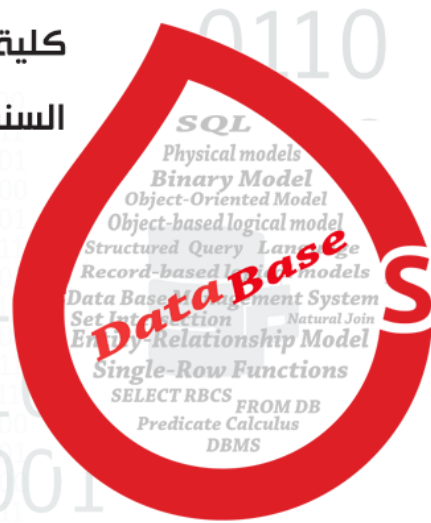


## ERD\_Part 3

عملي مشترك



## قواعد معطيات 1

25/04/2023

RB Informatcs;

السلام عليكم ورحمة الله وبركاته

في المحاضرة السابقة قمنا بحل مسألة مشفى الأسد الجامعي , وفي هذه المحاضرة سنقوم بحل مسألة أصعب بعض الشيء لعلّ الفكرة تتضح بشكل أكبر , و ستكون هذه المحاضرة هي المحاضرة الأخيرة في تصميم و رسم مخططات قواعد البيانات.

مسألة:

المطلوب إنشاء المخطط التصميمي ERD لشركة توزيع أدوية , تشتري الأدوية من شركات تصنيع الأدوية , وتخزنها في مستودعاتها الموزعة في مراكز التوزيع الأساسية في المحافظات , و من ثمّ توزيعها للمتعاملين (صيدلية , مشفى). لكل دواء تتعامل معه الشركات هنالك رقم مميز و اسم تجاري و اسم علمي و تاريخ انتهاء صلاحية. تشتري شركة التوزيع الأدوية من الشركات المصنّعة على شكل دفعات من خلال تقديم طلبات شراء للشركات المصنّعة. تضم كل دفعة تستلمها شركة التوزيع أدوية موزّدة من شركة تصنيع محددة. وتنتهي صلاحية الأدوية التي لها نفس الاسم التجاري في الدفعة الواحدة في نفس التاريخ, وتسدد قيمة الأدوية المستلمة بموجب فواتير .

و تتلقى شركة التوزيع طلبات شراء من المتعاملين و تقبض ثمن الأدوية المباعة بموجب فواتير .

ملاحظة : المطلوب إنشاء ERD لشركة التوزيع , أي أنّ المخطط بالكامل سيمثل

شركة التوزيع و طبيعة عملها , إنما ليس هنالك كيان

بحد ذاته entity لشركة التوزيع.



إذاً نحاول من نص المسألة استنتاج الكيانات الرئيسية في هذا المخطط.

ذكر أنّ هناك شركات تصنيع للأدوية ← companies

و بطبيعة الحال يوجد هنالك دواء ← medications .

ذكر أنّ هناك مستودعات للتخزين ← Warehouses .

و ذكر أنّ هذه الأدوية توزع على المستودعات المنتشرة ضمن المحافظات

أي أن هنالك بدايةً محافظات ← State .

و هنالك المنطقة المحددة ضمن هذه المحافظة ← Location .

و ذكر أنه يتم توزيعها على المتعاملين سواء مشفى أو صيدلية , أي أن هنالك كيان للمتعاملين ← Dealers .

**لاحظ:**

في هذا الكيان (Dealers) ليس هنالك داعٍ للتمييز بين المشفى و الصيدلية و إنما يهمني فقط هو وجود جهة تقوم باستلام الأدوية.

**ملاحظة:**

بالنسبة لـ Location فإنه يعتبر أمر في غاية الأهمية بالنسبة لشركة التوزيع (و التي نقوم بإنشاء المخطط من أجلها) إذ أن المسافة تؤخذ بعين الاعتبار عند التوزيع .  
مثلاً : كلفة التوصيل لمسافة قريبة ليست نفسها عندما تكون المسافة أبعد .  
و لذلك قمنا بإنشاء كيان كامل يمثل الـ Location ولم نقم بوضعها كـ attribute مثلاً.

