

# Cinema Database Report

## 1) تقسيم الطبقات:

طبقة (DAL(Data access layer) وفيها 3 بنى نستطيع من خلالها التعامل مع قاعدة المعطيات مباشرة وتربط الواجهات البيانية مع قاعدة المعطيات عن طريق تواجدها:

DatabaseCallFunction نتمكن من خلاله تنفيذ الإجراءيات والتوابيع الموجودة في قواعد المعطيات

DatabaseConnect نستطيع من اجراء اتصال مع قاعدة المعطيات

DatabaseUse قمنا بتنفيذ جميع التوابيع التي تربط الواجهات مع قاعدة المعطيات مثلا ValidateUser() يقوم بالتحقق من ان المستخدم موظف ويحدد نوعه وهكذا تم فصل الواجهات البيانية تماما عن قاعدة المعطيات وإجراءياتها.

## 1) الواجهات :

قمنا بإنشاء واجهة وحيدة تحوي عدة tabs ولكل tab يوجد user control له مع data grid view وازرار لاختيار الاجرائية المطلوب تنفيذها من البرنامج بالتالي قمنا بتجزئة كل عملية (إضافة زبون, إضافة فلم, ...) مستقلة عن بعضها

الواجهة الاولى تحوي ادخال ID الموظف لأننا افترضنا وجود رقم معرف لكل موظف سري يتمكن من خلاله بالدخول والقيام بالعمليات الخاصة به إذ إن موظف الاستقبال مسؤول عن إضافة زبون جديد وإضافة حجز جديد اما موظف الإدارة يستطيع القيام بالأمر الأخرى الموجودة.

وضعنا في بعض ال tabs زر بحث عن ID او اسم فلم لتسهيل عملية قراءة البيانات من الواجهة.

يمكن التعديل على بعض الحقول في data grid view لأي tab (بعض الحقول وليس كلها لأن بعض الحقول مرتبطة بأمور مختلفة وتغييرها يمكن ان يولد اخطاء في البرنامج . مثلا تغيير اسم الفلم وهو يحوي رقم معرف له مسبقا تركه واضافة فلم جديد افضل من تغييره.

وبشكل عام الواجهات جميعها واضحة وسهلة التعامل معها.

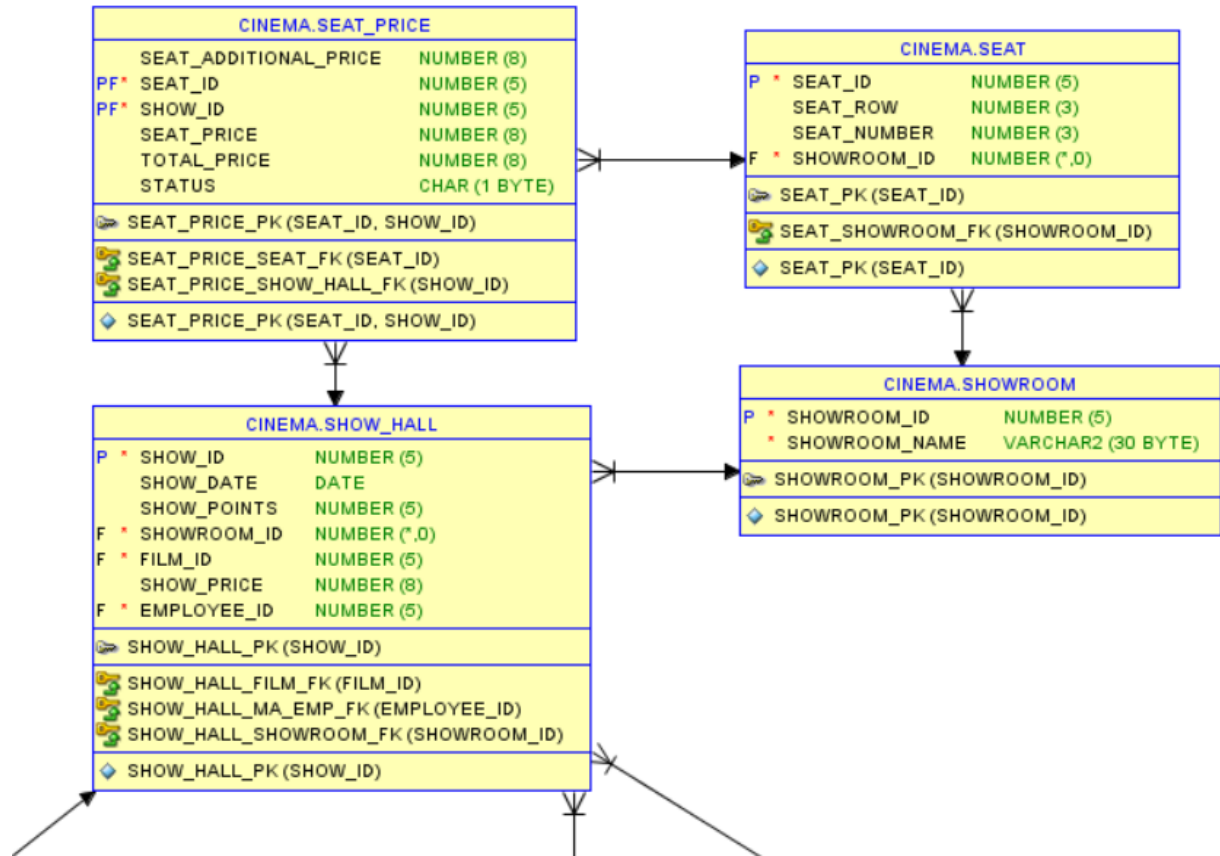
## 2) تنجيز المطلوب:

