات SQL التي تسمى اللغة غير الاجرامية (Non Procedural Language).
- الاستعلام ١٣: Stock2: لحساب الربح بطرح الكلفة من المبيعات ( $(tsaleid - tcostid)$ ،  $(tsaleud - tcostud)$ ) بالاعتماد على الاستعلام ١٢.
- الاستعلام ١٤: Stocklist: لحساب نسبة الربح لكل مادة من خلال قسمة الربح على الكلفة مضروباً في مائة .
- الاستعلام ١٥: Stocklist: الاستعلام النهائي ويتم من خلاله حذف المواد التي ليس فيها رصيد وليس فيها مبيعات في نفس الوقت.
- ج. كشف المادة SubStock: يقوم هذا العرض بإصدار كشف خاص بالمادة يتضمن عملية الشراء والبيع التي قد تمت على المادة المطلوبة. ويستخدم في هذا العرض الكائنات List, SaleList, PayList, and Customer كما في الشكل ١٣ .



الشكل ١٣ العرض

**SubStock**

**الاستعلام ١٦: Orgin**: من خلال هذا الاستعلام يتم تضمين الرصيد القديم من المادة عند بدء العمل بالنظام.

- **الاستعلام ١٧ Stk1:** في هذا الاستعلام يتم ربط الرصيد القديم (الاستعلام ١٦ Orgin) مع المشتريات والمبيعات باستعمال UNION.

- **الرصيد Mony:** من خلال هذا العرض يتم الحصول على كشف يضم الحركة النقدية للشركة من استلام مقوضات او مدفوعات. وتم استخدام الكينونتين Customer و list في هذا العرض كما في الشكل ٢.

- **الاستعلام ١٨ Mony:** الاستعلام الوحيد المستخدم في هذا العرض والذي يضم كشف الحركة النقدية ضمن المدة الزمنية المحددة من خلال المتغيرين m.bdate و .m.edate.

**المبيعات Sales:** تم بناء هذا العرض لعرض الحصول على كشف المبيعات مرتبًا حسب التاريخ، ويتم من خلاله تدقيق قوائم البيع الداخلة إلى النظام. وفي هذا العرض تم استخدام جميع كينونات قاعدة البيانات كما في الشكل ٩.

- **الاستعلام ١٩ List Num:** لعرض حساب مجموع كمية البيع لكل قائمة للمادة نفسها، وبالسعر نفسه. كما تم استخدام GROUP BY مع SUM.
- **الاستعلام ٢٠ List1:** في هذا الاستعلام يتم حساب عدد المواد في كل قائمة من خلال GROUP BY و COUNT.

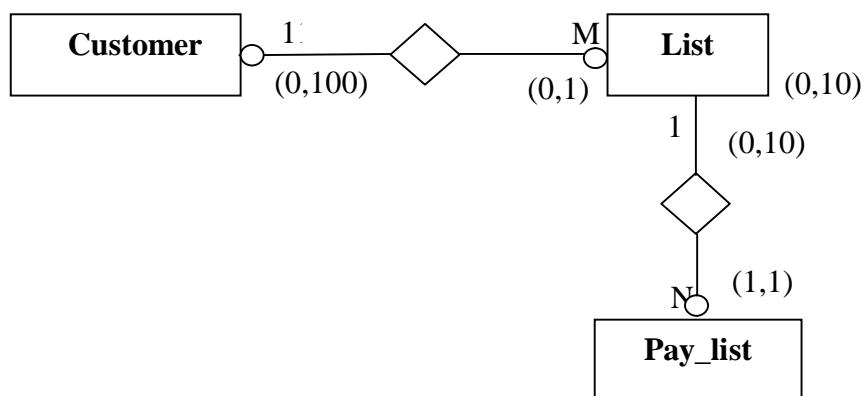
- **الاستعلام ٢١ Sales1:** في هذا الاستعلام يتم تحديد اسم الزبون في القائمة، أسماء المواد، الكمية (مع تجنب تكرار المواد في القائمة من خلال حساب مجموع الكميات للمادة نفسها في القائمة نفسها)، مع باقي تفاصيل القائمة.