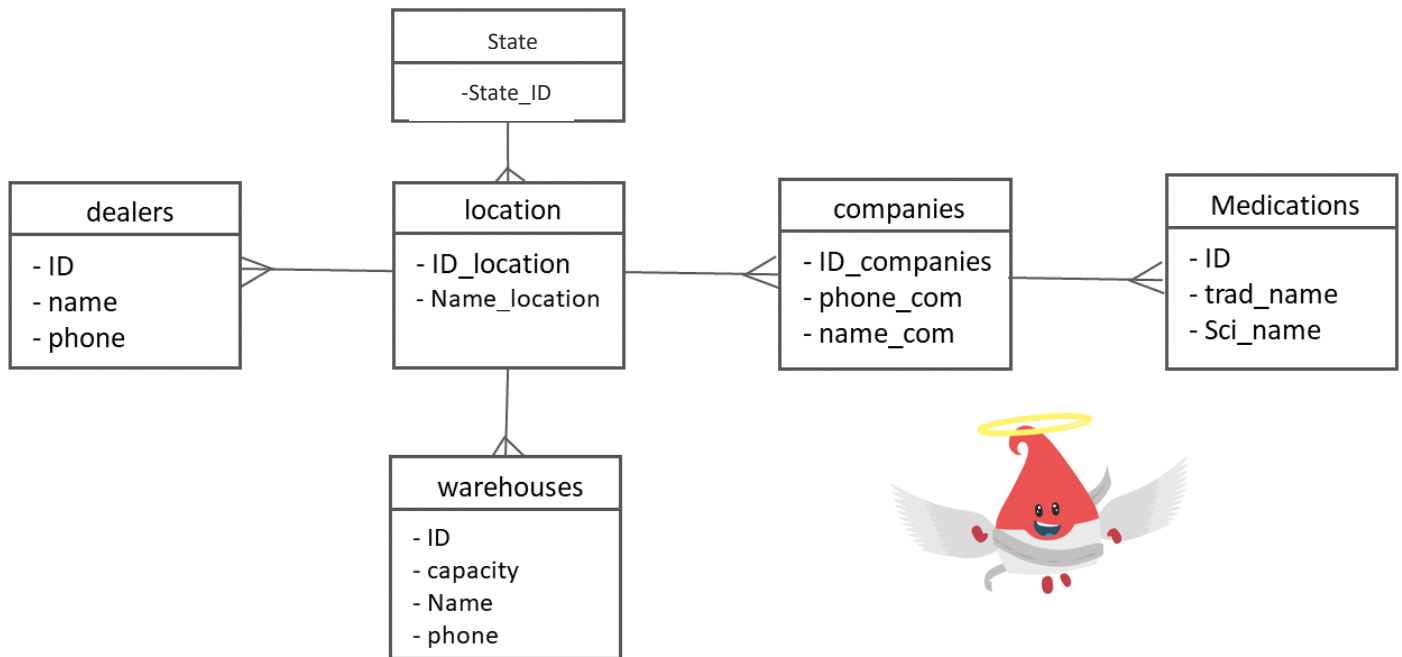
الآن نحاول أن نحدد أنواع العلاقات بين بعض الكيانات التي ذكرناها:

- نوع العلاقة بين الـ Location والـ State ؟!  
هي علاقة one to many من طرف الـ State إذ أن في كل محافظة يوجد عدّة مناطق موزعة ضمنها المستودعات .
- نوع العلاقة بين الـ Location و الـ Warehouses ؟!  
هي علاقة one to many من طرف الـ Location أي أنه ضمن المنطقة الواحدة يمكن أن يكون هنالك أكثر من مستودع.
- و كذلك الأمر بالنسبة للعلاقة بين الـ Location و الـ Dealers ...
- نوع العلاقة بين الـ Companies والـ medications ؟!  
علاقة one to many من طرف الـ companies , إذ أن شركة التصنيع تحوي عدّة أدوية ولكن الدواء نفسه هو تابع لشركة تصنيع واحدة و يجب الانتباه على أننا هنا نقصد هذا الدواء بحد ذاته و باسمه , و ليس تركيبته الدوائية على سبيل المثال.



Nobody Cares About Your Story  
Until You Win , So Win

أي أن المخطط المبدئي أصبح كالتالي :



### الآن بالنسبة لكيان ال medications و ال attributes الخاصة به و من خلال قراءة نص المسألة و المتطلبات , نستنتج أن الدواء يجب أن يكون له :

اسم تجاري trad\_name و اسم علمي sci\_name و تاريخ انتهاء صلاحية exp\_date ورقم دفعته وسعره price.

### ✎ ولنأخذ مثلاً مباشرة:

بفرض أنه لدينا دواء له ID1 و اسمه التجاري paracetamol و اسمه العلمي Acetaminophen تاريخ انتهاء صلاحيته هو 1/1/2026 وينتمي للدفعة الثالثة و سعره 2000 ليرة .  
بعد فترة قامت الشركة بتصنيع هذا الدواء نفسه مرة ثانية ذو الـ ID1 و اسمه التجاري paracetamol و اسمه العلمي Acetaminophen و تاريخ انتهاء صلاحيته هو 1/2/2026 وينتمي للدفعة الرابعة و سعره 4000 ليرة.

