

By The Coder in April 07

لكن لا بأس بالاعتباس منها ومن طريقة عملها لمصلحتنا ، ولكن بحدود ، فبرنامج خاص بالأسهم يقرر ماذا يشتري وماذا يبيع .. يعني برنامج خاص بالأسهم ومصائبها ومكاسبها وكل ما يتعلق بها فقط لا غير ..
إذا زدنا عليه برنامج يقرر المخطوات المقترحة في تنفيذ مشروع ما (وهذه أكيد تحتاج لشغل معقد) ، ستبدأ سرعته تقل بشكل عجيب لدرجة العجز ، في ذلك الوقت سوف تعرف قوة العقل ومدى ضعف الكمبيوتر في تلك المعالجات المعقدة .

الآن لو قلت لك ... ماذا يعني لك اللون الأحمر
سيقول احد الرومانسيين "يرمز للحب طبعاً " ... ترجع لخبراته نجده دائماً يحب ان يهدي الورد الحمراء لمن يحب
وسيقول آخر "يرمز للموت" ... ترى انه معقد من صغره وكانت بلدته في حروب دائمة... لذا ارتبط هذا اللون بالدم والموت
وسيقول آخر سطحي جداً "لاشيء" ... لأنه لا يتفكر في الحياة بعمق

بعد هذه المقدمة الطويلة عن العقل أريدك ان تخبرني ، قال احدهم في كتابه "قواعد البيانات هي مجموعة متكاملة من البيانات"
ولم يشرح ما قصد منها ، وقرأ كتابه
١٠٠٠٠٠ شخص مبتدئ ، كم معنى سوف يتم إصداره من هؤلاء الأشخاص ؟
الكاتب في كثير من الأحيان معذور ، فهو بروفييسور ويتكلم على قد عقله ولا ينزل الى مرتبة المبتدئين ولن ينزل طبعاً ، فهو رمى كلمة كبيرة كـ "متكاملة"
والتي لها معاني كثيرة ثم ذهب .

اعرف انه يوجد أمور متعارف عليها عالمياً ولكن ، أكيد لها معنى خاص لكل منا ... مثلاً الذكاء ، أتحدى أي عالم ان يصف لي ويعرف لي هذه الكلمة بكل ما تعنيه ،
تجد في كل كتاب تعريف مختلف تماماً عن الآخر .
كل شخص لازم يغلط هذه الغلطة ، مستحيل ما يغلط ، فاعذروني على التقصير وأحاول ان أبين مالذي اعنيه بالضبط كي لا تهز رأسك وتقول "فاهم فاهم" وأنت لست فاهم ، فلو ساورك أدنى شك في معنى أنا كتبتّه لا تتردد اسأل

*** بداية الدرس :**

ذكرنا بعض التعاريف لقواعد البيانات ، واستعرضنا بعض المفاهيم الأساسية التي تخص قواعد البيانات العلائقية ، ومنها المفتاح الأساسي والمفتاح الأجنبي والعلاقات المفتاح الأساسي لماذا ؟ هل تذكر ؟ لكي نفرق بين الصفوف . المفتاح الأجنبي لماذا ؟ لكي نربط جدول بجدول آخر .

اليوم سأعرض المزيد من المفاهيم والقواعد عن قواعد البيانات العلائقية بشكل مفصل أكثر .

لكن أريد أن أذكر إلى تعريف جميل لقواعد البيانات وجدته :

مجموعة من المعلومات المنظمة بشكل يسهل على أي برنامج الحصول على البيانات المطلوبة بشكل سريع أي معلومات منظمة تجعل أي برنامج طلب البيانات أن يحصل عليها بشكل سريع ، أتساءل هل تساءلت وقلت "ما قصد بهذه البرامج ؟ من هي هذه البرامج التي تطلب البيانات ؟ و ما البيانات المطلوبة أصلا ؟" هذه دروس متقدمة قليلا عن البرمجة وربط البرمجة بقواعد البيانات ، لعلني استعرضتها هنا لذوي الخبرة المتوسطة ، ولعلني جعلتك تتعلم كيف تتساءل عن أي جملة تمر عليك .

اذكر بأنني استعرضت معك قاعدة البيانات المسطحة ، وتوجد بها مجموعة مشاكل منها عدم دعم المستخدمين المتعددين (سوف اشرحها لاحقا) ، وصعوبة استرجاع ومعالجة البيانات ، ومشاكل التكرار ، أيضا صعوبة التعامل مع البيانات المعقدة وغيرها ... اذهب واسترجعها .

*** مفاهيم مهمة جدا لقواعد البيانات :**

اذكر أنني عرضت درجة العلاقات ، وذكرنا بعض الأمثلة ، دعنا نستعرض درجات العلاقات ثنائية

١- one to one

٢- one to many or many to one

٣- many to many

هناك بعض القوانين لقواعد البيانات العلائقية التي اقترحها العالم Codd ، والتي تضمن لنا قاعدة بيانات ممتازة لا أخطاء فيها ، لن أتكم عنها بذلك التفصيل ولن اسردها ، ولكن سأعطيك خلاصتها في هذا الدرس لشيء ما في نفسي .

في الدرس السابق فصلنا جدول الموظفين و جدول الأقسام ، صحيح ؟ نسيت ؟؟ راجع الصورة رقم L2_P7 في الدرس الثاني لا أريد أن أقول "لماذا فصلناهما " ، اعتقد تكلمت كثيرا عن عملية الفصل هذه ، قبل أن تنتظر إلى الصورة التالية ، سوف اعرض من الآن وصاعدا معظم أمثلي وتسمياتي بالانجليزي ، لا تسأل لماذا .

الآن انظر إلى الملف المسطح التالي ، وهو جدول الموظفين بأقسامهم :

الصورة L3_P1

e_no	e_name	e_sal		d_name	d_loc	
74758	ahmed	7500		account	jeddah	
78312	yasser	8500		IT	dammam	
54289	khaled	4560		account	jeddah	
45368	sami	5530		account	jeddah	
96385	othman	6000		sales	riyadh	
78925	ali	7500		sales	riyadh	
54124	basem	7500		IT	dammam	

إليك الترجمة :

E_no رقم الموظف وهي اختصار لـ Employee's number

E_name اسم الموظف

E_sal وهو راتب الموظف sal=salary

كذا ما حنخلص ،،، اشترى قاموس ، ركب برنامج الوافي ، احضر خبير انجليزي ، تصرف عاد

D_name اسم القسم

D_loc موقع القسم

IT أي قسم تقنية المعلومات

Account أي قسم المحاسبة

Sales أي قسم المبيعات

مالذي تعنيه النقاط بالصورة ؟

تعني ١٠٠٠ حقل ... ليس لي الطاقة بكتابة هذه الحقول ، فتخيلها يعني .

النقاط الأولى التي باللون الأحمر هي حقول خاصة بالموظفين ، النقاط الأخرى هي حقول خاصة بالأقسام ... وكما قلت ممكن ١٠٠٠ حقل وممكن أكثر من ذلك أو أقل ... فقط تخيل .

لم تفهم مسألة النقاط ؟ دعك منها الآن تجاهلها ، ولكن تخيل بدلا منها انه يوجد بعض الآلاف من الحقول .

نفصلهم كما اتفقنا ، على أي أساس ؟ على أساس شيء يسمونه بالكائن أو الكيان ، Entity

قال تعالى : (إِنَّمَا قَوْلُنَا لِشَيْءٍ إِذَا أَرَدْنَاهُ أَنْ نَقُولَ لَهُ كُنْ فَيَكُونُ) (النحل: ٤٠)

كان يكون كن كينونة حيث الكيان والكائن حينما يكون ، وهو كل كائن كان إنسان أو حيوان ، أو حتى لو كان من الجان .
أحب أن اسميه بكائن ، ولا ادري هل ترجمة Entity تعني فعلا كائن ؟ لأن الوافي يقول لي كيان وليس كائن ، لكن نتفق بأنني اقصد قول بالـ Entity هو الـ Entity
سوف اشرح الكائنات لاحقا ، الآن عندنا كائنين هما كائن الموظفين ، وكائن الأقسام .
في عملية الفصل هذه حللنا مشاكل كثيرة ، أريد أن اذكر بشكل سريع ثلاث منها ، مشكلة الإضافة ، والحذف والتعديل ...
مشكلة الإضافة ، لو أضفنا الموظف أسامة في قسم تقنية المعلومات فسوف نكرر معلومات قسم تقنية المعلومات ثانية ، ولو أضفنا ١٠٠٠ موظف في نفس القسم نعيد ونكرر نفس بيانات القسم
مشكلة التعديل ، لو عدلنا في قسم المحاسبة ، يجب أن نعدل كل البيانات المكررة ، وهذا يجهدنا ويضيع وقتنا و بلخيطنا .
مشكلة الحذف .. لو نريد نحذف قسم المحاسبة فهذا يعني أن نحذف وجود الموظف أيضا ، نحن نريد أن نحذف القسم وليس الموظف !