- **الاستعلام ٢٢ Sales2:** يتم من خلال هذا الاستعلام حساب كلفة الكمية المباعة

بالدينار وبالدولار باستخدام دالة totalcost، وهي دالة لغرض حساب كلفة مادة في قائمة بيع.

- الاستعلام ٣ Sales3: لحساب الربح من خلال طرح الكلفة من مبلغ البيع، ويتم حساب اجمالي الكلفة بالدينار او بالدولار اعتماداً على قائمة البيع مع احتساب سعر الدولار مقابل الدينار بتاريخ القائمة. هذا الاستعلام هو الاستعلام النهائي في هذا العرض.

خ. كشف مشتريات مادة من مبيعات SubSales: يتم من خلال هذا العرض تحديد مصدر مبيعات مادة في قائمة يتم اختيارها من العرض السابق. في هذا العرض تم استخدام (SQL)، فضلاً عن البرمجة الاجرامية. اذ يتم بناء الاستعلام، ومن ثم بالاعتماد على هذا الاستعلام يتم بناء جدول يحتوي على الكشف. بالتأكيد إن استخدام (SQL) اكثر كفاءة، ولكن لايمكن الحصول على كل شيء من خلال (SQL)، من هنا نضطر للرجوع الى البرمجة التقليدية. وفي هذا العرض تم استخدام الكينونات Customer, List, Pay\_list كما في الشكل ٤.



الشكل ٤ ERD العرض SubSales

الاستعلام ٤ Pays: من خلال هذا الاستعلام يتم جلب المشتريات للمادة المطلوبة لغاية تاريخ البيع للقائمة المطلوب جردها. وتم الحصول على مجموع الحقل number لكل قائمة وبالسعر نفسه في حالة تكرار المادة في القائمة.

- الجدول ١ Stk: هذا الجدول مشابه في الهيكل للاستعلام السابق، وتم تكوينه من خلال الاياعز COPY STRUCTURE. اولاً يجب حساب المتبقي من المادة لغاية تاريخ البيع باستخدام الدالة remain. بعد ذلك نقوم باستبعاد قوائم الشراء في ابتداء من النهاية لحين استفاد المتبقي، وما يسبقه هو مصدر الكمية المباعة لحين استفاد هذه الكمية. وتم استخدام الاياعز APPEND BLANK لاضافة قيد الى الجدول GATHER. كما تم نقل القيد من الاستعلام ٤ من خلال الاياعز Stk SCATTER، وادخل القيد الى الجدول Stk باستخدام الاياعز MEMVAR

كما في الشكل ١٥. المتغير `m.listnumber` هو الكمية المتبقية من المادة المباعة.

```

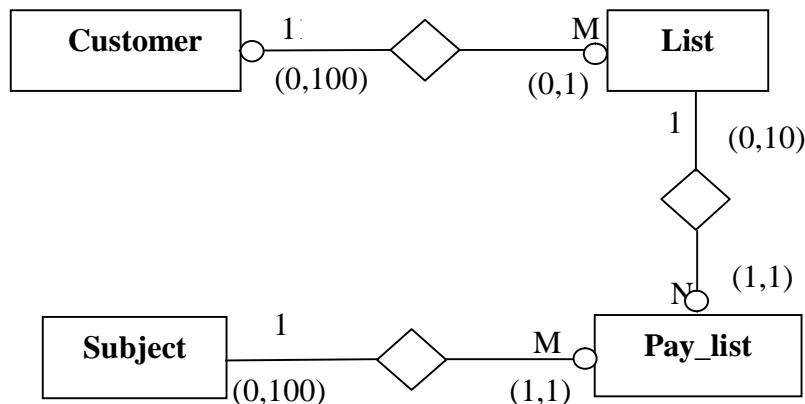
DO WHILE !BOF() AND m.listnumber>m.number
    m.listnumber=m.listnumber-m.number
    SELECT stk
    APPEND BLANK
    GATHER MEMVAR
    SELECT pays
    SKIP -1
    SCATTER MEMVAR
ENDDO

```

الشكل ١٥ المقطع البرمجي الخاص بنقل البيانات الى Stk

هـ . المشتريات Pays: تم بناء هذا العرض لغرض تدقيق قوائم الشراء. يضم هذا العرض المشتريات بكامل التفاصيل بضمنها اسم الزبون واسم المادة مرتبة حسب التاريخ. وفي هذا العرض تم استخدام الكينونات Customer, List, Pay\_list, Subject كما في الشكل ١٦.