■ هل نقوم بإنشاء Record آخر له في هذه الحالة؟!  
كما تعلمنا فإنه من الخطأ أن نقوم بإنشاء 2 Records تتكرر فيها المعلومات ذاتها.  
إذاً بما أن هذه الخصائص attributes متغيرة مع الزمن فلا يمكننا وضعهم داخل الكيان الرئيسي و سنرى الآن كيف نقوم بتمثيلهم.

لكن لنعود للمسألة قليلاً ولنقوم بإعادة ترتيب أفكارنا

أثناء الحل نحن نقوم بتصميم مخطط قاعدة البيانات لشركة توزيع أدوية, و شركة توزيع الأدوية بشكل عام تتمثل مهمتها في جزأين:

**الجزء الأول :** يتمثل بشراء الأدوية من شركات التصنيع و تخزينها في المستودعات .

**الجزء الثاني :** يتمثل ببيع هذه الأدوية إلى المتعاملين من صيدليات ومستشفيات.



## ■ ملاحظة:

إنَّ علاقة كل من الـ companies و الـ Dealers مع الـ Location هي علاقة one to many من طرف الـ location.

إذ أنه يمكن أن يكون هنالك أكثر من شركة تصنيع ضمن هذه المنطقة و كذلك الأمر بالنسبة للمتعاملين.

نعود لنص المسألة.. فقد ذكر أن شركة التوزيع ستقوم بشراء الأدوية من شركات التصنيع من خلال تقديم طلبات شراء... وبالتالي استطعنا استنتاج وجود كيان آخر وهو buy\_order و لكن كيف سيتم ربطه مع الـ entities الموجودة لدينا ؟؟



بشكل منطقي فإن طلبات الشراء تخرج من شركة التوزيع إلى شركة التصنيع.

وفي طلبات الشراء نحن نقوم بطلب أدوية.

إذاً سنقوم بربط هذا الكيان مع الـ companies و الـ medications .

ما نوع العلاقة بين الـ companies و الـ buy\_order ؟

هي علاقة one to many من طرف الـ companies وذلك لأنَّ : شركة التصنيع بحد ذاتها من الطبيعي أنه سيصلها أكثر من طلبية شراء, ولكن من المؤكد أن طلبية الشراء ستكون متوجهة إلى شركة تصنيع واحدة.

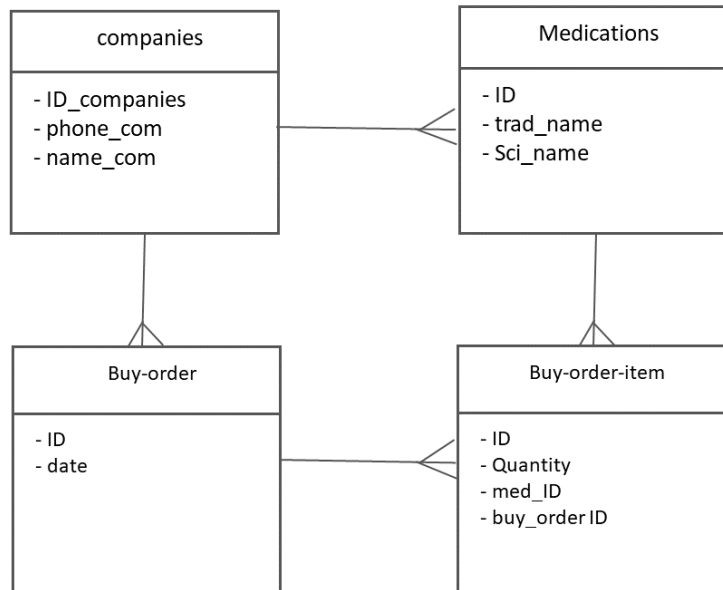
ما نوع العلاقة بين الـ medications و الـ buy\_order ؟

هي علاقة many to many و ذلك لأنه في طلبية الشراء يمكن أن يتواجد أكثر من نوع من الدواء والدواء بنفسه يمكن أن يكون متواجد بأكثر من طلبية.

■ لكن كما نعلم لا يجب أن يكون هناك علاقات many to many في تصميم الـ Database إذاً سنحتاج إلى كسر العلاقة السابقة من خلال كيان مساعد .. ولكن ما هو هذا الكيان ؟ وما هي الـ Data التي سيحويها؟ سنقوم بكسر العلاقة بكيان Item buy order وفيه الـ attributes الـ primary key و الـ foreign keys و الـ quantity.