الآن انظر إلى عملية الفصل كيف تمت :

الصورة L3_P2

Emp_table					Dept_table			
e_no	e_name	e_sal	d_no		d_no	d_name	d_loc	
74758	ahmed	7500	10		10	account	jeddah	
78312	yasser	8500	20		20	IT	dammam	
54289	khaled	4560	10		30	sales	riyadh	
45368	sami	5530	10		1000 records here			
96385	othman	6000	30					
78925	ali	7500	30					
54124	basem	7500	20					
1000000 records here								

الآن جدول الموظفين Emp_table ، و جدول الأقسام Dept_table
انظر إلى النقاط التي في جدول الموظفين ، يعني ألف حقل تخصص الموظفين ، ماذا يلبس عادة ، وما طوله ، وعرض أكتافه وهل متزوج أو مطلق أو أرمل ؟ وما هي سيارته الحالية وما تاريخ إنتاجها ؟
هيا إليك السؤال الأول من الواجب و هو رسم جدول به ١٠٠٠ حقل للموظفين !!

مجرد دعابة ،،،،

انظر إلى جملي 1000000 records here ، 1000 records here ، أي يوجد ١٠٠٠٠٠٠ سجل هنا (سجل =صف) ، ويوجد ١٠٠٠ صف هنا ، ولا طاقة لي في كتابتها فتخيل اتفقنا ؟
أشير إلى أنني مرة اقول سجل ومرة اقول صف ، لا ادري ولكن كلاهما معنى واحد ، فلا تلخبط .

الآن ... مالعلاقة بين الأقسام والموظفين ؟ واحد لمتعدد ، الموظف الواحد يعمل تحت قسم واحد ، والقسم الواحد يعمل تحته عدة موظفين ... لاحظ ما زلت أتكلم في هذه العلاقة فقط ... اكرر ... في هذه العلاقة فقط !
دعني اسمي جدول الموظفين (الجدول المتعدد) لأنه من جهة المتعدد الذي في العلاقة واحد لمتعدد
واسمي جدول الأقسام (الجدول الواحد) لأنه من جهة الواحد الذي في العلاقة واحد لمتعدد اعلم أنني أزعتك بهذه التسميتين ، وتقول "أعطيني أمثلة أكثر ، مافهمت شي الخ"
القسم الواحد يعمل به (مجموعة موظفين) إذن جدول الموظفين هو جدول المتعدد ! وقس على ذلك
اسف ولكن مضطر إلى بعض التسميات ، واليك تسمية أخرى ... يجب أن تعلمها ... الجدول المتعدد (كجدول الموظفين) يسمى أيضا جدول الأبناء ، والجدول الواحد (كجدول الأقسام) يسمى جدول الاباء

هل لاحظت المفاتيح الأساسية ماهي في الجدولين ؟ Primary Keys ،
إذن لماذا المفاتيح الأساسية ؟ ولو كررناها مالذي سيحدث ؟ ولو جعلناها قيمة خالية ، مالذي سوف يحدث ؟ اعتقد أنني شرحت ذلك ،
للتذكير المفتاح الأساسي مجرد حقل في الجدول ولكن له مواصفات خاصة وهي انه لا يتكرر ولا يمكن أن يكون خاليا ، وغرضه هو التمييز بين صف وصف ، ما معنى التمييز ؟ استقلالية كل صف ، مثال عليه كما قلنا سابقا
نريد ترقية ابن عمنا الموظف احمد سالم وليس الموظف احمد سالم الآخر ، كيف نميز بينهما في قاعدة بياناتنا ؟؟

*** المفتاح الأجنبي Foreign Key يسمى أيضا المفتاح الغريب ، و اسميه بعض الأحيان الحقل الأجنبي ، فلا تلخبط .**

ماغرضه ؟ ربط جدولين ، هيا يا صاحبي ، في الصورة L3_P2 انظر إلى الموظف Ahmed ، في أي قسم الرجل ؟ في قسم رقم ١٠ صحيح ؟
هيا في ورقة عندك أو في برنامج الرسام ، أو حتى لو أتيت بقلم ورسمت في جبهتك ، ارسم جدول الموظفين و الأقسام ولكن بدون المفتاح الأجنبي Dept_no الموجود في جدول الموظفين ، الآن يا صاحبي الموظف احمد في أي قسم ؟ كيف نعرف ؟ اعتقد انها مسألة لا تحتاج ذلك الذكاء ، حقل لا شأن له بجدول الموظفين ، استخدمناه لربط جدول الموظفين مع جدول الأقسام ... بقي عليك أن تعرف انه سمي بالأجنبي لأنه فعلا أجنبي عن الجدول نظريا ، فهل رقم القسم صفة ملاصقة للموظفين ؟ دعك من هذا ، واليك قوانين الحقل الأجنبي .
الحقل الأجنبي لا يصح أن يكون خاليا ، يجب أن يطابق المفتاح الأساسي الموجود في الجدول الآخر ، يجب أن يحتوي على قيم موجودة في سجلات المفتاح الأساسي الموجود في الجدول الآخر ، و أخيرا ممكن أن يتكرر في حالات ولا يتكرر في حالات أخرى ، على حسب درجة العلاقات
اعرف أنني صرت ارمي في الكلام ولا أبين ما عنيته ، الآن سوف اشرحها بشيء من التفصيل ، بدون هذه التفاصيل فسوف لن تفقه شيئا في قواعد البيانات .

كقاعدة وضعها كود Codd لكي نبني قاعدة بيانات خالية من الأخطاء ، طبعاً فيه حالات سوف نحلها برمجياً ولكن أنا أتكلم عن قاعدة البيانات كبناء .. في هذا الدرس سوف نبني قاعدة بيانات ، وسوف أتكلم عن ذلك .

مممم ، اكره أن أتلقى الأوامر والتعليمات والاقتراحات والقواعد اعتقد أنك تكرهها أنت أيضاً ، لكن دعنا نتذاكى ونرى لماذا قال Codd ذلك ؟ أعيد سرد بعض القوانين ، الحقل الأجنبي لا يكون خالي ، لا يتكرر ويتكرر على حسب العلاقة ، يجب أن يكون نفس نوع المفتاح الأساسي ، يجب أن يحتوي على نفس السجلات الموجودة في المفتاح الأساسي (في الجدول الآخر).

انظر إلى هذا المثال وهذه الصورة ، وقد تفننت في أنواع "التطنيش" لهذه الاقتراحات ، وسوف أرى مالذي يحدث هنا الآن قبل أن تشاهد الصورة التالية ، انظر إلى الصورة السابقة رقم L3_P2 ، ثم خذ ورقة ، أو افتح الرسام ، أو تخيل في رأسك مايلي :

- في جدول الأقسام :

عدل رقم قسم تقنية المعلومات إلى ٢٢

احذف قسم المبيعات نهائياً

أضف قسم جديد وسمه قسم التسويق ورقمه ٥٠

- في جدول الموظفين :

اجعل قيمة المفتاح الأجنبي الخاص بأحمد قيمة خالية

غير رقم قسم خالد إلى الرقم ٦٦

انظر إلى الصورة التالية رقم L3_P3

Emp_table					Dept_table			
e_no	e_name	e_sal	d_no		d_no	d_name	d_loc	
74758	ahmed	7500			10	account	jeddah	
78312	yasser	8500	20		22	IT	dammam	
54289	khaled	4560	66		50	marketing	makkah	
45368	sami	5530	10		1000 records here			
96385	othman	6000	30					
78925	ali	7500	30					
54124	basem	7500	20					
1000000 records here								

يظهر أن Codd سيصنف قاعدة بياناتنا هذه كأسوأ ما يكون ، بناء خاطئ ١٠٠% انظر إلى الحالات السبع ، وهي تلخيص لـ ٣ أو ٤ محاضرات :

- الحالة الأولى في جدول الأقسام :

عدّلنا رقم قسم تقنية المعلومات إلى ٢٢ ، مالذي حدث ؟ هل ترى أرقام أقسام الموظفين باسم و ياسر (الكتابة الخضراء بالصورة) ؟ ماذا حدث معهما ؟ يعملان في قسم لا وجود له ، رقم هذا القسم هو ٢٠ بحثنا وبحثنا عنه في جدول الأقسام ، ولا أثر ... إذن الموظفين ياسر وياسم ... منبوزين في نظامنا هذا هذه الظاهرة تسمى بالسجلات اليتيمة ، مثال شهير عليها جدول الآباء و جدول الأبناء ، جدول الأبناء به الحقل الغريب ، يوجد ثلاث أبناء رقم آبائهم غير موجود في جدول الآباء ، هنا نسمي هذه السجلات "سجلات يتيمة" اسمحو لي أن اسمي أي جدول في حالة جدول الأقسام (جدول الواحد) جدول الآباء ، وأي جدول في حالة جدول الموظفين (جدول المتعدد) جدول الأبناء .