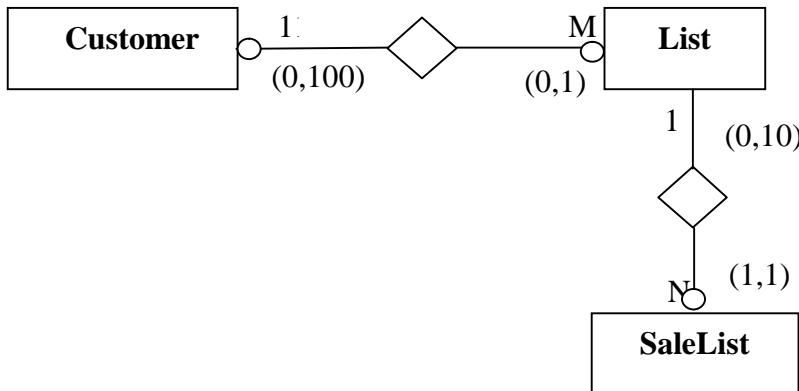
- الاستعلام ٢٥ Pays: هذا الاستعلام يتضمن المشتريات بكامل التفاصيل، وهو الاستعلام الرئيس في هذا العرض. وتم استخدام `SUM` للحقل number للتخليص من المواد المتكررة في القائمة الواحدة، فضلاً عن `GROUP BY`.



الشكل ١٦ مخطط ERD للعرض Pays

**د. كشف مبيعات مادة من مشتريات Subject:** هذا العرض مشابه للعرض في الفقرة و. يتضمن استعلام وجدول يتم بناءه بالاعتماد على الاستعلام. ولتنفيذ هذا العرض تحتاج الى الكيالونات Customer, List, Salelist كما موضح في الشكل ١٧.

---



الشكل ١٧ مخطط ERD للعرض Subject

- **الاستعلام ٢٦ Sales:** ضم هذا الاستعلام المبيعات للمادة المطلوبة لغاية تاريخ قائمة الشراء المختارة من العرض السابق. كذلك تم استخراج مجموع الحقل number لكل قائمة وبالسعر نفسه في حالة تكرار المادة في القائمة.
- **الجدول ٢ Stk:** هذا الجدول مشابه للجدول ١ غير أنه يعتمد على الاستعلام ٢٦.

#### ٤. تصميم النظام التفصيلي Detail System Design

في هذه المرحلة ننتقل إلى المستوى الفيزيائي Physical Level، يضم هذا المستوى تصميم الشاشات (screens) المستخدمة في إدخال البيانات وعرض المعلومات، تقارير الطابعة في كلتا الحالتين، والشاشة والقائمة (menu) الرئيسية في النظام. فضلاً عن تحديد جميع ملفات النظام. الجدول ٢، ٣ يحتوي على جميع شاشات النظام مع الملفات الخاصة بالشاشات<sup>(٢)</sup>.