إذ أنَّ الطلبية ستحتوي عدّة أنواع من الأدوية و كل نوع يوجد منه كمية معينة , كما أنه من الممكن أن تصل الطلبية على دفعات كما ذكر في نص المسألة.

■ إذاً أصبح لدينا كيان الـ Buy Order Item بالشكل التالي:



و لحد الآن نكون قد انتهينا من عملية الطلب من شركات التصنيع للأدوية التي أريد و الكميات التي أريد .  
والمرحلة التالية هي التجهيز للاستلام.

■ **ذكر في النص :** تضم كل دفعة ستستلمها شركة التوزيع عدّة أدوية مورّدة من شركة التصنيع.  
أي أنه من الممكن أن ترسل شركة التصنيع الكمية المطلوبة إلى شركة التوزيع دفعة واحدة و من الممكن أيضاً أن تقوم بإرسالها على عدّة دفعات .

**أي على سبيل المثال:** قامت شركة التوزيع بإرسال طلبية شراء إلى شركة تصنيع محددة و تحوي هذه الطلبية :

20000 علبة دواء من نوع Panadol

10000 علبة دواء من نوع paracetamol

فقامت شركة التصنيع بإرسال الكمية المتوفرة من هذه الطلبية وهي :

15000 علبة دواء من نوع Panadol

7500 علبة دواء من نوع paracetamol

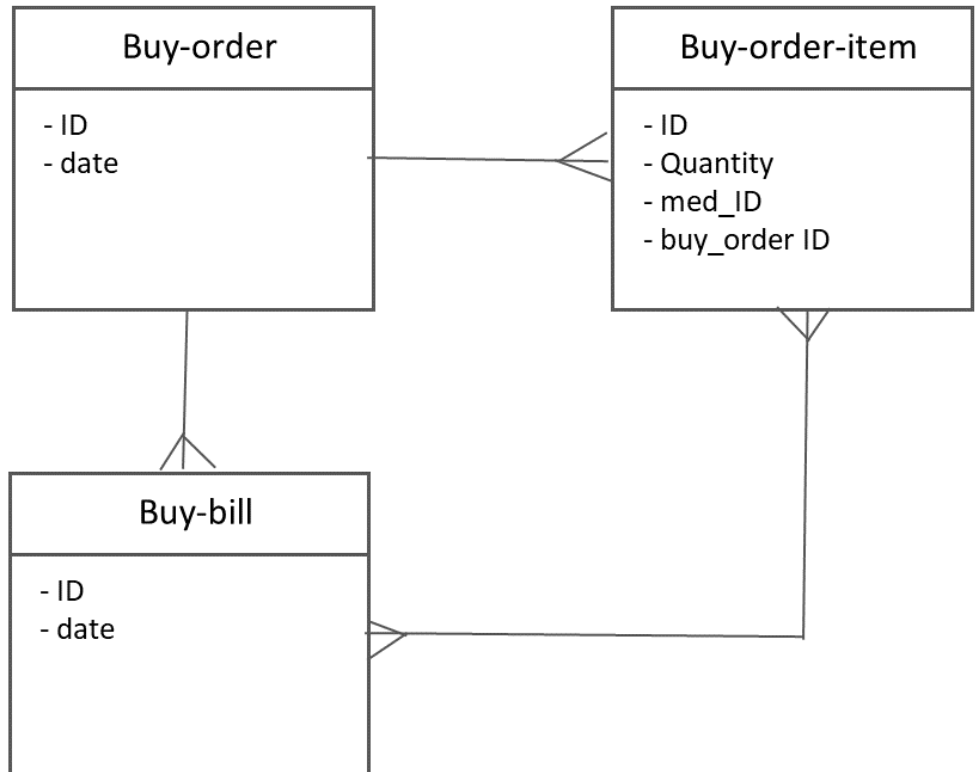
ثم بعد ثلاث أيام على سبيل المثال توفرت كمية جديدة من الأدوية فقامت بإرسال ما تبقى من الطلبية

5000 علبة دواء من نوع Panadol

2500 علبة دواء من نوع paracetamol

إذاً الآن تحصل عملية الاستلام , ووجود عملية استلام تعني وجود دفع الآن .

