قواعد البيانات (Data Base)

DBMS

Data Base Management Systems

نظام اداره قواعد البيانات

المقدمة

زادت أهمية المعلومات في وقتنا الحاضر من أهمية نُظم قواعد البيانات لتنظيم المعلومات التي نحتاجُها وتوفيرُها للأستخدام بالطريقة المُناسبة وفي الوقت المُناسب . هكذا تطور علم أدارة نُظم قواعد البيانات من موضوع فرعي في تطبيقات الحاسوب الى موضوع أساسي ومكون رئيسي من علم الحاسوب الحديث ، وبذلك فقد أصبحت نُظم قواعد البيانات جُزءاً رئيسياً في منهاج علم الحاسوب.

ماهي قاعدة البيانات ؟

هي مجموعة منظمة من البيانات المرتبطة ببعضها البعض منطقيا.، وهي من أهم الدعائم التي تقوم عليها المعلومات حيث من خلال قواعد البيانات نستطيع حفظ وتعديل وحذف المعلومات بطرق سلسة ، وكذلك تتيح لنا استخراج البيانات المحفوظة كما نريد.

مثال على ذلك دليل الهاتف الذي يشتمل على أسماء وعناوين وأرقام هواتف سكان مدينة بغداد يمكن أن نعتبره قاعدة بيانات وبتحقيق الأستفاده من قاعدة البيانات هذه بإدخال رقم المشترك والحصول على أسمه وعنوانه أو إدخال أسم المشترك والحصول على رقم هاتفه وعنوانه وهكذاً.

- **أهمية قواعد البيانات:-**

أ- تخزين جميع البيانات بكافة الأنشطة لجهة ما بطرق متكاملة ودقيقه وتصنيف وتنظيم هذه البيانات بحيث يسهل استرجاعها في المستقبل.

ب- متابعة التغيرات التي تحدث في البيانات المخزنة وإدخال التعديلات اللازمة عليها، حتى تكون دائماً في الصورة الملائمة لاستخدامها فور طلبها.

ج- تخزين كم هائل من البيانات التي تتجاوز الإمكانيات البشرية في تذكر تفاصيلها ومن ثم إجراء بعض العمليات والمعالجات التي يستحيل تنفيذها يدوياً.

د- تساعد على تخزين البيانات بطريقه متكاملة، بمعنى الربط بين النوعيات المختلفة للبيانات المعبرة عن كافة الأنشطة.

هـ- تساعد على تحقيق السرية الكاملة للبيانات المخزنة بها بحيث لا تتاح أية معلومات لأي شخص ليس له الحق في الإطلاع عليها.

- **- وظائف قواعد البيانات:-**

أ- إضافة معلومة أو بيان جديد إلى الملف.

ب- حذف البيانات القديمة والتي لم تعد هناك حاجة إليها.

ج- تغيير بيانات موجودة تبعاً لمعلومات استحدثت .

د- البحث والاستعلام عن معلومة أو معلومات محددة .

هـ- ترتيب وتنظيم البيانات داخل الملفات .

و- عرض البيانات في شكل تقارير أو نماذج منظمه .

ز- حساب المجموع النهائي أو المجموع الفرعي أو المتوسط الحسابي لبيانات مطلوبة .

- **تنظيم البيانات داخل قاعدة البيانات:-**

تخزن المعلومات المطلوبة لقواعد البيانات داخل (ملفات)، وتوضع هذه الملفات على أحد وسائط التخزين المساعدة مثل القرص المغناطيسي.

كل ملف عبارة عن جدول يشتمل على سطور وأعمده ، ويشتمل كل ملف على مجموعه من السجلات Records ويحتل كل سجل سطراً داخل الملف ، ويقسم كل سجل إلى عدد من الحقول Fields .

إذا أردنا إعداد دليل تلفونات لسكان مدينة بغداد ، والملف المطلوب يشتمل على البيانات التالية:-

رقم المشترك الاسم العنوان رقم الهاتف

رقم المشترك	الاسم	العنوان	رقم الهاتف
01	علي محمد علي	بغداد	0778989898
02	خالد محمد	بغداد	0771123456

يوضح كيفية تنظيم البيانات داخل جدول قاعدة البيانات.

- **بعض مصطلحات قواعد البيانات**

البيانات DATA:-هي الأرقام أو الحروف أو الرموز أو الكلمات القابلة للمعالجة بواسطة الحاسب مثل: الرقم(65) أو كلمة بيانات.