## الجدول ٢

### شاشات النظام مع الملفات

الملفات	الابعاد المستخدمة	الوظيفة	اسم الشاشة	ت
Intro.scx Intro.sct Intro.spr	Screen Builder	الواجهة الرئيسية للنظام	Intro	١
Login.scx Login.sct Login.spr	Screen Builder	إدخال كلمة المرور للدخول إلى النظام	Login	٢
List.scx List.sct List.prg	Screen Builder Wizard	إدخال البيانات إلى الكائنات List، customer، Pay_List، SaleList	List	٣
Subjecte.scx Subjecte.sct Subjecte.spr	Screen Builder	اختيار مادة ضمن قائمة شراء او بيع كذلك يمكن اضافة مادة جديدة الى الكيوننة Subject	Subjecte	٤
Fhras.prg	Modify Command Define Window Browse	قائمة بالمواد الموجودة في النظام مع سعر البيع لكل مادة	Fhras	٥
Custfhrs.prg	Modify Command Define Window Browse	قائمة الزبائن في النظام	Custfhrs	٦
Viwstock.prg	Modify Command Define Window Browse	جرد المخزن (الرصيد، المبيعات، والربح) لكل مادة	Viwstock	٧
Substock.prg	Modify Command Define Window Browse	كشف مادة (الشراء، والبيع)	Substock	٨
Payment.prg	Modify Command Define Window Browse	قائمة ديون الزبائن (الرصيد، الدائن له، والمدين عليه) لكل زبون	Payment	٩
Customer.prg	Modify Command Define Window Browse	كشف زبون (الشراء، البيع، التسديد، السحب، والتحويل)	Customer	١٠
Custdtl.prg	Modify Command Define Window Browse	كشف زبون فضلاً عن المادة، الكمية، والسعر	Custdtl	١١
Mony.prg	Modify Command Define Window Browse	كشف الحركة النقدية (شراء نقدى، بيع نقدى، تسديد، وسحب)	Mony	١٢
Sales.prg	Modify Command Define Window Browse	المبيعات	Sales	١٣
Subsales.prg	Modify Command Define Window Browse	كشف مشتريات مادة ضمن المبيعات	Subsales	١٤
Pays.prg	Modify Command Define Window Browse	المشتريات	Pays	١٥

### الجدول ٣

#### تابع شاشات النظام مع الملفات

الملفات	الإيعازات المستخدمة	الوظيفة	اسم الشاشة	ت
Subejct.prg	Modify Command Define Window Browse	كشف مبيعات مادة ضمن المشتريات	Subject	١٦
Exhange.prg	Modify Command Define Window Browse	التحويلات	Exchange	١٧
Bakcup.scx Backup.sct Backup.spr	Screen Builder	تكوين نسخة احتياطية من البيانات	Backup	١٨
Restore.scx Restore.sct Restore.spr	Screen Builder	استرجاع البيانات من النسخة الاحتياطية	Restore	١٩
Password.scx Password.sct Password.spr	Screen Builder	تغيير كلمة المرور للنظام	Password	٢٠
Indexing.prg	Delete File *.cdx Index on	إعادة فهرسة البيانات	Indexing	٢١
برنامج لتكوين شاشة باستعمال الأسلوب المرئي وهو مرحلة أولية من البرمجة المرئية. لكتابة برنامج Modify Command Define Window لتعريف نافذة. لعرض محتويات جدول او استعلام.				

الجدول ٤ يحتوي على جميع التقارير المستخدمة في النظام.

القائمة الرئيسية باسم Salebuy تم بناؤها باستخدام البرنامج Menu Builder. ومن خلال هذه القائمة يتم تنفيذ جميع الشاشات في النظام. وتحتوي القائمة على شفرة الإعداد (setup code)، اذ في هذه الشفرة يتم تنفيذ الشاشة رقم ١ والشاشة رقم ٢. أيضاً تم بناء البرنامج Open لفتح الجداول، البرنامج Reset لتعريف المتغيرات العامة في النظام، والبرنامج Debts لحساب الديون.

بعد إكمال هذه المرحلة تكون قد حصلنا على نظام متكامل بجميع أجزائه.