أعيد ،، جدول الآباء (جدول الأقسام في مثالنا) ، و جدول الأبناء (جدول الموظفين في مثالنا) ... جدول الأبناء به مفتاح أجنبي اسمه "رقم الاب" (مثل رقم القسم الموجود في جدول الموظفين في مثالنا) ...

يوجد ابنين اثنين رقم أبوه ليس موجود في جدول الآباء (كحالة ياسر وياسم) ، إذن كما اتفقنا .. سجلات الابنين الاثنين تسمى بالسجلات اليتيمة (سجلات باسم وياسر هي سجلات يتيمة)

السجلات اليتيمة ما مصيبتها ؟ تخلخل موازين النظام بأكمله ، صعب التعامل معها صعب حل مشكلتها والأدهى من ذلك .. صعب الكشف عنها ، أنا أتكلم عن الأنظمة الكبيرة طبعاً .

إذن مالحل مالعمل ؟ نمنع ظهورها من الأساس طيب من يساعدنا في منع ظهورها ؟

الـ DBMS طبعاً (نظم إدارة قواعد البيانات .. هل تذكرها ؟ راجع الدروس السابقة !) ، أمثلة على الـ DBMS الاوراكل الذي ندرسه Oracle ، أو Microsoft SQL Server ، أو Microsoft Access

وغيرها الكثير ، لعلي أتكلم في هذه النظم فيما بعد ،

هذه النظم تساعدنا في منع ظهور هذه السجلات اليتيمة ، مثلاً لو غيرنا قسم تقنية المعلومات من ٢٠ إلى ٢٢ ، مالذي سيحدث ؟

هذه النظم تقول مايلي : "هيه انتبه ! مالذي تفعله يا هذا ؟ أنت تعدل في جدول الأب ، سوف تيتّم أبناءه ... احذر يا هذا ، ما رأيك ؟ هل نلغي الأمر أو نسوي المسألة ؟ "

كيف يسوي المسألة ؟ بتغيير كل قيم الحقول الأجنبية الخاصة بالأبناء التابعين للأب المراد تعديله إلى الرقم الجديد .

سوف ندرس هذه الأمور فيما بعد بشكل تطبيقي أن شاء الله .

- الآن انظر إلى الحالة الثانية في جدول الموظفين :

جعلنا رقم قسم الموظف أحمد قيمة خالية !

مالذي سوف يحدث ؟؟؟ سجل يتيّم آخر ، نفس الشيء يساعدنا في ذلك الـ DBMS ... يقول "لا تجعل هذه القيمة خالية .. هل تريد أن نيتّم السجل ؟؟ "

الآن فيه مصطلحين بسيطين ، هما اختياري (Optional) و إجباري (Mandatory) .
احفظهما جيدا ، لا ادري لماذا في بعض المراجع يقومون بتجاهلها ، بالرغم من أهميتها ، لكن اعتقد أنهم تجاهلوا فقط ليشرحوها فيما بعد لا ادري إليك أمثلة

كنت أقول "العميل الواحد يفتح حساب واحد ، والحساب الواحد يخص عميل واحد"
الآن لنطور الجملة أكثر "العميل الواحد من الممكن أن يفتح حساب واحد ، والحساب الواحد يجب أن يخص عميل واحد"
هاه ، أين الأدكيا ؟ مالذي اثار انتباهك ؟

العميل يفتح حساب اختياري ...

لكن ، الحساب يخص عميل اجباريا ...

يعني العميل براحته ، بكيفه ، هو حر ، يفتح حساب أو لا يفتح ، لكن إذا حصل وتم فتح هذا الحساب ، خلاص ! مستحيل الحساب أن لا يخص أي عميل ... يلزم بل ضروري بل يجب أن يخص عميل !!

اعرف أن البعض هنا لم يفهم ، اعرف أن البعض يريد أمثلة أخرى ، والبعض الآخر يريد أن يتفلسف ، وهناك شريحة من المجتمع لغويين وفهموا كل ما قلت ...

اقول ، سوف أزيد أمثلة ، وسوف ارسمها على الورق كي تصدقوا ، وسوف أتفلسف أكثر لكي اضمن فهم الجميع .

وكمعلومة جانبية ، إذا كنت مندوب مبيعات ، أو أي شخص لديه محل بيع وتريد أن تقنع الزبون ، افهمه أولا ، بعضهم يحتاج إلى خطوات منطقية كي يفهم ويقتنع ، و آخر عبيط مستحيل أن تقنعه لذا ابدأ وظهر كل عيوب بضاعتك لكي يقتنع

والبعض الآخر يلزمه الوقت .. يعني ماطله واجعله يفكر أكثر ويجلس عند السلعة لكي يقتنع ، وهناك من يشبه سيدنا إبراهيم عليه السلام قال تعالى : (وَإِذْ قَالَ إِبْرَاهِيمُ رَبِّ أَرِنِي كَيْفَ تُحْيِي الْمَوْتَىٰ قَالَ أَرَأَيْتَ إِنْ أَتَيْنَا بِسَعْيٍ أَلَمْ تَكُنْ تُؤْمِنُ قَالَ بَلَىٰ وَلَكِنْ لِيَبْظُنَّ قَلْبِي قَالَ فُخِّدْ أَرْبَعَةً مِنَ الطَّيْرِ فَصُرْهُنَّ إِلَيْكَ ثُمَّ أَجْعَلْ عَلَىٰ كُلِّ جَبَلٍ مِنْهُنَّ جُزْءًا ثُمَّ ادْعُهُنَّ يَأْتِينَكَ سَعْيًا وَاعْلَمْ أَنَّ اللَّهَ عَزِيزٌ حَكِيمٌ) (البقرة: ٢٦٠)

قال أولم تؤمن ؟ قال بلى ... ولكن ليظمن قلبي ...

بعضهم يريد ان يظمن قلبه ويريد أن يرى ، يلمس ، يتحسس السلعة كي يقتنع حتى لو لم يفقه ما هي مميزات السلعة ، فقط يريد أن يرى بوضوح ...

ونتعامل مع هذا النوع بعرض الصور كي يرى الجداول والعلاقات وما إلى ذلك

شيء جميل أن نشرح بالصور ، والشيء الأجمل هو أن معظم البشر يحبون أن يرون ، ويحبون أن يفهمون بالنظر فالصور شيء جميل للشرح ، وعرض الفلاش رائع ، والفيديو أجمل وأجمل ، ولو أن النت سريع ، لعلمي اعمل الدروس بالفيديو ، لكن .. إن شاء الله في عام ٢٠٦٥ ميلادي .

الآن نرجع إجباري ، واختياري ... مثال آخر .

الموظف يجب أن يتبع قسم (إجباري)، القسم من الممكن أن يحتوي على موظف أو أكثر (اختياري).

قطعة الغيار الواحدة ممكن أن تحتوي على قطعة غيار أخرى(اختياري) ، والقطعة الغيار الأخرى من الممكن أن تتكون من قطعة غيار أخرى(اختياري) . (هذه صعبة الفهم حقا كيف تكون علاقة بين جدول القطع نفسه ؟ .. سنشرحها لاحقا)

الموظف يجب أن يملك رخصة واحدة(إجباري)، والرخصة يجب أن تخص موظف واحد فقط(إجباري).

عرضت كل الاحتمالات تقريبا .. إجباري إجباري ، إجباري اختياري أو اختياري إجباري ، اختياري اختياري .

مالغرض من إجباري اختياري ؟ لماذا نستخدمها ؟ سؤال غريب ، و الإجابة ستكون اغرب ... ستكون الإجابة مبهمة الآن لذا سوف أجاب السؤال فيما بعد .

الآن عرضنا درجات العلاقات ، وعرضنا إلزامية العلاقة ام إجبارية

الآن درجات العلاقات نستعرضها رسوميا .. لعلك تتعمق في فهمها .