- المعلومات Information:-هي بيانات تم تنظيمها أو معالجتها لتحقيق أقصى استفادة منها. مثال: الرقم (6) والرقم (5) إذا استخدما في عملية الضرب 5×6 أصبحا معلومة مفيدة.

- الجدول: Table - يحتوي الجدول على المعلومات التي نود حفظها في قاعدة البيانات في ترتيب معين على شكل صفوف وأعمدة. وهذه الصورة تمثل جدولاً ما.

	Number	Name	Age
	0	Ahmed	18
	0	Hosame	17
	0	Khaled	19
	0	Mohammed	21
	0	Ali	30
	0	Sami	80
*	0		0

- الإستعلام: Quires -

يقوم الإستعلام بمهمة تعيين سجلات معينة أو حقول معينة والتي يمكن أن تحقق شرطاً معينا. وهذه صورة لاستعلام يتضمن اسماء الأشخاص الذين تزيد أعمارهم عن 18 سنة.

	Name
▶	Khaled
	Mohammed
	Ali
	Sami
*	

- السجل: Record -

يشكل كل سجل شخصاً أو سلعة و مجموعة بيانات لوصف عنصر مستقل ، ما وتوجد لها قيم في كل حقل.

- الحقل Field -

يشكل الحقل نوعاً معيناً من البيانات سيتم اضافته لكل سجل.

. Database Management Systems (DBMS)

هو برنامج خاص بإدارة قواعد البيانات، ويسمح لك بإضافة وتعديل وحذف البيانات عبر واجهات ونوافذ ، وأيضا من خلال ربط قاعدة البيانات بإحدى لغات البرمجة ، ويتيح لك أيضا مشاركة قواعد البيانات داخل شبكة Network ، وتوجد الآن العديد من أنظمة إدارة قواعد البيانات ولعل أشهرها DB2, Microsoft SQL MySQL, PostqreSQL, Sybase, IBM, Microsoft ACCESS, Server, Oracle .

- **اداره نظام قواعد البيانات تسعى لتحقيق ثلاثة أهداف هامة:**

1. دمج البيانات Data Consolidation

هذا الهدف يشير إلى إمكانية ضم أو توحيد ملفات البيانات المنفصلة في بنية مركزية ، وتخزين البيانات بصيغة خالية من الفائض ؛ الذي ينشأ في قاعدة البيانات عندما يخزن في موقعين أو أكثر ، فمثلاً قد نجد التخصص العلمي للموظف مخزناً ليس فقط في جدول البيانات الشخصية وإنما نجده أيضاً في جدول الوظائف ، وجدول التاريخ الوظيفي ، وعندها نكون أمام قاعدة بيانات غير مركزية تحوي معلومات زائدة ، ولذلك ينبغي إذا أردنا أن نبني نظاماً مثالياً ومتكاملاً وخالياً من الفائض أن يحتوي على تخصص الموظف في جدول واحد.

2. المشاركة على البيانات Data Sharing

أي قدرة النظام على السماح لعدة مستخدمين بالوصول إلى أجزاء مستقلة من البيانات ضمن قاعدة البيانات في نفس الوقت ، وهذه خاصية تتميز بها تطبيقات DBMS بما يعرف بالتوازي.

3. حماية البيانات Data Protection

أي قدرة الـ dbms على المحافظة على سلامة البيانات أمام الحوادث الطارئة خلال المعالجة (فشل البرنامج أو توقفه فجأة ..الخ) ، إذ ينبغي على الـ DBMS أن تمتلك القدرة على إعادة البيانات إلى

حالتها السابقة قبل التعديل غير الكامل عليها أو قبل حدوث الخطأ فيها وتسمى هذه العملية أحياناً بالتراجع(undo) .

هناك ثلاثة أنواع شائعة من نظم إدارة قواعد البيانات وهي.

1. نظم إدارة قواعد البيانات الهرميةHierarchical DBMS
2. نظم إدارة قواعد البيانات الشبكية Network DBMS
3. **نظم إدارة قواعد البيانات العلائقية Relational DBMS**

- **Relational Model**

الانموذج العلائقي : ظهر هذا النظام على يد Edgard Franck Codd سنه 1970 وينبني هذا النموذج على مفهوم(الجبر العلائقي) مفهوم رياضي الذي يتيح الاستعلام عن البيانات الموجودة في النظام و يضم وحدات مترابطة فيما بينها.