### ٥. التنفيذ والتحميل Implementation and Loading

في هذه المرحلة نبدأ بالعمل على الحاسب الآلي. من خلال هذه الخطوة نقوم بتحويل التصميم النهائي من نظام مكتوب على الأوراق فقط إلى نظام يعمل في الحاسب الآلي. ويضم الخطوات الآتية:

## ٥-١. كتابة الشفرة، الاختبار، والتصحيح Writing Code, Testing, and Debugging

### الجدول ٤ تقارير النظام

الملفات	الوظيفة	اسم التقرير	ت
List.frx List.frt	طباعة قوائم الشراء والبيع	List	١
Mony.frx Mony.frt	طباعة وصل القبض، ووصل التسديد	Mony	٢
Fhras.frx Fhras.frt	طباعة فهرس المواد	Fhras	٣
Custfhrs.frx Custfhrs.frt	طباعة فهرس الزبائن	Custfhrs	٤
Stocksud.frx Stocksud.frt	طباعة المخزن بالدولار فقط	Stocksud	٥
Stocksid.frx Stocksid.frt	طباعة المخزن بالدينار فقط	Stocksid	٦
Stocksui.frx Stocksui.frt	طباعة المخزن بالدينار وبالدولار	Stocksui	٧
Substock.frx Substock.frt	طباعة كشف المادة	Substock	٨
Payment.frx Payment.frt	طباعة قائمة الديون (عامة، او وكلاء) او حسابات شخصية	Payment	٩
Customer.frx Customer.frt	طباعة كشف زبون	Customer	١٠
Custdtl.frx Custdtl.frt	طباعة كشف تفصيلي للزبائن	Custdtl	١١
Mony1.frx Mony1.frt	طباعة الحركة النقدية في الصندوق	Mony1	١٢
Salesid.frx Salesid.frt	طباعة المبيعات بالدينار فقط	Salesid	١٣
Salesud.frx Salesud.frt	طباعة المبيعات بالدولار فقط	Salesud	١٤
Salesui.frx Salesui.frt	طباعة المبيعات بالدينار وبالدولار	Salesui	١٥
Subsales.frx Subsales.frt	طباعة كشف مبيعات مادة من المشتريات	Subsales	١٦
Pays1.frx Pays1.frt	طباعة المشتريات	Pays1	١٧
Subject.frx Subject.frt	طباعة كشف مشتريات مادة من المبيعات	Subject	١٨
Exchange.frx Exchange.frt	طباعة التحويلات	Exchange	١٩

في هذا المرحلة نقوم ببناء قاعدة البيانات، الشاشات، التقارير، القائمة الرئيسية،

وكتابة الشفرة الخاصة بكل جزء. أثناء ذلك يتم اختبار كل جزء وتصحيح أي خطأ يمكن أن يظهر.

مثال على ذلك:

عند تفزيذ الاستعلام ١٣ (Listratio) واختباره لوحظ انه بسبب ربط الكينونة List مع Salelist في الجزء الاول و Pay\_list في الجزء الثاني، تم اهمال وصولات الاستلام والدفع وكذلك التحويلات، فحدث ان عند عرض كشف كل زبون نجد فقط قوائم الشراء والبيع.

لتصحيح هذه المشكلة تم اضافة جزء ثالث يحتوي على الكينونة List فقط، وتمت تصفية قوائم باستعمال التركيبة not exist مع استعلام فرعى على Salelist في حالة البيع، وعلى Pay\_list في حالة الشراء لجلب تفاصيل القائمة. وبهذه الطريقة يتم اضافة القيود التي ليس فيها تفاصيل إلى الاستعلام عن طريق Union.

## ٥ - التنصيب Installation

عند تنصيب النظام على حاسبة الشركة ظهرت مشكلة أن النظام لا يعمل على نظام التشغيل ويندوز أكس بي لأن 2.6 FoxPro for Windows (DBMS) مصمم على نظام التشغيل ويندوز بالإصدار ١١، ٣، ١١، ويعمل على الإصدارات من ويندوز لغاية الإصدار ويندوز ميلينيوم Windows Me. عليه يتشرط لتنصيب النظام أن لا يتجاوز نظام التشغيل الإصدار ويندوز ميلينيوم.

## ٦. التشغيل Operation

تتضمن هذه المرحلة تشغيل النظام في الشركة وتتدريب الكادر الوظيفي على كيفية استخدام النظام لكي يكون بالامكان استخدامه من قبل موظفي الشركة.