نعرض العلاقات ودرجاتها وهل إلزامية أو لا ، للذين لا يحبون الرياضيات ، البرمجة رياضيات ، تريد أن تكون مبرمج ، كن رياضيا (رياضيات وليس رياضة ،) ، هيا اقرأها لا تأخذ معك أكثر من ربع ساعة :

الصور الأربع كتلخيص لحالاتنا ، والسؤال لماذا أربع صور ؟ لأنه لدينا أربع حالات

One to many

Many to one

Many to many

One to one

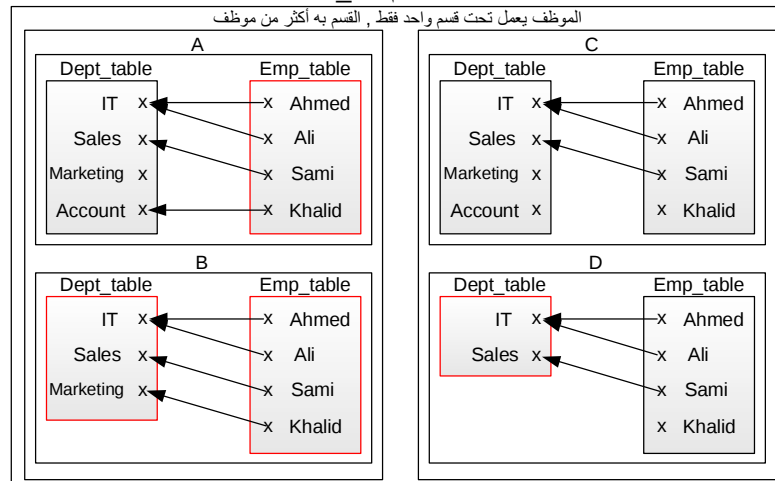
... بالرغم من الثانية والثالثة نفس الفكرة ، ولكن نظريا ليستا نفس الشيء

المشكلة أن لكل حالة من الحالات الأربع ... بها اربع حالات ... أي سوف استعرض ١٦ حالة لحسن الحظ كلها متشابهة في الفكرة ، إذا فهمت أول ٤ حالات ... سوف تفهم البقية .

الآن بخلاف أن يكون درجة العلاقة بين القسم والموظف صحيحة أو خاطئة ، تجاهل هذه النقطة ، الأصل هي أن الموظف يعمل تحت قسم واحد ، القسم به أكثر من موظف لكن أردت إيضاح الفكرة الرئيسية في مثال واحد وهو القسم والموظف

الإطارات ذات اللون الأحمر تعني إلزامي ، والتي ذات اللون الأسود تعني اختياري !

الصورة رقم L2_P4



توضح هذه الصورة علاقة واحد لمتعدد .. أي الموظف الواحد يعمل تحت قسم واحد ، والقسم يعمل تحته أكثر من موظفين

في الإطار A

لغويا

الموظف يجب أن يعمل تحت قسم واحد ، القسم الواحد ممكن أن يعمل به أكثر من موظف

هل ترى احد وعلي كلاهما في قسم IT ؟ يعني من الممكن أن قسم المبيعات أيضا يحتوي على أكثر من موظف ، لكن في مثالنا يوجد موظف واحد ، وانظر إلى قسم التسويق ، لا يوجد به أي موظف ... لأنه العلاقة ماذا تقول ؟

تقول في الإطار A أن الموظف يجب أن يعمل تحت قسم واحد فقط ، لكن القسم ممكن يعمل تحته أكثر من موظف (وممكن لا !) ، لاحظ إطار الموظفين لونه احمر أي يجب أن يحتوي على قيمة

طبعا هذا مثال قليل الموظفين والأقسام ، فتخيل إذا كانوا ألف موظف و مئة قسم ، وكل قسم ممكن يعمل تحته عشرين موظف ، أي ممكن يصله عشرين سهم أو أكثر

....

رياضيا

يجب أن ينطلق سهم واحد فقط من جدول الموظفين ، وممكن أن يصل سهم أو أكثر (وممكن لا يصل شيء) في جدول الأقسام

الإطار B

لغويا

الموظف يجب أن يعمل تحت قسم واحد ، القسم يجب أن يعمل به موظف أو أكثر .

رياضيا

يجب أن ينطلق سهم واحد فقط من جدول الموظفين ، ويجب أن يصل سهم أو أكثر إلى جدول الأقسام

الإطار C

لغويا

الموظف ممكن أن يعمل تحت قسم واحد ، القسم ممكن أن يعمل به موظف أو أكثر

رياضيا

ممكن أن ينطلق سهم واحد فقط من جدول الموظفين ، وممكن أن يصل سهم أو أكثر إلى جدول الأقسام

الإطار D

لغويا

الموظف ممكن أن يعمل تحت قسم واحد ، القسم يجب أن يعمل به موظف أو أكثر

رياضيا

ممكن أن ينطلق سهم واحد فقط من جدول الموظفين ، ويجب أن يصل سهم أو أكثر إلى جدول الأقسام .

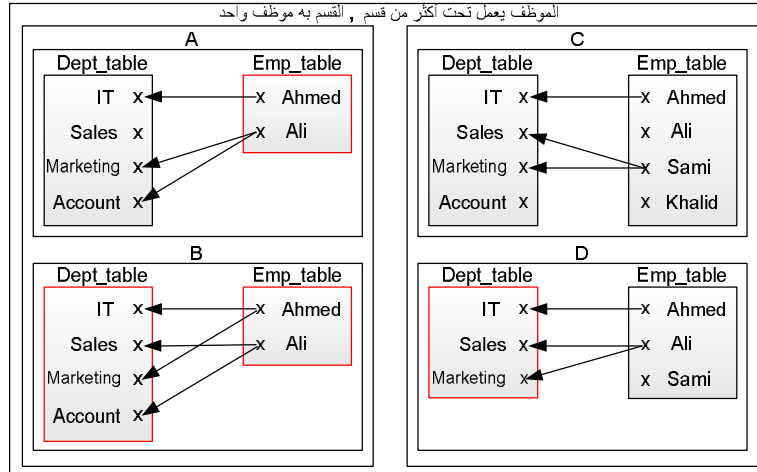
الآن

هل ترى الصور L3_P5 , L3_P6 , L3_P7 ؟؟؟؟؟؟؟

الواجب صف لي كل اطار في كل صورة من تلك الصور الثلاث رياضيا ولغويا .. لا تنسى .. كل صورة بها أربع حالات ، ABCD

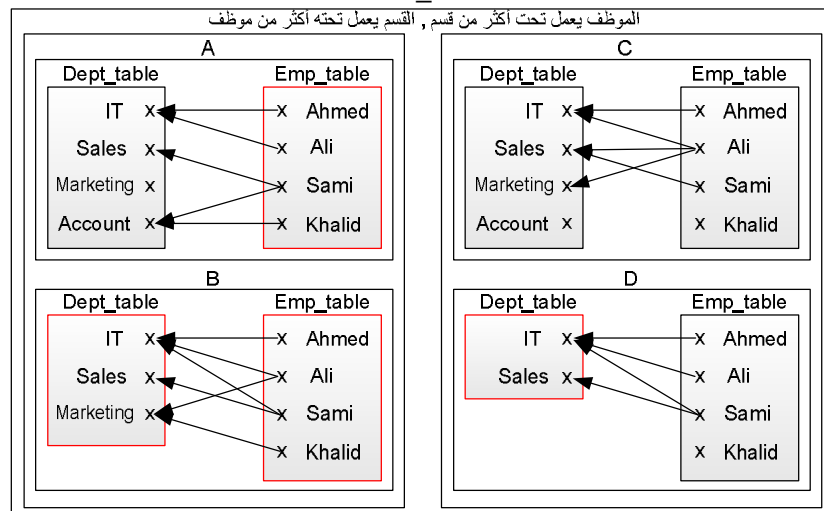
L3_P5

الموظف يعمل تحت أكثر من قسم , القسم به موظف واحد



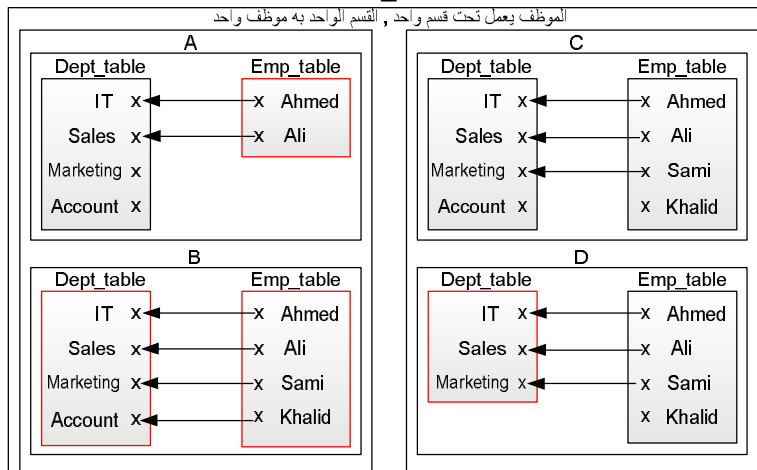
L3_P6

الموظف يعمل تحت أكثر من قسم , القسم يعمل تحته أكثر من موظف



L3_P7

الموظف يعمل تحت قسم واحد , القسم الواحد به موظف واحد



الآن حل الواجب لعلك تنتبه لشيء لم تفهمه ، هي سهلة لكنها ثقيلة الدم ، قبل أن نبدأ في تصميم قواعد البيانات ، وقبل أن نتوسع في العلاقات وقبل أن نتكلم أين ومتى نضع المفاتيح الأجنبية أريد أن أتكم عن بناء وتطوير النظم وما المراحل الأساسية المتبعة .