فى حال وجود علاقة بين جداول قاعدة البيانات يسمى هذا

Relational Database Management System - RDBMS

- مبادئ النموذج العلاقي Relational Data Model Concepts

1. يقوم النموذج العلاقي بتمثيل قواعد البيانات كمجموعة من العلاقات (الجداول)
2. النموذج العلاقي يتم استخدامه على نطاق كبير بسبب سهولته وبسبب وجود أساس رياضي له
3. كل جدول في النموذج يحتوي على مجموعة من الصفوف التي تمثل مجموعة من البيانات المترابطة.

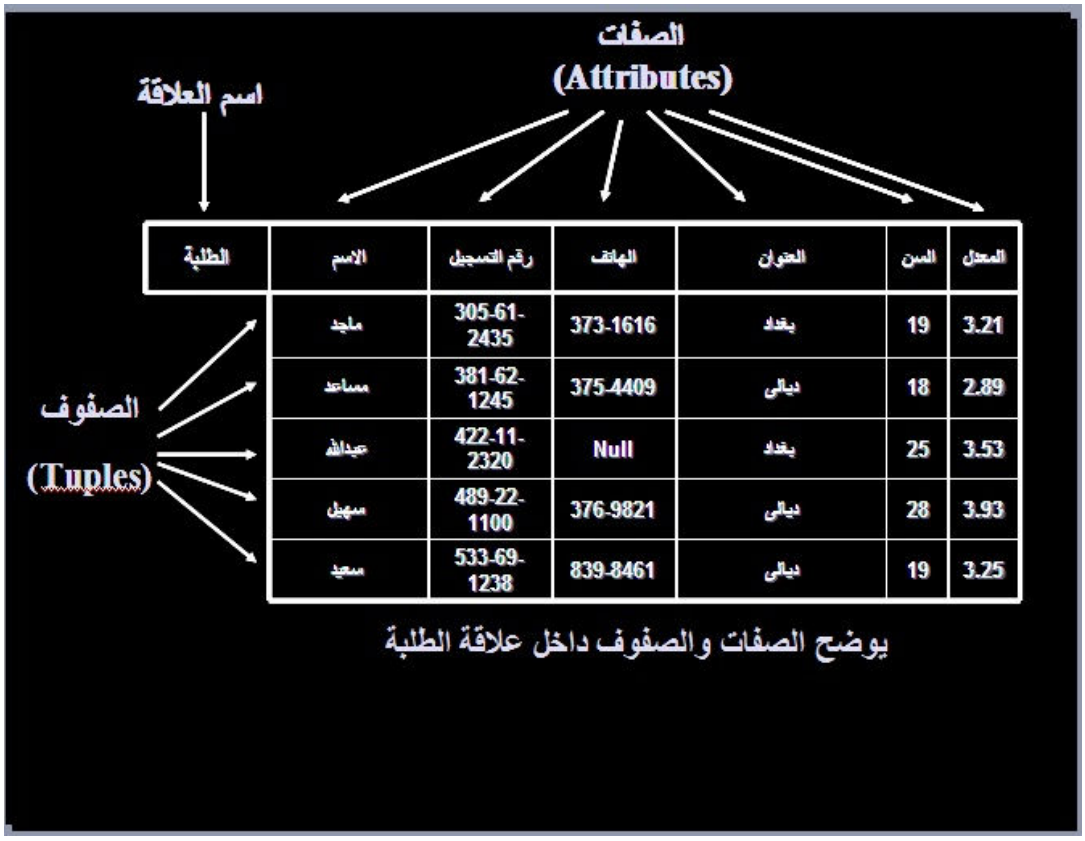
- ظهور الانموذج العلائقي لحل جمله من المشاكل وهي :

1. يمكن فهم قاعدة البيانات لمن لم يدرسوا علوم الحاسب.
2. يمكن تعديل وإضافة وحذف بيانات دون تغيير المخطط المنطقي للقاعدة.
3. تتيح للمستخدم اعلي درجة من المرونة في التعامل مع البيانات.

في عام 1970 أستحدث Codd أسلوبا لتنظيم وفرز بيانات قواعد البيانات . وهي قواعد البيانات العلائقية. وقد وجد العالم الأمريكي Codd أن هذا لا يتحقق ألا برص البيانات على هيئة جداول لان الإنسان تعود على الجداول منذ طفولته بداية من جدول الحصص إلى جدول الضرب إلى كشف الأسماء و الدرجات وهذه النظم تتعامل مع اكثر من ملف في نفس الوقت وتعامل البيانات داخل الملف كما لو كانت جدولا مكونا من صفوف و أعمدة ويسمى علاقة Relation وتمثل أعمدة الجدول حقول قاعدة البيانات Fields وتسمى أيضا Attributes بينما تمثل صفوفها سجلات قاعدة البيانات وتسمى Tuples و النظام العلائقي Relation يقوم بربط البيانات بين العلاقات بناء على حقل مشترك بينهما. والنظم العلاقية قامت أساسا علي النظريات العلاقية في الرياضيات وقد بدأ تطبيقها على الحاسبات الكبيرة أولا مثل ORACLE . SQL ثم ظهرت عدة نظم علائقية على

الحاسبات الشخصية PCs مثل برامج . DBaseIV . +DBaseIII . DBaseII
FoxBase . FoxPro ..