## ٧. الصيانة Maintenance

بعد البدء بالعمل على هذا النظام يجب متابعة المستخدمين لهذا النظام ومعالجة أي خطأ قد تظهر في النظام، وهذا ما يسمى بالصيانة التي تستمر لغاية انتهاء العمل بهذا النظام والبدء بنظام جديد. نقسم الصيانة إلى ثلاثة أقسام وكما يأتي:

### ٧-١ . تقييم Evaluation

في هذه المرحلة يتم تقييم عمل النظام من قبل المستخدمين وقد تظهر نقاط ضعف في النظام وهنا يجب تجاوزها.

من نقاط الضعف التي تم اكتشافها في الشاشة List. عند ادخال قائمة بيع يجب اختيار الحساب الذي يمثل الارباح في الحقل الحساب المقابل. عند اختيار أي حساب يتم عرض رصيد هذا الحساب بوساطة الدالة balance1 للدينار والدالة balance2 للدولار. وفي حالة الربح تتأخر هذه العملية.

لتجاوز هذا الضعف تم الغاء عملية حساب الربح من الدالة balance1 والدالة balance2 في حالة الشاشة List. للقيام بهذا التحكم تم اضافة متغير بعنوان m.input يحمل هذا المتغير القيمة T. عندما تكون في الشاشة List، والقيمة F. عندما تكون في

الشاشة Payment، وبهذه الطريقة اصبح عمل الشاشة List افضل بالنسبة للمستخدمين.

#### ٢-٧. صيانة Maintenance

اثناء عمل النظام ظهرت بعض المشاكل في العرض Customer والعرض CustDtl، وهو شاشة كشف الزبائن، اذ لوحظ اختلاف بين رصيد الزبون في الشاشة Payment والشاشة Customer والشاشة Custdtl. بعد متابعة الكشف تم اكتشاف وجود قوائم بأسماء زبائن غير موجودين. بالتأكيد لا يمكن ادخال قائمة برقم زبون خطأ! ولكن يمكن بعد ادخال القائمة حذف الزبون.

هذه الحالة أدت الى عدم ظهور القائمة في كشف الزبون، ولحل هذه المشكلة تم استخدام الاستعلام ٢٧ بدلًا من الجدول Customer1 .

```
SELECT * FROM CUSTOMER;
UNION;
SELECT ... FROM list;
WHERE NOT EXIST ;
( SELECT * FROM customer ;
WHERE customer.code=list.custoemr );
INTO CURSOR Customer1
```

الشكل ١٨ الاستعلام ١ Customer1

- الاستعلام ٢٧ Customer1: يتم من خلال هذا الاستعلام الكشف عن زبائن لهم قوائم وغير موجودين في كيغونة الزبائن. استخدم في هذا الاستعلام NOT EXIST للكشف عن الزبائن ودمجه مع باقي الزبائن باستخدام UNION كما في الشكل (٣,١٧). هذه طريقة بديلة عن LEFT OUTER JOIN اذ إن Foxpro for Window 2.6 لا يحتوي على هذه الخاصية.
- الاستعلام ٢٨ Subject1: كذلك يمكن أن تظهر المشكلة نفسها في المحرر، لذلك تم اضافة هذا الاستعلام بديلاً عن الجدول Subject ، لتحديد القوائم التي تحتوي على ارقام مواد غير موجودة ضمن كيغونة المواد واكتشافها في كيغونة Salelist أو pay\_list. هذا الاستعلام مقارب للاستعلام ٢٧.

#### ٣-٧. تحسين Enhancement

بعد عملية تقييم النظام وعملية إجراء الصيانة، تأتي عملية اضافة التحسينات أو التطوير على النظام. لغرض تحسين النظام في مجال حساب الديون، اذ لوحظ أن عملية حساب الديون تأخذ بعض الوقت. بعد تدقيق العرض Payment تم اكتشاف سبب التأخير، وهو استخدام البرمجة الاجرامية (التقليدية) في حساب الديون بدلًا من استخدام SQL بشكل كامل.