* بناء تطوير النظم :

ما قصدي بناء و تطوير النظم ؟ ماهي النظم ؟ وكيف يتم تطويرها و بناءها أصلا؟؟؟
النظم هي مجموعة برامج ، هل تعرف ما البرامج ؟ هي التي نريد أن نعملها ، نريد أن نبرمجها
طبعاً من يحتاج البرامج ؟ الشركات والمؤسسات ... وربما أشخاص أو بضع أشخاص هؤلاء الذين يريدون البرامج نسميهم مستخدمي الأنظمة أو مالكي الأنظمة ...
سوف استخدم هذه المصطلحات كثيراً لا تنسى ذلك ، احفظها الآن ...
الأنظمة اقصد بها ماذا ؟ الأنظمة أو النظم ومفردها نظام ، مجموعة برامج عادة تكون كبيرة ، هذه البرامج تعمل سوية بانتظام لكي تؤدي غرض واحد ، لكي تعمل عمل واحد ، برنامج متكامل وشامل واحد ،
لو تريد تضخيم برنامج ما ... قل عنه نظام .
مثلا البرنامج الكبير Microsoft Windows ... نسميه نظام Microsoft Windows ، أو برنامج المبيعات ... نسميه نظام المبيعات ، وهكذا
طبعاً كلمة نظام هي كلمة كبيرة جداً ، ولها عدة استخدامات هذه الكلمة ، فسوف أخبرك ما معنى كلمة نظام في كل مرة مختلفة ، لكن عادة حينما أقولها اقصد بها البرنامج ، اتفقنا؟؟
أنت الآن صاحب نظام ، تريد نظام جديد .. من يعمل لك هذه الأنظمة ؟ المبرمجين ، تذهب عند احدهم وتخبره بماذا تريد الآن الأشياء والأمور التي تريدها في نظامك ماذا نسميها ؟ نسميها متطلبات requirement ، نسميها ماذا ؟ متطلبات النظام .سوف أتكم عن هذه المتطلبات بعد قليل .

الآن لعلك ستلاحظ كلمتي "بناء النظام" ، لماذا هذه التسمية ؟ وكأننا سنبنى بيت !!

هو فعلاً كبناء البيت ، لكن بناء برمجي ، بناء معنوي وقليل منه مادي ، بناء النظم سهل وسريع ، ليس كبناء الخرسانى أو البناء الملموس الذي نستطيع أن نلمسه ونتحسسه ونذكره بحواسنا ، كبناء البيوت والمنازل والناطحات السحاب ، لكن البناء البرمجي اخطر بأضعاف المرات ، لماذا البناء البرمجي اخطر من البناء الملموس ؟ المعلومات لو ضاعت لا يمكن أن تتعوض أبداً ، إذا كان البناء البرمجي سريع وسهل ، كان الهدم أسرع وأسهل بالآلاف بل بملايين المرات ...
أمر واحد فقط كفيل بأن يضيق جهد وتعب سنين ... وربما قرون ، أمر واحد فقط كافى لذلك ،خطر جداً البناء البرمجي ، سهل البناء لكن صعب المحافظة عليه .
وأنا أتكم لماذا التنظيم ، ولماذا التنظيم في كل شيء ؟ ، لكي نحافظ على المعلومات ونحاول قدر الإمكان الإسراع في البناء البرمجي ونحاول قدر الإمكان المحافظة عليه ، بوضع قيود وصلاحيات للمستخدمين ... سوف أتكم عن هذه القيود وهذه الصلاحيات أن شاء الله .

* المبرمج Programmer:

قبل أن أبدأ بالمبرمج ، ما رأيك .. ما هي البرامج ؟ ما حقيقتها ؟

هي مجرد أوامر تم برمجتها وكتابتها من قبل المبرمج ، ماهي تلك الأوامر ؟ افتح نافذة أغلق نافذة ارسم النافذة شغل برنامج آخر اعمل كذا افعل كذا لو ضغط المستخدم الزر المعين اعمل كذا وكذا الخ
انظر إليها ، أوامر كثيرة جداً تكاد تكون لا نهائية ، طبعاً المبرمج يحفظ أوامر أساسية ويبدأ يتقن في عمل أوامر تجعل الكمبيوتر يعمل بذكاء .
ليست درسنا اليوم تلك الأوامر ، والتي تختلف من لغة برمجة إلى أخرى ماهي لغات البرمجة ؟ هي أدوات يستخدمها المبرمج لعمل البرامج ، اسأل ثانية .. ماهي الأدوات التي يستخدمها المبرمج كي يعمل لنا برنامج ؟ شيء اسمه لغة البرمجة ... الكمبيوتر لا يفهم شيء من كلامنا ، ولا يفهم ما معنى الصور ولا يفهم أي شيء ...

حتى الأرقام لا يفهمها !! ... تقول له ٩ أو ٤ أو احمد لا يفهم أي شيء أبداً .

هناك شيئين يفهمهم الكمبيوتر ... شيئين فقط ... وصول إشارة كهربائية أو عدم وصول إشارة كهربائية .

طيب خردة كهذه لا تفهم إلا وصول إشارة كهربائية أو عدم وصول تلك الإشارة ... كيف صارت الآن تحسب وتجمع وتطرح ؟
العلماء استفادوا من عمل العقل ، العقل هو الآخر مجرد خردة تسمى دماغ لا يفهم إلا وصول إشارات أو عدم وصول إشارات .. لكن بطريقة الخالق سبحانه تمكن من فهم الأرقام والحروف والحساب والمنطق والإدراك والاستنتاج وووو ... الخ .

وصول إشارة تعني واحد ، عدم وصول إشارة تعني صفر ...

هل سمعت عن "رقمي" ؟ يعني الكمبيوتر جهاز رقمي يعني صفر أو واحد اعتقد الكل يعرف ذلك المصطلح .

الكمبيوتر يمثل البيانات في صفر وواحد (بوصول إشارة أو عدم وصولها) ، كيف نجعله يفهم الرقم اثنين إذا لم يفهم أصلاً إلا صفر أو واحد؟؟ من يجعله يفهم تلك الأمور ؟

هذا ليس درسنا اليوم كيف نجعله يفهم (يظهر أنني سوف اردد هذه الكلمة في كل مرة .. ليست درسنا ليست درسنا ..) ، لكن الذي يجعله يفهم ذلك هو المبرمج ، باستخدام لغة البرمجة ، ولغات البرمجة هي الوسيط بين المبرمج والبرامج ، لغات البرمجة تسهل بشكل كبير على المبرمج عمل البرامج
يعني بدل ما يقول المبرمج "يا كمبيوتر ... هذا الرقم يدعى بـ ٣ ، وهذا الرقم يدعى بـ ٦ ... و هذا الرمز هو + ، والزائد هي إضافة شيء مع شيء ، وهذا = ، أي ناتج او يساوي ، ويا كمبيوتر ٩=٦+٣ ويا كمبيوتر ٩=٣+٦ الخ "
لغات البرمجة كأنها تعلم الكمبيوتر التعرف على الأرقام والأحرف والجمع والطرح والعمليات الحسابية والمنطقية الخ ، برمج هذه اللغات مبرمجون آخرون لكي يسهلوا على المبرمجين عمل البرامج الكبيرة ولكي لا يهتموا لصغائر الأمور كتعليم الكمبيوتر عملية الجمع ، بل استخدام عملية الجمع مباشرة ، وهكذا

الآن المبرمج ليس هم الآن كيف يجمع أو يطرح الكمبيوتر قيمتين ، بل هم كيف يعمل برنامج يفكر وماذا يعمل لو صار كذا وماذا يفعل لو صار كذا ، وكيف يقرر كذا وكذا ... الخ

أي يعطي أوامر للكمبيوتر ، "يا كمبيوتر ... أغلق النافذة ، يا كمبيوتر احسب كذا وكذا ، يا كمبيوتر اعمل ملف ، يا كمبيوتر احذف جدول"

وهكذا ... مجموعة أوامر باستخدام لغات البرمجة .
لعل شرحي مبهم على الكثير الكثير من الناس ، لكن من اخذ شيء ولو البسيط عن هذه الأمور لعله يفهم ، واعذرنى لو لم تفهم ، لأنها تحتاج فعلاً لمحاضرة هذه الأمور .