وذلك يفرض علينا وجود كيان جديد هو فاتورة الشراء , وستكون علاقته مع كيانين هما Buy\_order\_Item ,  
.Buy\_order



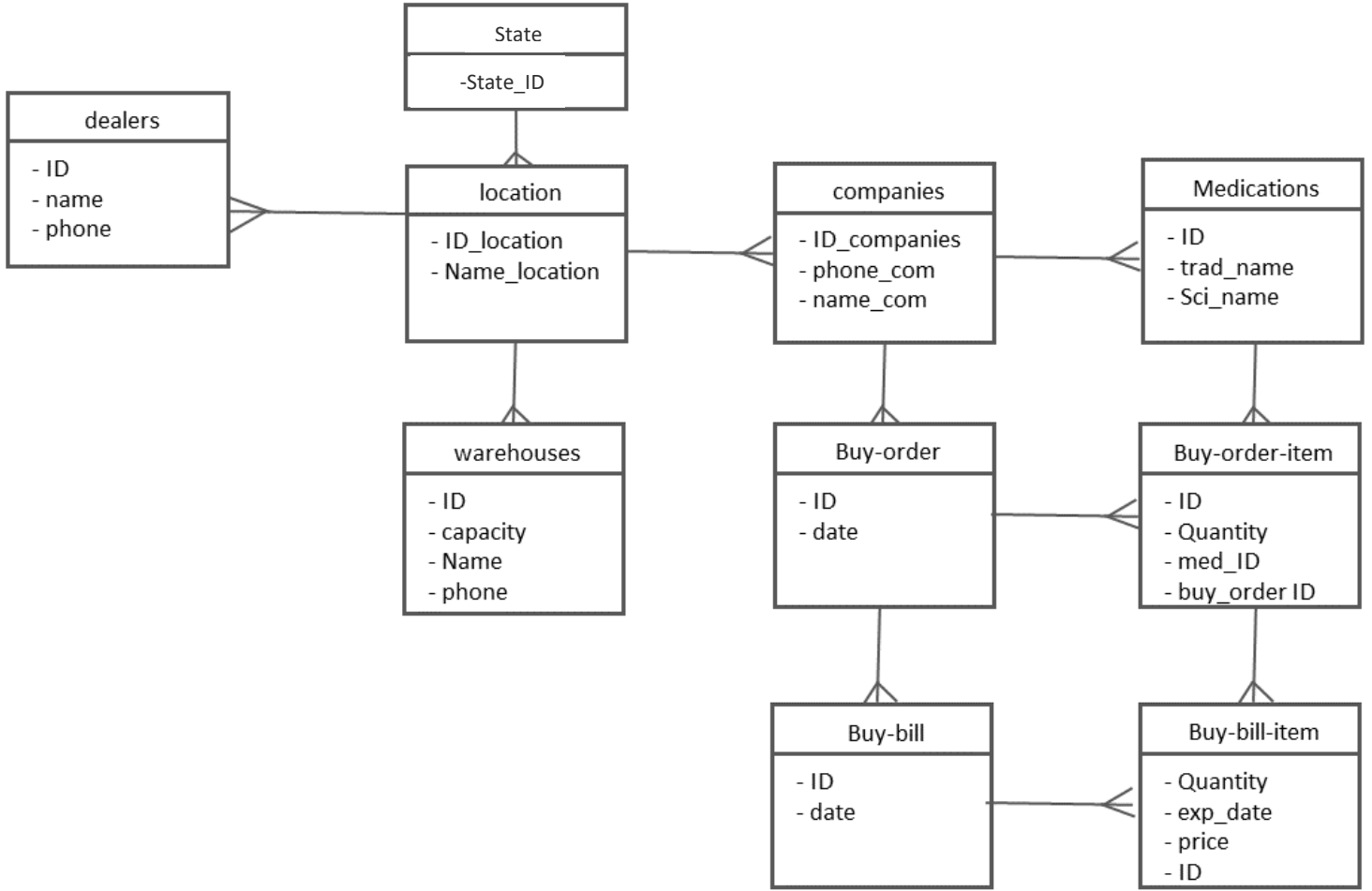
وكما نلاحظ أنها علاقة one to many من الـ Buy\_order إلى الـ buy\_bill وذلك كما تحدثنا بسبب أن كل طلبية يمكن أن تصل على عدة دفعات أو دفعة واحدة .  
أي أن يكون لها عدة فواتير أو فاتورة واحدة.  
وهي علاقة many to many بين الـ Buy\_order\_Item و الـ buy\_bill .  
أي أن كل فاتورة Bill يمكن أن تحوي عدة Items من الأدوية التي قمت بطلبها  
و Item الدواء الذي أقوم بطلبه يمكن أن يتوزع على أكثر من فاتورة  
و سنقوم بكسر هذه العلاقة بكيان Buy\_Bill\_Item وفيه الـ attributes الـ primary key و foreign keys و quantity و exp\_Date و Price و batch\_name .  
حيث أن الـ quantity هي الكمية التي تم استلامها من أصل الكمية التي تم طلبها .  
و الـ exp\_Date هي تاريخ انتهاء الصلاحية لكل دفعة من Item محدد من الـ Items التي قمت بطلبها .  
والـ Price سعر كل دفعة من Item محدد من الـ Items التي قمت بطلبها .  
و الـ batch\_name هو رقم الدفعة.