تقريبا 80% من التنجيز للمطلوب موجود في stored procedures مربوطة ب trigger او غير تنفذ عند الضغط على زر معين

هناك بعض الإجراءيات مكتوبة في التطبيق في databaseUse

معالجة الحسومات والاسعار تمت كلها في عدة stored procedures

## 3) بنية قواعد المعطيات:

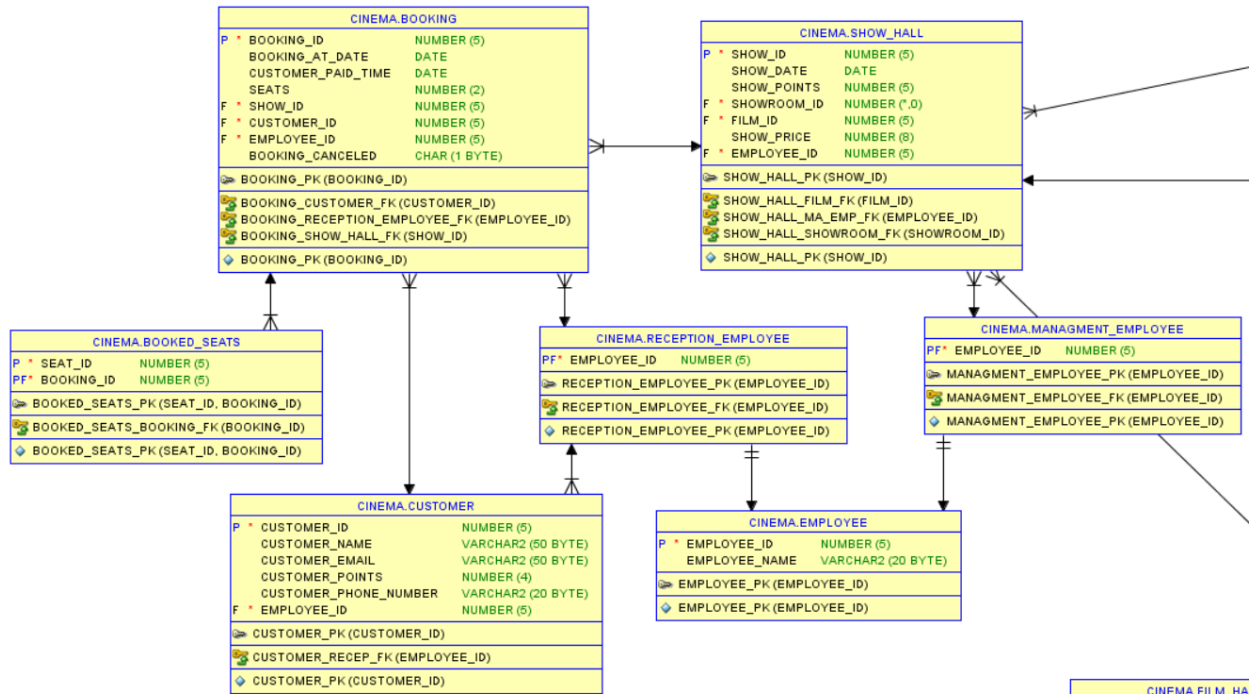


Show\_hall هو العرض

Seat\_Price فيه سعر كل مقعد في عرض ما وفيه حالة المقعد اذا كان محجوز ام غير محجوز