المبرمج هو من يفهم و يخبر الكمبيوتر ما يريده البشر ... أي لو نريد عمل برنامج word ... من يعمل لنا ؟ من يفهم تلك الخردة ما نريده بالضبط ؟ من يترجم لغتنا إلى لغة الكمبيوتر ؟ المبرمج هو من يحول من لغتنا وأماننا واحتياجاتنا إلى لغة الصفر والواحد .

إن المبرمج هو الأساس في صناعة البرامج ، وبدونه لا يوجد برامج ، ويجب أن تتوفر فيه مهارات تعتبر نادرة فعلاً ، حتى بعض الأذكاء جداً لا يمتلكونها ، واهم تلك المهارات هي مهارة حل المشاكل ولا يسعني هنا أن أتكم عنها بتفصيل .

حتى الآن لا يوجد مشكلة لو البرنامج كبير قليلاً ... نزيد مبرمجين أو ثلاثة ولا توجد أدنى مشكلة لكن ،،

حينما تكون البرامج ضخمة جدا ، حينما يعجز العشرة مبرمجين في برمجة هذه البرامج ، حينما تتوقف مصالح الشركات على هذه البرامج ، فسوف تتضاعف مدة بناء البرامج تلك ، وسوف تبدأ لخبطة غير منطقية أبدا و إهدار مالي وزمني ضخم

كنت أبرمج على برامج يعني يكفيها تقريبا مبرمجين ، فالحاصل انه كانت تلك البرامج تظهر بشكل جدا معقد لأنه لا يوجد تنظيم فعلي للبناء البرمجي ، فبدلا من شهر ، يأخذ البرنامج ثلاث شهور ، صدقني ... المدة ستتضخم وتتضاعف أكثر بكثير كلما كبرت تلك البرامج .

بعض المشاريع يصل عدد المبرمجين إلى الآلاف ، والمبرمجين مختلفين في المهارات والتخصصات البرمجية ، فسوف يحدث خلل ليس بالسهل ، وسوف يحدث إهدار مادي ومعنوي وزمني ضخم جدا جدا ، مما قد يؤدي إلى أضرار لا حصر لها أبدا

هنا الماعمل ، المصيبة تلك ؟ نريد أن نفتح شركة لعمل البرامج للمؤسسات الحكومية والشركات الأخرى ، نريد أن نستفيد من الوقت وان لا نهدر المال ، الماعمل؟؟ حل واحد يجب تنظيم وتقسيم العمل ، وكلن له مهامه ، ولا تسألني كيف يتم التنظيم ، التنظيم ليس له شكل محدد ، لكنه خير في كل شيء ، في الوقت وفي العمل وفي جميع ميادين الحياة ،

إنن نظمنا وقسمنا ، نتيجة لهذا التنظيم والتقسيم المادي حصل ؟ مالذي حدث ؟ ظهرت وظائف أخرى غير المبرمج ، لا تستغرب ان قلت لك أن تلك الوظائف توفر رواتب تصل إلى عشرة أضعاف راتب المبرمج !! ولا تستغرب أن قلت لك أن المبرمج من الممكن ان يستغرق في العمل عشرة أضعاف أو أكثر ما يتم استغراقه في تلك الوظائف ، بالرغم من ذلك ، فالمبرمجين نواة صناعة البرامج ، بدونهم نستطيع أن نلغي الكل .

أهم تلك الوظائف أو المسميات الوظيفية محلل النظم ، ومصمم النظم .

محلل النظم The Systems Analyst:

شخص عادي ... يأكل ويشرب ، لكن هو من يقابل مستخدم وصاحب النظام ، هو من يستفسر عن متطلبات النظام ، هو من يحدد هذه المتطلبات ، وطبعا رواتب هذا الرجل ضخمة فعلا ، وطبعا لا يوجد في شركة أو فريق العمل الذي سيبني النظم إلا واحد محلل فقط ، (في الشركات الكبيرة اعتقد انه يوجد اثنين أو ثلاث ... لا اعتقد انه يوجد أكثر ،)

لا تنسى ... مالكي النظام ممكن يكونوا شركة أو مؤسسة أو حتى شخص واحد ، وهؤلاء سوف يعرضون التقارير الخاصة بهم للمحلل ، قد تجد الأمر سهل ، لكن ثق تماما ان المتطلبات صعبة الفهم واستخراج المعلومات التي نحتاجها كفريق البناء البرمجي ، والمتطلبات ممكن أن تكون مربكة ومشوشة فعلا ، وغير واضحة لو استخدمنا اللغة المقروءة ، لذا الماعمل ؟ كيف نمثل هذه المتطلبات إذا قلنا أنها ستكون غير واضحة إذا مثلناها كلغة المقروءة ؟ مالحل ؟

يقول المثل "ورب صورة تغني عن ألف كلمة "

أفضل الحلول هي اللغة التصويرية ، أي بالصور ، أي بالنماذج ، أي بتمثيل المتطلبات كصور ونماذج ...

واقصد بالنماذج هنا الشرح المصور لأي شيء ، وهي مجموعة صور تشرح أن تمثل شيء ما ، مثلا نموذج لمنزل يستخدمه المهندسين لكي يمثلون البيانات الخاصة بالمنزل ، أو يجسدون المنزل بدلا من أن يقولون "مساحة الغرفة كذا في كذا ، وهي تقع غرب الصالة ، وتقع أيضا جنوب المجلس ... الخ من شروحات لغوية" .

مشكلة النماذج ممكن تكون مربكة لو (ضع خطين تحت لو) أن النماذج غير متعارف عليها مسبقا أو مشروحة . فلو انشروحت وتعارف الكل عليها وصارت معروفة لدى الجميع فسوف تغنيانا فعلا عن ألف كلمة ، وسوف نختصر الوقت والجهد الكثير الكثير في نموذج صغير نسبيا لحجم الكلمات المستغرقة ...

لا ادري ... هل أوصلت المعلومة إليك ام تحتاج إلى أمثلة ، إذا أردت أن تشاهد الأمثلة عن النماذج فشاهدها في الدرس القادم أن شاء الله .

لما كانت هذه النماذج رائعة بهذا النحو ، فنتج ما يسمى "بمصمم النظم " ، سوف أتكلّم عنه بعد قليل

الآن المحلل يقابل ويطلع على النظام ، ويبدأ باستخراج المعلومات التي تهمنا في بناء النظام ، ماذا نسمي هذه "المعلومات التي تهمنا في بناء النظام" ؟

نسميها ... Business Rules ... لا أحب ترجمتها ، ولا ادري ماذا اسميها بالعربي ، لكن .. نكتفي باسمها الانجليزي .

إنن في نهاية الأمر مالذي سيعمله ويستخرجه المحلل ؟ هذه الـ Business Rules

مثال على الـ Business Rules "الموظف يعمل تحت قسم واحد ، القسم به أكثر من موظف ، حقول الموظفين هي كذا وكذا الخ "

وعمل المحلل حساس جدا وصعب ، طبعا لا يوجد مالك للنظام يأتي ويقول "اعمل جدول موظفين وبه الحقول كذا وكذا ، واعمل جدول أقسام الخ"

لاا ... سوف تجده يقول "في شركتنا الموظفين سوف يعملون كذا ويفعلون كذا والأوراق تذهب كذا وكذا الخ "

المستخدم لا يعرف العلاقات ولا يعرف الكائنات ، وعادة يتم مقابلة كل شخص على حدة في مكان مغلق حتى يتم تجميع المعلومات كلن وخبرته ، و عادة يتكلم المستخدم عن الحياة الواقعية الفعلية عن العمل وما إلى ذلك .

الهاجس لدى المحلل هو تحديد ماذا ؟ تحديد الجداول والحقول والعلاقات ... الخ .. يعني تحويل كلام المالك للنظام إلى متطلبات و Business Rules

ومن هذه المتطلبات والـ Business Rules .. يتم تحديد الكائنات والعلاقات وما إلى ذلك ... سوف اشرح مفهوم الكائنات بتفصيل أكثر أن شاء الله .