رسم توضيحي للمصطلحات في الانموذج العلائقي



في قواعد البيانات العلائقية فالتكوين العلائقي تكوين منطقي بحيث يستخدم علاقات ضمنية Implicit Relationships بدلا من استخدامه لعلاقات صريحة Explicit Relationships وهي التي تستخدم في كل من قواعد البيانات الهرمية و الشبكية .وحتى نوضح مفهوم العلاقات الضمنية بين ملفات قاعدة البيانات العلائقية وكيفية استخدامها في تجميع البيانات مع بعضها من ملفات منفصلة عن بعضها نفرض أن لدينا جدولين في قاعدة البيانات جدول [أ] وجدول [ب] . جدول [أ [يعرف الاقسام لكل كلية باستخدام رقم القسم كحقل مفتاحي و الجدول [ب] يحدد الطلبه ويجد لكل طالب رقم قسم كمفتاح ثانوي و الجدولان منفصلان عن بعضهما أي لا يوجد أي اتصال طبيعي بينهما وتحدد العلاقة ضمنيا وذلك بإدخال حقل رقم القسم في كل من الجدولين.

- **الجبر العلائقي Relational Algebra**

عبارة عن مفهوم رياضي محض، أعتقد لو كنت من هواة الرياضيات سيكون قد مر على مسامعك من دون شك، وهو يقوم على أطروحة المجموعات Group Theory والغاية منه هو الحصول على بيانات جديدة من خلال بعض العمليات التي نقوم بها على وحدات (جداول) أخرى، وهذه العمليات هي:

- **العمليات التجميعية**:

تطبق هذه العمليات على مجموعتين.

الاتحاد (U) Union: الاتحاد هو علاقة تربط بين مجموعتين لهما نفس الحقول ونفس الخصائص، وتكون النتيجة عبارة عن مجموعة تضم كل عناصر المجموعتين، ويرمز لها رياضيا هكذا: **R1 U R2.**

حتى نستوعب المفهوم أكثر سنورد الجدولين التاليين بنفس البنية:

R1 : الفوج الأول من العمال

Code	Name	Age	Address
P1E1	Ahmed	26	Kuwait
P1E2	Idriss	24	Morocco
P1E3	Kamal	25	Egypt

R2 : الفوج الثاني من العمال

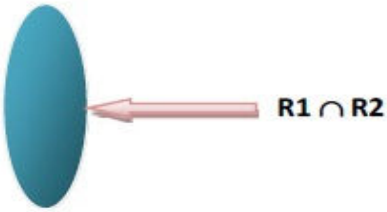
Code	Name	Age	Address
P2E1	Khalid	24	Tunisia
P2E2	Ismaîl	27	Syria

R1∪R2: اتحاد الوحدتين

Code	Name	Age	Address
P1E1	Ahmed	26	Kuwait
P1E2	Idriss	24	Morocco
P1E3	Kamal	25	Egypt
P2E1	Khalid	24	Tunisia
P2E2	Ismaîl	27	Syria

قائمة أفواج العمال

التقاطع (∩) Intersection :



التقاطع هو ناتج ربط جدولين لهما نفس عدد الحقول، ونفس البنية، ويرمز له رياضيا **(R1 ∩ R2)** . وهو يضم العناصر المشتركة بين جدولين. حتى نستوعب المفهوم أكثر سنورد الجدولين التاليين بنفس البنية :

R1 : مكتبه كتب كليه التربية الاساسية .

Code	Book
B1	Kalila wa dimna
B2	Moqadimat Ibn Khaldoun
B3	Truth of life
B4	C# Programming
B5	Java Programming

R2 : مكتبه كتب كلية العلوم .

Code	Book
B4	C# Programming
B13	Health & Body
B3	Truth of life
B15	PHP for beginners
B5	Java Programming

كما نلاحظ لدينا مجموعتان لهما نفس البنية، ولديهما بعض العناصر المشتركة بينهما كما يعرض الجدول التالي:

(R1∩R2) العناصر المشتركة بين المجموعتين:

Code	Book
B4	C# Programming
B3	Truth of life
B5	Java Programming

الاختلاف Difference (-):

الاختلاف هو الفارق الناتج عن طرح مجموعة من مجموعة أخرى، ويشترط أن يكون (للمجموعتين) الجدولين نفس البنية ونفس الحقول.