Seat معلومات عن المقعد (صف , رقم المقعد ضمن الصف)

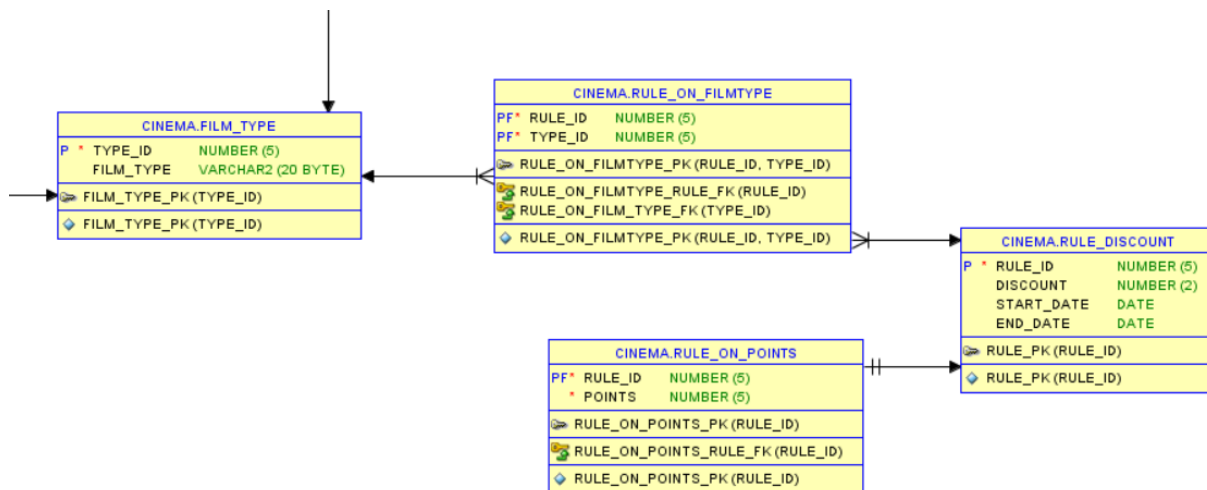
Showroom وهي صالة العرض لكل صالة رقم معرف واسم ما.



كل عرض فيه اكثر من حجز Booking وفيه موظف الذي حجز ل Customer وفيه Customer\_paid\_time بدل اذا دفع الزبون او لم يدفع

Booking\_Canceled ( 0 اذا لم يلغى الحجز , 1 اذا تم إلغاء الحجز)