الآن اسأل مرة أخرى ، من أين يتم جمع المعلومات ؟ الجواب من التقارير القديمة الخاصة بالمنظمة المالكة للنظام ، ومن المقابلات الشخصية مع الموظفين ومن المذكرات الورقية الخاصة بالمنظمة .

وبقي عليك أن تعرف أن لكل منظمة ولكل مؤسسة ولكل مالك للنظام الـ Business Rules الخاصة به المشكلة هنا واضحة تماما ، كل برنامج يختلف اختلاف جذري عن الآخر ، نادرا ما نرى نظام يشبه الآخر ، حتى نظام المبيعات مثلا في شركة ما ، من الممكن أن يختلف كثيرا عن نظام المبيعات في شركة أخرى .

المتطلبات تختلف والأنظمة دائما مختلفة ، مستحيل أن نصنف أو نوجد البرامج ، دائما مختلفة ، و متزايدة في النمو مع مرور الزمن .

إنن يجب أن يطابق النظام الذي سنبنيه متطلبات النظام المقترح وان يطابق الـ Business Rules الخاصة به .

حتى أن الأنظمة العالمية العالية التقنية ، والتي تم صنعها على يد أمهر المحللين والمصممين قد لا تتفعا أبدا إذا لم تطابق تلك الأنظمة احتياجات صاحب النظام .

هل انتهى عمل المحلل ؟ لا ... المحلل يريد إعطاء العمل للمصمم ... تخيل مصمم النظم رجل كالألة ... لا يفهم إلا بالنماذج لا يفهم إلا بالصور ، هنا يضطر المحلل أن يعمل بعض النماذج المساعدة للمصمم كي يكمل المصمم المشوار ، وحقيقة يحتاج تحليل وتصميم النظم إلى منهج كامل ، من يدري لعلي اكتب مستقبلا منهجا عنها إن شاء الله

لماذا رواتب المحللين ضخمة ؟

لأنه مسؤوليته كبيرة جدا ، فأدنى الغلطات والتي ممكن أن لا ترى أو حتى ندرکہا ، تؤدي إلى خسائر كبيرة ...

البناء يبدأ بالمعمل ، ثم يرسل النتائج إلى المصمم الذي يعمل النماذج والرسومات لكي يتم تنفيذها إلى المبرمج ، الذي يبني النظام ويبرمجه ويرسله إلى المختبر الذي يختبر النظام كافة ، ولو وجد أخطاء برمجية أرجعها إلى المبرمج ... ولو وجد أخطاء في متطلبات النظام ووجد نتائج متناقضة أعيدت إلى المحلل الذي يعدل ثم يعدل المصمم ثم يعدل المبرمج ثم ترجع للمختبر الخ . ممكن أن يحدث إهدار لا يستهان به بهذه الطريقة ، لو أن المختبر لم يجد خطأ وتم تسليم المشروع إلى المالك النظام ، ووجد خطأ ؟ سوف ترجع نفس الخطوات ونفس التكرار ...

فلو تم تحليل النظام جيدا ، وبدون أخطاء أو تناقضات ، فسوف يتم تفادي هذه المشاكل .

طيب هل نتوقع أن خبرة المحلل يجب أن تكون قليلة ام كثيرة ؟ طبعا يجب أن يكون خبيرا جدا ، من ممكن أن المحلل لم يكن مبرمجا يوما ، لكن أفضل المحللين هم الذين كانوا مبرمجين ، لماذا ؟ لأنه عاش بناء النظم ، لأنه عرف مالذي يريده المبرمج بالضبط ، فتجربة شيء خير من تعلمه في اغلب الأحوال .

*** مصمم النظم The Systems Designer :**

عادة محلل النظم هو مصمم النظم ، أي محلل ومصمم النظم ، لكن بعض الأحيان تصبح منفصلتين إذا كانت المشاريع كبيرة جدا ، فكرة عمل هذا الرجل هي كفكرة المقاول والمهندس في إنشاء مشروع مثلا كمبنى تجاري أو ما شابه المهندس يصمم نماذج المشروع على الورق ، وهذه النماذج تعتبر كما قلنا شرح صوري للمشروع وبه بعض التعليقات وبعض الكتابات اللغوية تصف هذا النموذج الذي يصف المشروع . يرسل المهندس النماذج إلى المقاول الذي سوف يبدأ ببناء المبنى . المهندس في مثالنا هو مصمم النظم ، والمقاول هنا هو المبرمج ، المصمم يرسل نماذج المشروع إلى المبرمج الذي سوف يبنيه . اعتقد أن هذا واضح . إذن المصمم يتوقع من المحلل تحديد الكائنات (الجدول) والصفات (الحقول) ، والعلاقات ، وتسميات وتعريف تلك العلاقات والكائنات ، ويتوقع أيضا توثيق المعلومات.

الآن ، لماذا المحلل ولماذا المصمم ولماذا التعب والإرهاق بدلا من أن نعطي المبرمج ليبنى مباشرة ؟

لو أن لديك جدار مهديم ، هل تذهب إلى المهندس لكي يهندس لك الجدار وكم طوله وعرضه ؟ أم تذهب إلى الباني مباشرة وتخبره بالطول والعرض وانتهت المشكلة ؟ هكذا إذا كانت البرامج صغيرة ، بإمكان المبرمج أن يبني لك هذا البرنامج مباشرة بدون التحليل أو التصميم المعقد وما إلى ذلك . لكن في المشاريع الكبيرة ، ممكن استعراض كل المشروع في نماذج بدون عمله ، ومن الممكن دراسته والنظر إليه واتخاذ القرار بإكماله أو بتوقع فشله ، وحساب تكلفة المشروع ، وطبعا الخسائر حتى الآن قليلة جدا مقارنة بعمل المشروع بواسطة فريق عمل برمجي ، واختباره والتعب ، وفي النهاية التورط في تكلفة باهظة وما إلى ذلك و أيضا توفر لنا الكثير الكثير ، فسوف نرى المستخدم تلك النماذج ونقول له انظر إلى النظام انظر كيف شكله سيصبح ، غالبا ما تجده يريد تغيير شيء ، فتخيل عملنا برنامج ما ، ثم قال المستخدم .. أريد تغيير هذه الجزئية ؟ هل تتخيل قدر الوقت والجهد المهدور ؟ إذن نتأكد من هذه النماذج مطابقة متطلبات النظام للنظام المبني .

*** المختبر Tester :**

هو الذي يأتي عمله بعد المبرمج ، المبرمج بعد أن ينتهي من البرمجة يرسل البرنامج إلى المختبر الذي يتقمص شخصية المستخدم الجاهل الذي لا يعرف مالذي يحدث داخل البرنامج ، ويبدأ يضغط الأزرار بشكل عشوائي ، ويبدأ بتصيد الأخطاء ، ومن ثم يبدأ يستعرض الأوامر ويرى إذا ما كان هنالك خلل في تلك الأوامر التي كتبها المبرمج .

طبعا المختبرون لهم مهارات قد لا تجدها في غيرهم ، فهم يفكرون بشكل مقلوب ، فإذا كان الناس ينظرون إلى نصف الكوب الممتلئ ، هم ينظرون إلى النصف الآخر المسكوب ، وطبعا لهم تخصص بأكمله يدعى بـ "ضمان الجودة" ، حيث يضمن جودة البرنامج وفق لمعايير ومقاييس عالمية ! أي أن الموضوع ليس بالسهل أبدا . لا احد يستطيع إلغاء عمل المختبر ، يجب أن يحتوي كل نظام على مجموعة ، ومجموعة كبيرة من الأخطاء ، فكيف يتم تصحيحها ؟ وكيف يتم أصلا اصطياها ؟ من الممكن ان تقضي معظم وقتك حينما تعمل برنامج وأنت تريد فقط اصطياها خطأ أرهقك فعلا ، ولا تجد له حل ، وسوف تستغرق وقت أكثر بكثير من عمل البرنامج نفسه ! ... سوف أعلمك فيما بعد (حينما أبدأ بشرح برمجة قواعد البيانات) بعض التقنيات في اصطيا الأخطاء أن شاء الله .

الجدير بالذكر أن المبرمج ممكن أن يكون محلل ، ممكن أن يكون مصمم ، وحتى ممكن أن يكون مختبر للبرامج ، فالمبرمج هو الأساس فعلا ، ولو أن لديك برنامج متوسط الحجم ، فسوف تضطر إلى تحليله وتصميمه وبرمجته واختباره ...