لنأخذ نفس المثال الأول ونطبق عليه عملية الاختلاف.

R1 : الفوج الأول من العمال

Code	Name	Age	Address
P1E1	Ahmed	26	Kuwait
P1E2	Idriss	24	Morocco
P2E1	Khalid	24	Tunisia
P2E2	Ismaîl	27	Syria
P1E3	Youssef	32	Algeria

R2 : الفوج الثاني من العمال

Code	Name	Age	Address
P2E1	Khalid	24	Tunisia
P2E2	Ismaîl	27	Syria
P2E3	Karim	29	Qatar
P2E4	Mahmoud	31	Arabic Saudia
P2E5	Ibrahim	36	Libya

R1-R2: الفوج الأول ناقص الفوج الثاني:

Code	Name	Age	Address
P1E1	Ahmed	26	Kuwait
P1E2	Idriss	24	Morocco
P1E3	Youssef	32	Algeria

R1-R2: تعني جلب العناصر الموجودة في R1 وغير الموجودة في R2.

بالمقابل يمكننا القيام بعملية الاختلاف بشكل عكسي، على شكل R2-R1 في هذه الحالة ستكون النتيجة عبارة عن مجموعة تضم فقط العناصر الموجودة في R2 والتي لا توجد في R1 أي كما يعرض الجدول التالي.

R2-R1: الفوج الثاني ناقص الفوج الأول:

Code	Name	Age	Address
P2E3	Karim	29	Qatar
P2E4	Mahmoud	31	Arabic Saudia
P2E5	Ibrahim	36	Libya

- **العمليات الأحادية :**

تطبق هذه العمليات على مجموعة واحدة، وتنقسم إلى:

1- الانتقاء Selection :

وتعني انتقاء بعض العناصر/الأسطر Rows من مجموعة معينة، مثلا لو عندنا جدول العمال التالي:

ID	Name	Function	City
1	Younes MAADANE	Developper	CasaBlanca
2	Ismaîl WAHBI	Conceptor	CasaBlanca
3	Reda Hamdi	Designer	Rabat
4	Hamid MAKBOUL	Director	CasaBlanca
5	Mohammed ELKHAL	Web MASTER	Agadir

الانتقاء في هذه الحالة يعني الاستعلام عن بعض العمال الموجودين ضمن المجموعه مثلا:

انتقاء العمال الذين عندهم صفة مبرمج Developper:

ID	Name	Function	City
1	Younes MAADANE	Developper	CasaBlanca

2- الاسقاط Projection :

الفرق بينه وبين الانتقاء هو كون الاسقاط يكون بغرض انتقاء الأعمدة Columns وليس الأسطر، فمثلا لو عندنا نفس الجدو ل السابق:

جلب أرقام وأسماء العمال فقط:

ID	Name
1	Younes MAADANE
2	Ismaîl WAHBI
3	Reda Hamdi
4	Hamid MAKBOUL
5	Mohammed ELKHAL

أو جلب مهن العمال فقط:

Function
Developper
Conceptor
Designer
Director
Web MASTER

العمليات الثنائية العلائقية :

1- الجداء الديكارتي (x) :

ويكون الناتج عن هذه العملية عبارة عن مجموعة جديدة، تضم خارج جداء كل عنصر من المجموعتين بباقي عناصر المجموعه الأخرى، فمثلا لو عندنا الجدولين :

R1

1
2
3

R2

10
11

R1 X R2

1	10
2	10
3	10
1	11
2	11
3	11

2- القسمة Division (÷) :

ويعني قسمة جدول على جدول آخر، بشرط أن تكون حقول الجدول الثاني متواجدة في الجدول الأول، وتكون النتيجة عبارة عن جدول يضم عناصر الجدول الأول التي تضم كل عناصر الجدول الثاني وصيغتها الرياضية هكذا R3=R1÷R2

نفترض مثلا أن عندي جدولان، الأول يضم قائمة للممثلين والأفلام التي شاركوا فيها، والثاني يضم قائمة للأفلام السينمائية:

R1

Actor_Name	Film
Franck Richard	Sun & Moon
José Melany	Like a boss
Franck Richard	Like a boss
Michel Ravaud	Sun & Moon
Katherine elise	Like a boss
José Melany	Sun & Moon