- حل هذه المشكلة تم اعادة صياغة العرض Payment بحيث عُوض عن تنفيذ الدالتين Balance1 و Balance2 ببعض الاستعلامات وكما يأتي:
- الاستعلام ٢٩ **Lastbalance**: لغرض تصفية القوائم بحيث يعطي فقط القوائم المرتبطة بالحسابات المطلوبة اعتماداً على نوع الحساب `m.important` (٠، ١، ٢) وكلاء حسابات شخصية). من خلال هذا الاستعلام يتم تقليل القوائم الداخلة في المعالجة.
  - الاستعلام ٣٠ **Balance**: يحتوي على مجموع كل نوع من القوائم لكل حساب بالأعتماد على الحقل `List_type` وكما يأتي قائمة شراء، ١ فائمة بيع، ٢ وصل استلام، ٣ وصل دفع، ٤ تحويل. كما تم الربط بين المجاميع باستخدام `UNION`.
  - الاستعلام ٣١ **Benfit1**: لتصفية قوائم البيع فقط التي فيها ارباح `= receiver!`.
  - الاستعلام ٣٢ **Benfit**: لحساب ارباح القوائم التي تم تصفيتها في الاستعلام السابق. باستخدام الدالة `calbenfit`. في حالة عدم وجود قوائم بيع هذا الاستعلام لا يأخذ أي وقت، لأن تنفيذ الدالة `Calbenfit` تتطلب وقتاً ليس بالقليل.
  - الاستعلام ٣٣ **Balance**: لغرض دمج استعلام الارباح مع استعلام `(٣)` `balance` السابق عن طريق `UNION`.
  - الاستعلام ١ **Payment**: هو الاستعلام النهائي في هذه الفقرة الذي يتضمن اسم الحساب (او الزبون) (`cname`), الرصيد من الدينار (`balance1`), والدولار (`balance2`), مجموع ماله من الدينار (`creditid`) والدولار (`creditud`), ومجموع ما عليه من الدينار (`debitid`) والدولار (`debitud`). في هذا الاستعلام تم حساب المجموع باستخدام `SUM GROUP BY` فضلاً عن `BY`. وبهذه الطريقة تم حساب الديون بوقت قياسي، وبالاخص في حالة عدم وجود حساب ارباح، والنتيجة النهائية تم تحسين عمل النظام.

## ٨. الاستنتاجات

خرج البحث بالاستنتاجات الآتية:

١. الفرق الكبير بين البرمجة الاجرائية والبرمجة غير الاجرائية ( باستخدام SQL).
٢. يتم استخدام البرمجة الاجرائية فقط في الحالات المعقدة جداً بحيث لا يمكن تطبيقها على SQL.
٣. ضرورة متابعة النظام بشكل دقيق لاكتشاف الاخطاء التي من الصعب اكتشافها من دون التنفيذ التدريجي للشفرة (Trace).
٤. الجهد المبذول لصيانة النظام لا يقل اهمية عن الجهد المبذول في تحليل وتصميم وتنفيذ النظام. واهمال الصيانة قد يؤدي الى فشل النظام حتى في حالة كون التصميم جيد.

#### ٨. المراجع

#### أولاً - المراجع باللغة العربية

١. منيب احمد قطيشات، "قواعد البيانات"، الدار العربية، ١٩٩٩.

#### ثانياً - المراجع باللغة الاجنبية

1. Alan Dennis and Barbara Haley Wixom, "SYSTEM ANALYSIS AND DESIGN", John Wiley and Sons, Inc, 2000.
2. C. J. Date, "An Introduction To DATABASE SYSTEMS", ADDISON-WESLEY, 2000.
3. Fred R. McFadden, Jeffrey A. Hoffer, and Mary B. Prescott, "Modern Database Management", ADDISON-WESLEY, 1999.
4. Peter Rob and Carlos Coronel, "DATABASE SYSTEMS Design, Implementation, And Management", Course Technology, 2000.