أردت في هذا الدرس أن أبين فقط بعض المعلومات القليلة جدا عن بناء الأنظمة لأنني مبرمج ، وكمبرمج كيف ابني الأنظمة ؟ احتاج إلى التحليل ، احتاج إلى التصميم ، احتاج إلى الاختبار ، وهذه الأمور يجب تعلمها حتى تعرف أن تبني المشاريع البرمجية ، وسوف أتطرق إلى شرح بعض طرق التحليل والتصميم في الدرس القادم وطبعا ليست كافية لكي تبني مشروع ولكن نحاول أن نتقمص عمل المحلل والمصمم لكي نعرف أصلا موقعنا كمبرمجين من الخريطة ، كل عملنا هو أخذ النماذج من المصممين ونبدأ بالبرمجة ، ارجوا أن يكون هذا واضح وأسف لأنني هيجت الأمور فقط وأكثر التساؤلات في رأسك .

*** توجد نظريتين لتنظيم بناء الأنظمة :**

بدأت بعض النظريات والطرق لتحسين أداء ، وظهر تخصص يسمى بهندسة البرمجيات Software Engineering ، لتعريف هذا التخصص الجميل جدا وجدت تعريف في موقع يدعى بالموسوعة العربية للكمبيوتر والانترنت ، تعريف ممتاز :

" ما هي هندسة البرمجيات؟

لنفهم معا علاقة هندسة البرمجيات بعلوم الكمبيوتر ، دعونا نأخذ هذا المثال عن علم الكيمياء واستخدامه في حل المشاكل التي نقابلها في حياتنا اليومية.

يهتم الكيميائي بدراسة المواد الكيميائية (تركيبها، تفاعلاتها، والنظريات التي تحكم سلوكها).

بينما المهندس الكيميائي يستخدم النتائج التي توصل إليها الكيميائي لحل المشاكل التي يطلب منه إيجاد حل لها.

من وجهة نظر الكيميائي الكيمياء هي موضوع الدراسة بحد ذاتها.

ومن وجهة نظر المهندس الكيميائي الكيمياء هي أداة tool تستخدم لإيجاد الحلول لمشاكل عامة (وقد لا تكون هذه المشكلة ذات طبيعة كيميائية بحد ذاتها).

وبنفس الفكرة يمكن النظر إلى علم الحوسبة computer science حيث يكون تركيزنا على الحواسيب ولغات البرمجة لدرستها وتطويرها في حد ذاتها.

أو يمكن النظر إليها والتعامل بها على أنها أدوات نستخدمها عند تصميم وتطوير حل لمشكلة ما تواجهنا أو الآخرين.

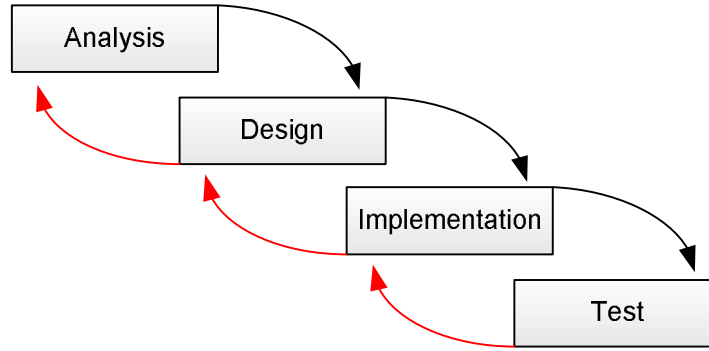
مهندس البرمجيات Software Engineer يعتبر أن الكمبيوتر هو أداة لحل المشاكل problem-solving tool.

وعليه أن يستخدم معلوماته حول الحاسوب وعلم الحوسبة للمساعدة في حل المشكلة التي يطلب منه إيجاد حل لها. "

انتهى كلامهم .

بالنظر إلى هندسة البرمجيات هي خلاصة استخدام تقنية الكمبيوتر من برمجة وتحليل وتصميم ... الخ أي الاهتمام بالمشاكل الحياتية باستخدام الكمبيوتر ، أي لا يركز كيف يطور الكمبيوتر وتقنياته ونظرياته ، بل كيف يستفيد من تلك التقنيات والتطورات لحل المشاكل . لا عليك إذا لم تفهم جيدا من هو مهندس البرمجيات هذا . نلخص أشهر طريقة لتنظيم بناء الأنظمة في هذه الصورة وتسمى نموذج الشلال ...

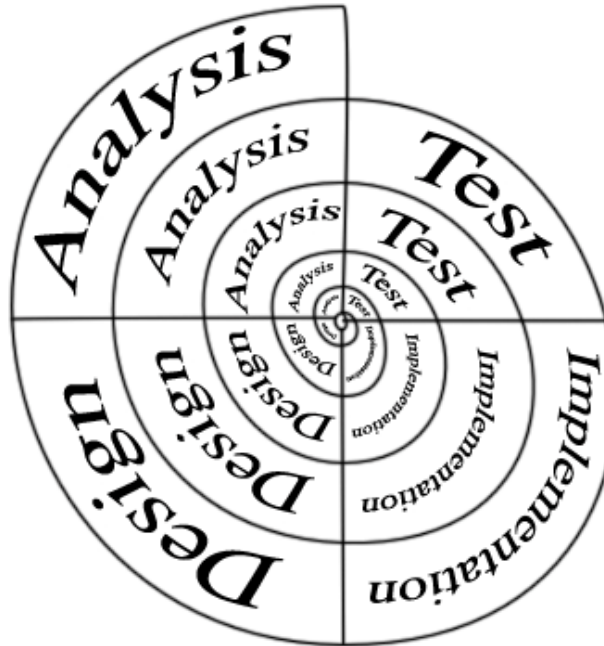
الصورة L3_P8



كما ترى ، من تحليل Analysis ، ثم تصميم Design ، ثم برمجة وبناء Implementation ، ثم أخيرا الاختبار Test ... كما ترى ، مثل الشلال ، لا يستطيع المبرمج أن يبرمج إلا حينما ينتهي المصمم ، والمصمم لا يستطيع البدء بعمله إلا حينما ينتهي المحلل . هذا ليس بعيب ولكن هذا هو النظام المتبع ، فليس من المعقول البدء بعمل برنامج لم يتم تحليله وتصميمه . اعتبر هذا التنظيم نظريا أكثر منه عمليا ، انظر إلى الأسهم الحمراء ، ما هي ؟ هي الخطوات التي سوف تتم عندما يحدث خطأ . كما قلت سابقا ، لو حدثت أدنى مشكلة في عملية التحليل ، سوف يؤثر على التصميم وعلى البرمجة وعلى الاختبار ... وسوف يتم الرجوع منذ البداية ، وهذا كما قلت إهدار ليس بالسهل . ولا يمكن أبدا يمر نظام بدون أخطاء ، حتى لو لم يكن هنالك خطأ ، ستلاحظ كم هم المستخدمون متقلبين . "لا ... أريد تغيير هذه الجزئية " ، نقول له " يارجل .. أنت قلت ذلك " ، سيقول "أريد تغييرها وحسب " .

إليك النموذج الآخر وهو نموذج كائنه حلزوني

الصورة L3_P9



فكرته هي تحليل جزء كبير من النظام ، ثم تصميم هذا الجزء وبرمجته واختباره ... في لحظة التصميم والبرمجة والاختبار يتم تحليل جزء آخر اصغر وبتفصيل أكثر ، نفس الفكرة ، يتم تصميم ذلك الجزء وبرمجته واختباره ، وهكذا .. تلاحظ في الصورة السابقة تلاحظ التحليل في البداية كجزء كبير ، ثم يصغر ويصغر حتى يتم الانتهاء من البناء كافة ، والتصميم كذلك . يتم مقابلة المستخدمين ثم يتم تحليل المتطلبات بشكل عام ثم تصميمه وبرمجته واختباره ، ثم يتم اطلاع المستخدمين إلى النتائج الجزئية ، ثم يتم التحليل ثانية مع تعديل وتصحيح الأخطاء والتأكد من أن النظام يمشي بشكل صحيح ، ثم التصميم والبرمجة وهكذا ... لا اذكر من صمم هذا النموذج لكنه مهندس برمجيات عبقري فعلا ، واعتقد انه نموذج عملي وفعلي ، وواقعي أكثر .

في الدرس القادم سنتقصد شخصيتي المحللين والمصممين ونرى المشاكل التي سوف يواجهها هذين المسكينين و مالحول المتبعة عادة ، لكي نستطيع إكمال مشوارنا كمبرمجين .

انتهى الدرس ،،،