إذا فقد أصبحت العلاقة one to many من الـ Buy\_bill إلى الـ Buy\_bill\_Item .  
و أيضاً one to many من الـ Buy\_order\_Item إلى الـ Buy\_bill\_Item .

- ولاحظنا أننا أضفنا الـ attributes كالـ price و exp\_Date في هذا الكيان .. لأننا فقط عند الاستلام نهتم بسعر هذا الـ Item أو بمدة صلاحيته.
- فمن غير المنطق على سبيل المثال إضافتها عند الطلب أو الشراء أو بمعنى آخر (أريد 500 علبة من الدواء الذي تاريخ انتهاء صلاحيته كذا و سعره كذا) فبالأكيد هذا الكلام غير منطقي إذ أننا لا نهتم بهذه التفاصيل إلا عند الاستلام.
- و بمعنى آخر يمكننا أن نشبه كيان الـ buy\_bill برأس أو header الفاتورة و التي نراها عند التسوق وشراء أغراض معينة .
- فهو يحوي المعلومات الأساسية كالـ ID رقم الفاتورة و الـ Data التاريخ الذي أصدرت به.  
و الـ Buy\_bill\_Item أشبه بالـ Body الخاص بهذه الفاتورة, فهو يحوي الكميات الموجودة من كل صنف مع السعر و تاريخ الصلاحية....



If You Don't Stand For Something,  
You'll Fall For anything.



و لحد الآن نكون قد انتهينا من عملية الاستلام, و المرحلة التالية هي التخزين في المستودعات .

#### ملاحظة:

من المتعارف عليه في إدارة المستودعات أن كل مستودع يجب أن يكون فيه دفتر إدخال – دفتر إخراج – دفتر توالف

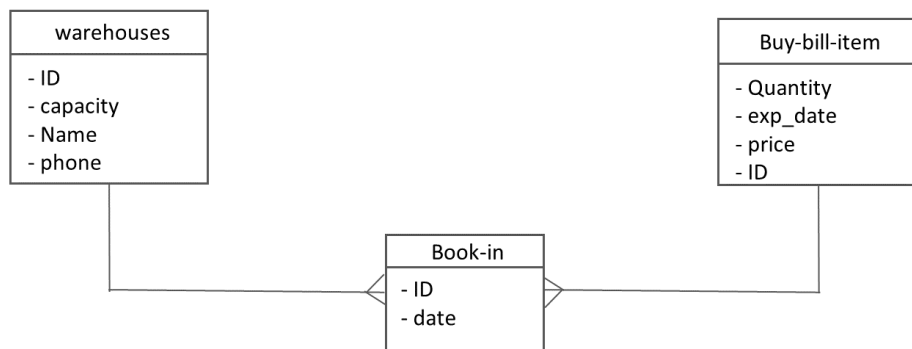
بالنسبة للتخزين فيمكننا مباشرة تشكيل علاقة بين الـ Buy\_bill\_Item و الـ warehouse

و نوع هذه العلاقة هي many to many.

إذ أنني من ضمن الأدوية التي قمت باستلامها \_ يمكنني تخزين نوع من الأدوية في العديد من مستودعات التخزين.

و أي مستودع للتخزين يمكن أن يحوي على أكثر من نوع من الأدوية التي قمت باستلامها

ونقوم بكسر هذه العلاقة باستخدام كيان الإدخال Book\_in.

