


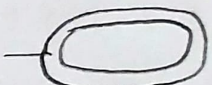

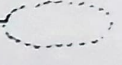

Data Base (ERD)

Entity → طالب → كائن
↓
instance → طالب معرف

ERD steps

- 1- ما هي الكائنات (Entity)
- 2- ما هي صفات الكائنات (attributes)
- 3- ما هو ال ID للكائنات
- 4- ما هي العلاقة بين الكائنات

types of attributes

- 1- key → instance الميزة للـ ID (student number) 
- 2- Multi-value → instance الواحد له أكثر من قيمة 
- 3- composite → يمكن أقسامه لأكثر من جزء (Name - Address) 
- 4- Derived → يمكن استنتاجه من صفات أخرى (Age) 
- 5- Normal → لا ينتمي لأي نوع آخر 

key types

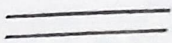

- 1- single key → صفات غير صف واحد تنفع key
- 2- candidate key → أكثر من صف تنفع key
- 3- composite key → كل واحد ممكن يتكرر لو اصة بين الاثنين مع بعض صيغته

ملاحظة:


* attribute معرف على اثنين entity يبقى مكانه طالع من ال Relation التي بينهم


Relationship


1- participation constraints :

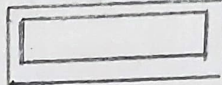
Double line  must - mandatory - total
single line  may - partial - optional

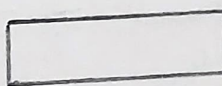
2- Cardinality :

one - one 

one - many 

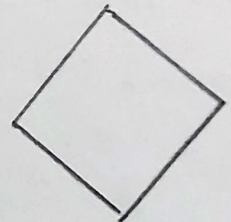
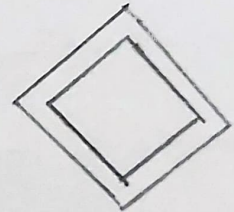
many - many 

Weak entity \rightarrow key \rightarrow 

regular entity \rightarrow key \rightarrow 

relations \rightarrow one weak one regular \rightarrow

\rightarrow two regular \rightarrow



Mapping

(نحوه ERD الى جاده)

1 - Mapping Regular Entity → لكل entity بهمة جدول والاعادة هي
ال attributes ويحدد ال Primary key

- primary key → (unique - not Null - table contains 1 pk only)

2 - Mapping weak Entity → entity بهمة جدول والاعادة هي ال attributes
منضبط ال pk يناعت ال entity المرتبط به ^{weak} Foreign key
منعقد pk من النوع ال composite واحد منه Foreign key

- Foreign key → (Not unique - may be Null - table contain 1 or more)

* لو فيه علاقة 1:1 و must من الناحيتين → منطوق جدول واحد ومنافذ ال pk يناعت ال
من متواجدة في الحقيقة

* لو فيه علاقة 1:1 و may من الناحيتين → منضبط جدول ثالث فيه ال pk يناعت ال اثنين

* لو فيه علاقة 1:1 و may من T₂ و must T₁ → ال T₂ زي ما هي وال T₁ منضبط ال pk
composite key
T₂ (C₁, C₂) pk
Fk منضبط ال T₁
لو فيه attribute معتمد على اثنين معتمد في جدول من ناحية ال must
بقية ال pk يناعت ال T₂ معتمد من ال must

4 - 1: N

باض ال pk يناعت ال ابطلها فk في ال N

* لو في جدول معتمد بنفسه بضيف ال pk يناعت ال فk ثاني

* لو فيه attribute معتمد على اثنين معتمد في جدول ال many

5 - N: N

منضبط جدول ثالث فيه ال pk يناعت ال اثنين ال composite وكن key واحدة هيقع

لو فيه attribute معتمد على الاثنين بخطه كعمود محاري في الجدول الثالث
Fk

6 - Multi value

منضبط جدول جديد فيه ال pk ال FR وال multi وفاض القبهيت دول composite
pk

7 - composite attribute

محل القيم الصغيرة في الجدول

8 - derived

من منطوق في الجدول

مجموع الجداول = مجموع ال many + multi + N: N

SQL

where

is → to deal with Null

لو عايز أجيب العلامة، بنستخدم Meo - للتعريف عن خانة واحدة و بنستخدم Meo % للتعريف عن القيم
like → قيلة بعده (أي عدد من الفئات)

لو عايز أجيب Range من d → a مثلاً [a-d]
لو عايز أجيب قيم محددة بـ a, b, c مثلاً [a, b, c]

select distinct top(n) []
↓
تتمنع التكرار للأعمدة
بجيب أول n عمود
order by

order by → للترتيب desc → order by عدد مصدر
لو الاسم متساوي المرتب الأكبر الأول
اختيار ترتيب عشوائي → newid()

Join

Cross join → لو عايز كل الاحتمالات
inner join → حيث المتساوية هو العمود المشترك قيمة بينهم
From employee, department
↓
بعد where dno=dno
* لو اجمع العمود متشابه بخط اسم الجدول الأول. العمود
* لو العلاقة N-N بنستخدم الجدول المشترك
where ... = ... مشترك
and ... = ... مشترك

outer join → * بجيب ناتج ال inner + كل عناصر أحد البصريين
* يشوف الجدول الب هطلع كل عناصره يمين ولا شمال واحد left, right
و ال where هتبقى on
* ال Full بجيب كل عناصر البصريين

self join * بربط جدول بنفسه

Group

Count(*) → عدد ال Row في الجدول
لو عايز اطلع أي Function من الأبريكيت على عمود بخط اجمع لعمود
مثلاً، ال select ويضيف صطر "اسم العمود" group by
Having → نفق Where بس لو على Function
لو عايز select 2 اقدر اجمعهم مع بعض بس عدد الأعمدة نكي حيلولة
Union → بقا المشترك بين الاثنين
intersection →
except → بجيب اللي في الجدول الأول مش في الثاني