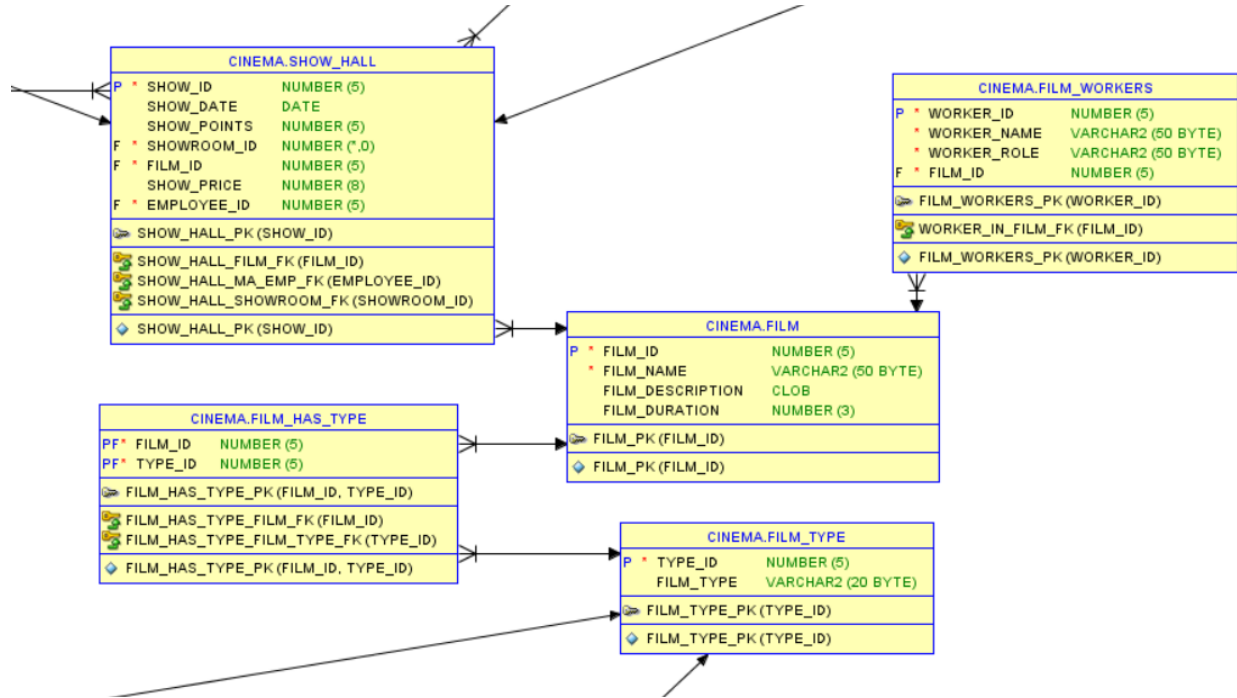
Employee يتكون من نوعين استقبال وإدارة.



جداول الحسومات

يمكن انا يكون هناك حسم عام من فترة لفترة وهو له اولوية ثم اولوية حسم على نمط ثم اولوية النقاط اخيرا لكل حسم له تاريخ بداية ونهاية وقيمة الحسم ورقم معرف له.

في جدول film\_type يوجد انماط ثابتة يتم ادخالها او لا قبل البدء بالتطبيق (موجود في Notes.txt)



الفلم له اكثر من نمط وتم التعبير عن جدول كسر العلاقة بين الافلام والانماط من خلال جدول FILM\_HAS\_TYPE

لكل فلم له اكثر من ممثل او مخرج او .. (Worker\_role) موجود في Film\_workers

#### (4) شرح بعض النقاط:

لم يتم تنجيز بطاقة زبون لكنها محفوظة ويمكن استدعائها

يمكن انا يتم استخدامها عند اتصال البرنامج مع طابعة مثلا حيث لم نجد منطقيا انا يكون هناك جدول في قاعدة المعطيات يعبر عنها لانه يمكن جلب اي معلومة لها عن طريق جدول Booking مع SHOW\_HALL (نفس الشيء تم تطبيقه مع Customer\_interests) حيث لم يتم وضع جدول به فقط تابع يولده ويطلع للموظف ما هو اكثر فلم شاهده الزبون في CustomerTab

تم افتراض انه لا يوجد تقاطع بين حسمين من نفس النمط لكن يجب مناقشته واخذ الاكبر لكن لم نجد هناك فائدة من وجود حسمين بنفس الفترة بنفس النوع (مثلا على النقاط) يمكن انا يوجد حسمين مختلفي النوع وقد تم مناقشة اولوية الانماط سابقا.

الحسم العام والحسم على نمط الفلم يتم اضافته من خلال trigger اما حسم النقاط يتم استخدامه عند اضافة حجز ولم يكن هناك حسم من نمط اخر تتم تغيير قيمة total price في seat price والتي هي تعبر عن سعر المقعد الكلي بعد الحسمات وبعد اضافة قيمة اضافية لسعر المقعد additional\_seat\_price

عند كل اضافة عرض يتم توليد جميع المقاعد بحالة غير محجوزة في جدول seat\_price

اسم جدول seat\_price لا يعبر عن معنى الجدول بعد اضافة حالة المقعد له  
سابقا كان لدينا جدولين seat\_price و seat\_status وقررنا دمجهما ولكن لم نغير اسم الجدول لضيق الوقت.