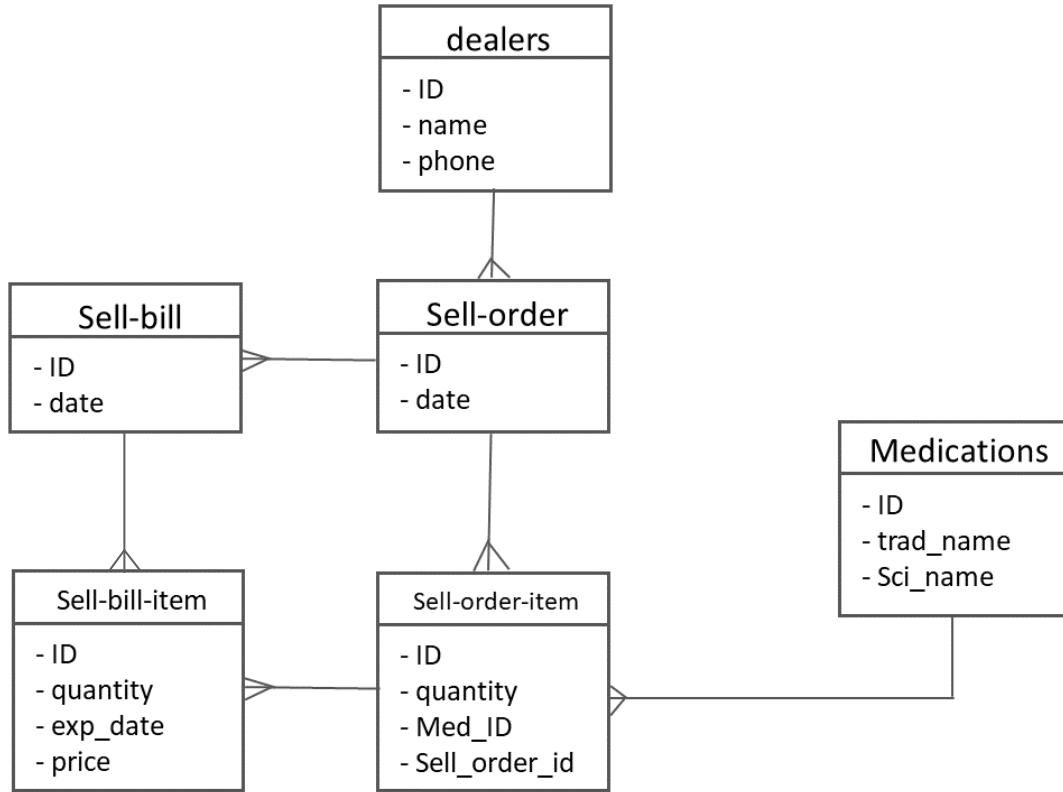




و ذكرنا أن المسألة تتمثل بجزأين :

- الجزء الأول و هو شراء شركة التوزيع الأدوية من شركات التصنيع .. وانتهينا منه.
- الجزء الثاني و هو بيع شركة التوزيع الأدوية للمتعاملين ... وهو تقريباً مماثل للجزء الأول من ناحية الـ Design.

- أي تماماً كما قمنا بتمثيل عملية الشراء من شركات التصنيع سنقوم بتمثيل عملية المبيع إلى المتعاملين :



ولكن هنا يوجد علاقة بين الـ Sell\_Order\_Item و الـ medications لماذا؟؟

لأنّ المتعامل هنا أو الصيدلاني على سبيل المثال يقوم بطلب عينات من أدوية معينة ولا يهمه فيما إذا كانت هذه الأدوية تابعة لشركة واحدة أو لشركات تصنيع عدّة , فهو يهتم فقط بأنه يريد هذا الدواء المعين .. على عكس آلية الطلب التي تكون من شركة التوزيع إلى شركة التصنيع.

الآن بعد أن تمّ الطلب من قبل المتعاملين أو الصيدلية تبقى إيصال الأدوية التي تمّ طلبها من قبل الصيدلية أي التوزيع.

فكما فعلنا في الجزء الأول يمكننا مباشرة تشكيل علاقة بين الـ Sell\_Bill\_Item و الـ warehouses , نوع هذه العلاقة هو many to many و نقوم بكسرها من خلال كيان الإخراج Book\_out.

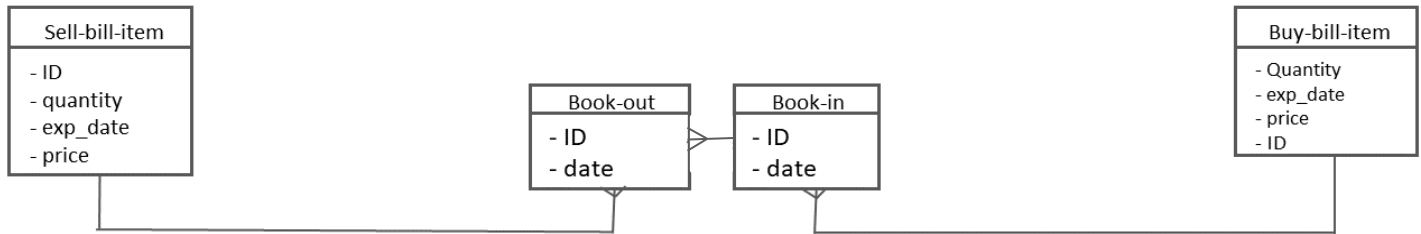




## لكن السؤال:

هل من الصحيح ربط كل من كياني book\_out و book\_in مع الwarehouses بشكل مباشر مثلما فعلنا ؟؟

الجواب هو أن الأصح ربط ال book\_in و book\_out مع بعضهما البعض بهذا الشكل :