R2

Film
Sun & Moon
Like a boss

كيف نستطيع جلب أسماء الممثلين الذين شاركوا في كل الأفلام؟

للجواب على هذا السؤال سنقوم بقسمة الجدول الأول R1 على الجدول الثاني R2 ونتيجه القسمة ستضم فقط الممثلين الذين شاركوا في كل الأفلام، أي هكذا:

R1÷R2

Actor_Name
Franck Richard
José Melany

3- الربط Join:

وهو من أبرز المفاهيم التي سنراها إن شاء الله مع لغة SQL ويقتضي هذا النوع من العمليات جدولين لهما حقل مشترك من نفس النوع، ويستعمل الربط بغرض البحث عن العناصر الموجودة في الجدولين من خلال تحقق شرط وجود الحقل المشترك بنفس القيمة
في الجدولين،

لنتأمل المثال الآتي:

R1

ID_Country	Country
1	Egypt
2	Morocco
3	Algeria

R2

ID_Citizen	Citizen	ID_Country
C1	Hamdi	1
C2	Khalid	3
C3	Saîd	2

يمكننا من خلال عملية الربط أن نجلب المواطنين والدول مادام رقم الدولة في المجموعه الأولى يتوافق مع رقم الدولة في المجموعه الثانية .
نقوم بجلب اسم المواطن من جدول المواطنين، ونجلب اسم الوطن من جدول الأوطان،
بشرط أن تتوافق قيمة الحقل المشترك بين المجموعتين.

خلاصة الجبر العلائقي:

مما سبق نستنتج بأن الجبر العلائقي هو وسيلة لمخاطبة الجداول / المجموعات، بغية جلب بعض البيانات بطرق متعددة، ويمكننا اختصار ما مضى في الرسم التالي :

- **Entity Relationship Model العـــلاقـــات .**

هو احد الاساليب الشائعه لوضع تصور للقاعده البيانات العلائقية وهو يعتمد على الاتى:

1. تقسيم قاعده البيانات الى جداول مثل جدول الموظف او القسم او جدول طلاب وجدول المرحلة والدرجات وغيرة.
2. كل جدول يحتوى على خصائص معينة تصف هذا الجدول مثل (الاسم والعنوان والسن والوظيفة).

وتحديد العلاقة بين هذة الجداول والخصائص نستطيع ان نعبر عنة بما يسمى **Entity Relationship Model**

- **أهمية Entity Relationship Model ؟**

الاهمية الاكبر انه يساعد على عمل قاعده بيانات على اساس علمي صحيح وبالتالى يعتبر خطوة مهمه جدا قبل البدء فى انشاء قاعده البيانات .

- **المفتاح الرئيسي والمفتاح الثانوي او الاجنبي :**

1. المفتاح الرئيسي primary key: (PK) هو حقل في جدول يتميز بأن قيمه وحيدة في جميع صفوف الجدول ، وتكون قيمه مميزة لكل صف عن أي صف آخر.
2. المفتاح الاجنبي او الثانوي Forgin Key : (FK) هو حقل موجود في جدول وهو لا يمثل واحدة من صفاته، ولكنه يعتبر مفتاح أجنبيا لأنه يمثل جدولا آخر، ويجب أن يكون هو نفسه المفتاح الرئيسي في ذلك الجدول، أو على الأقل تكون قيمه وحيده.

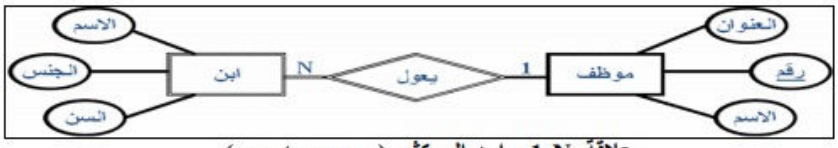
- **انوع العـــلاقات :**

يمكن ربط جدولين إذا كان كليهما يشتمل على حقل أو أكثر بهما نفس البيانات، وعادةً تسمى الحقول في كلا الجدولين بنفس الاسم.

وهناك ثلاث أنواع من العلاقات:-

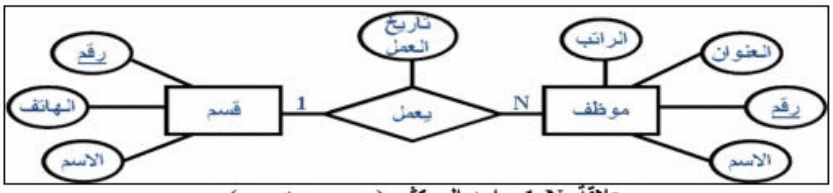
1. علاقة واحد مقابل مجموعة One - to - many:-

هي الأكثر استخداماً، وتعنى أن السجل الواحد في جدول البيانات (الجدول الرئيسي Primary Table) يقابله أكثر من سجل في جدول آخر (الجدول المرتبط Related Table) ويرمز لها بالرمز N:1
او بالعكس .