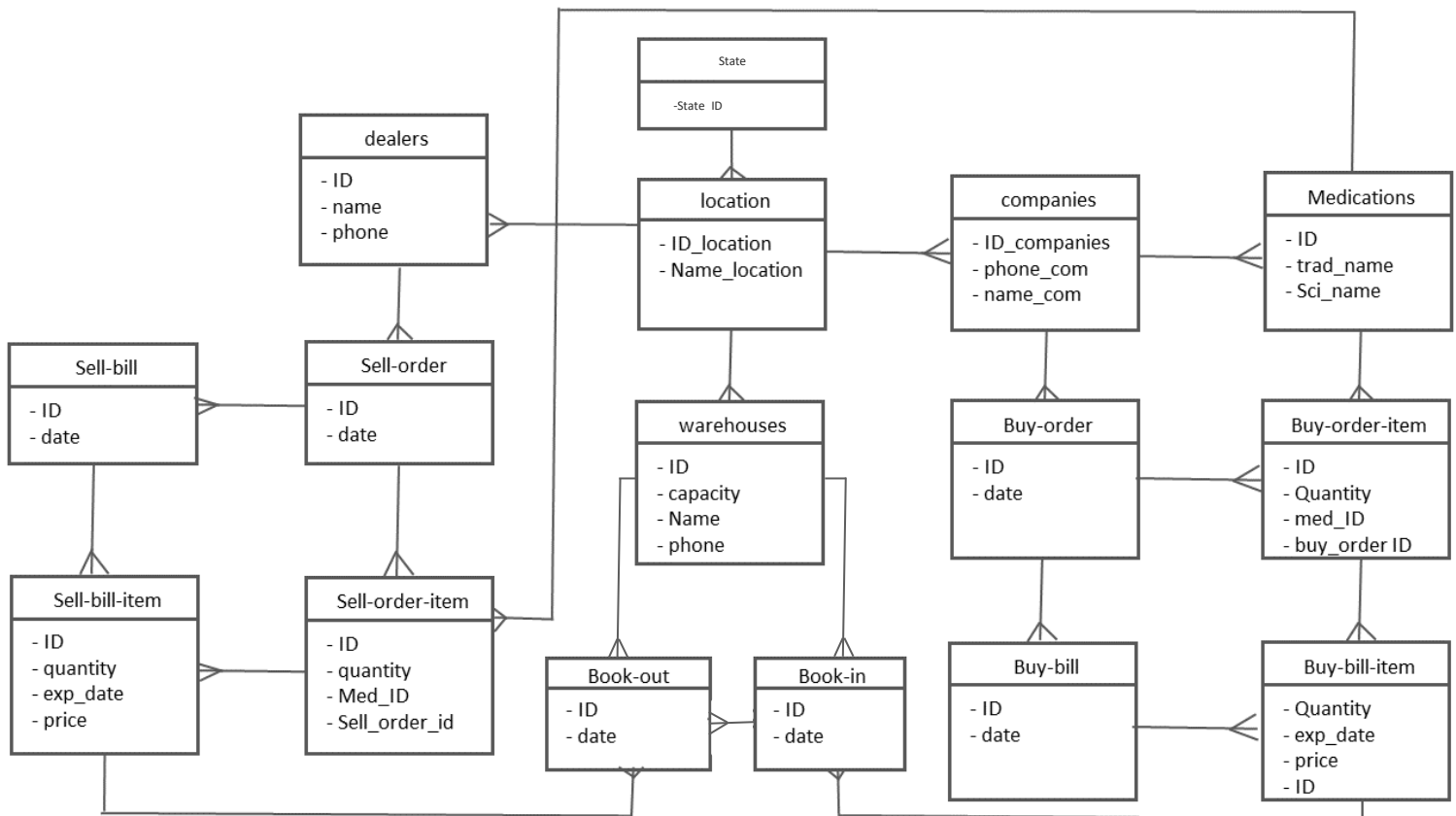
وذلك لأنه لا يمكن أن نقوم بعمل book\_out لبضاعة ما من دون أن تكون هذه البضاعة موجودة فعلياً في ال book\_in.

وأيضاً لتجنب ضياع معلومات في غاية الأهمية.

فلو أن علاقة ال book\_out مع الwarehouses فنحن في هذه الحالة لن نستطيع معرفة الكمية الأساسية للبضاعة التي تم تخزينها ولا مدة صلاحيتها ولا حتى سعرها .

إذ أن هذه المعلومات لا يمكننا الوصول إليها إلا عبر ال book\_in الذي بدوره يحوي معلومات ال buy\_bill\_Item

## و بالتالي يكون مخطط ال ERD النهائي للمسألة السابقة كالتالي:



--انتهت المحاضرة--