علاقة 1:N واحد-الى-كثير (one-to-many)

لا حظ أنه كل ابن يتبع لموظف واحد، لأنه لكل إبن أب واحد!، ولكن الموظف قد يكون له عدة أبناء.



علاقة 1:N واحد-الى-كثير (one-to-many)

لاحظ أنه لكل موظف قسم واحد، فالموظف لا يمكن أن ينتمي لأكثر من قسم إداري واحد، ولكن القسم قد يكون فيه عدة موظفين.

لو فرضنا لدينا جدولين

الاول جدول الاقسام ويتكون من

رقم القسم	اسم القسم
10	الحاسبات
20	الرياضيات
30	التاريخ

*حقل رقم القسم : يكون مفتاح رئيسي Pk

حقل اسم القسم : يمثل اسم القسم

وجدول ثاني جدول الطلبة ويتكون من

رقم الطالب	رقم القسم	اسم الطالب
1	10	محمود
2	20	محمد
3	10	احمد
4	30	حميد

حقل رقم الطالب : مفتاح رئيسي لجدول الطلبة.

حقل رقم القسم : المفتاح الثانوي لجدول الطلبة لربط جدول الطلبه مع جدول الاقسام عن طريق المفتاح الرئيسي والمفتاح الثانوي

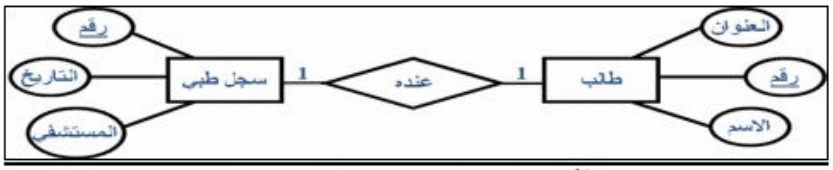
حقل اسم الطالب : يمثل اسم الطالب

هنا العلاقه one to mony اي لكل قسم مجموعه طلبه .

او mony to one مجموعه طلبه لهم قسم واحد فقط .

2. علاقة واحد مقابل واحد One - to - One:-

هذا النوع أقل استخداماً من النوع السابق، وفيه كل سجل في الجدول الرئيسي يقابله سجل واحد في الجدول المرتبط به (فمثلاً : عندما ترغب في فصل معلومات العميل إلى بيانات عامه وبيانات خاصة) ويرمز لها بالرمز 1:1 .



علاقة ١:١ واحد-الى-واحد (one-to-one)

لاحظ أنه لكل طالب سجل طبي واحد (نوع العلاقة ١)، والسجل يكون لطالب واحد (نوع العلاقة ١).

لو فرضنا لدينا جدولين

جدول الاقسام ويتكون من رقم القسم واسم القسم

رقم القسم	اسم القسم
10	الحاسبات
20	الرياضيات
30	التاريخ

حقل رقم القسم : يكون مفتاح رئيسي Pk

الجدول الثاني ويتكون من معلومات اضافية حول القسم

تسلسل	رقم القسم	عدد الاساتذه	عدد الطلبة
1	10	20	200
2	20	30	900
3	30	25	327

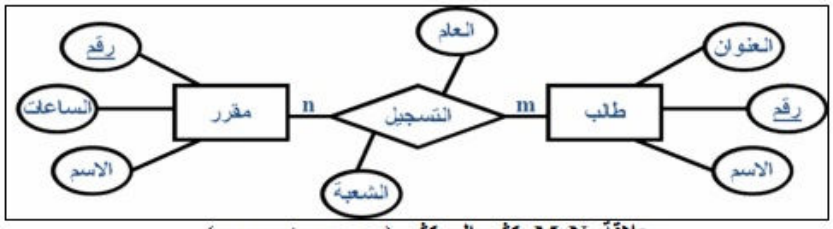
حقل التسلسل: يمثل المفتاح الرئيسي Pk

حقل رقم القسم: يمثل المفتاح الثانوي Fk

العلاقة one to one لكل قسم بالجدول الاول رقم مقابل لها بالجدول الثاني وبهذا تكون العلاقه one to one.

3. علاقة مجموعة مقابل مجموعه Many - to - many:-

هذا النوع نادر الاستخدام وفيه يقابل كل سجل من الجدول الرئيسي عدة سجلات في الجدول المرتبط، ويقابل السجل الواحد في الجدول المرتبط عدة سجلات في الجدول الرئيسي (المنتجات وأوامر الشراء)، وهذا النوع من العلاقات معقد، لذا يجب ربط الجدولين بأسلوب غير مباشر وذلك بإنشاء جدول ثالث يعمل على تجزئة علاقة (ارتباط مجموعه مقابل مجموعه) إلى علاقتين من نوع (واحد مقابل مجموعه)، وفى هذه الحالة تضع المفتاحين الأساسيين لكلا الجدولين في الجدول الثالث. ويرمز لها بالرمز N:M .



علاقة M:N كثير-الى-كثير (many-to-many)

لاحظ أن الطالب قد يكون له عدة مقررات، وكذلك المقرر يمكن أن يسجله عدة طلبة.

ربط الجداول ذات العلاقة المشتركة:-

قبل إنشاء علاقة بين جدولين تأكد أن او هناك شروط يجب اتباعها :

1- كلا الجدولين لهما حقل / حقول متشابهة.

2- تحديد الجدول الرئيسي والجدول التابع.

3- إذ لم يكن الجدول التابع يشتمل على حقل يتطابق مع حقل المفتاح الأساسي في الجدول الرئيسي، أضف حقلاً جديداً في الجدول التابع بنفس مواصفات حقل المفتاح الأساسي في الجدول الرئيسي.

4- لا يمكن تعديل مواصفات أو حذف الحقول التي أسست علاقة الربط إلا إذا ألغيت علاقة الربط بين الجدولين أولاً.

SQL :(DDL And DML)

SQL : هي اختصار ل Structured Query Language وتعني لغة الاستعلامات المرتبة، وتستعمل من أجل إجراء عمليات على قواعد البيانات.

حتى نستوعب هذا المعنى بصفة دقيقة، فلغة SQL هي التعبير البرمجي للجبر العلائقي ظهرت هذه اللغة سنة 1974 ثم بعد ذلك في سنة 1986 تم اعتمادها من طرف (ANSI) وفي سنة 1987 تم اعتمادها من قبل ISO لتصبح بذلك اللغة الأكثر شيوعا في أنظمة إدارة قواعد البيانات العلائقية .

لغة لتعريف البيانات DDL: وهي اختصار ل **D**ata **D**efinition **L**anguage أي انها تتيح لنا إنشاء وتعديل وحذف الكائنات (قواعد بيانات DataBase - جداول Table - مشاهد View - الفهارس Indexes - اجراءات مخزنه Procedures - القوادح Triggers).

لغة لمعالجة البيانات DML: وهي اختصار ل **D**ata **M**anipulation **L**anguage أي انها تمكننا من انتقاء Select وإضافة Add وتحديث Update وحذف Delete البيانات من قاعدة بيانات علائقية.

- **إنشاء قواعد البيانات:**

لإنشاء قاعدة بيانات فالصيغة كما يلي:

```
CREATE DATABASE MyDatabase ;
```

MyDatabase هو اسم قاعدة البيانات التي نريد إنشاءها.

لحذف قاعدة البيانات فالصيغة كما يلي :

```
DROP DATABASE MyDatabase ;
```

MyDatabase هو اسم قاعدة البيانات التي نريد حذفها .

- **الجداول Tables :**

الجداول عبارة عن وحدات لتخزين البيانات على شكل مصفوفة ثنائية الأبعاد تتكون من من صفوف واعمده .

انشاء جدول :

عند إنشاء جدول، يلزم عموما احترام مايلي:

- أن لا يكون اسم الجدول كلمة محجوزة في لغة SQL .

لإنشاء جدول فالصيغة كما يلي

```
CREATE TABLE MyTable (
ID INT,
FullName VARCHAR(50),
BirthDate DATETIME
)
```

الأمر أعلاه يقوم بإنشاء جدول اسمه MyTable ويتكون من حقول ثلاثة، الأول نوعه رقمي، الثاني نوعه نصي يتسع ل 50 حرف والأخير من نوع التاريخ DataTime.

حذف الجدول :

لحذف جدول نقوم بكتابة الأمر التالي:

```
DROP TABLE MyTable;
```

MyTable هو اسم الجدول المراد حذفه.

تعديل الجداول:

لإضافة بعض الحقول إلى جدول ما فالصيغة دائما هكذا:

```
ALTER TABLE MyTable ADD Age int;
```