Allergo Music Store

[Company name] | [Company address]

Design Phase – Table Identification

Makthum, KELLY, FLORA, chamila

[Year]

|  |
| --- |
| **Item (UPC, sellPrice, taxable )  -**  Represents – Strong Entity  Entity set – item  Primary Key – UPC  Foreign Key  - NA  Constraints – NA  **Functional Dependency**  UPC -> sellprice , taxabe |
| **Stored(UPC,name,stock)**  Represents – Relationship  Relationship entity  – stored  Primary Key – UPC , name  Foreign Key  - UPC references item table  , name references store table  Constraints – NA  **Functional Dependency**  UPC -> name , stock |
| **Store (name, address , city , phone , storeType)**  Represents – Store Entity  Entity Set  – store  Primary Key – store name  Foreign Key  - NA  Constraints –  store should be either regular store or warehouse  **Functional Dependency**  Name -> address , city , phone , storetype  Question  Did we create new tables for regularstore and warehouse ????? |
| **Book(UPC, title , publisher )**  Represents – Store Entity  Entity Set  –  Book  Primary Key – UPC  Foreign Key  - UPC references item table  **Functional Dependency**  UPC -> title , publisher |
| **HasAuthor( UPC , name) -  name is not null**  Represents – Relationship  Set  – Entity Book and Relationship set : HasAuthor  Primary Key – UPC , name  Foreign Key  - UPC references item table  Constraints –  Each author must have authored one or more books and each book should have one or more authors  **Functional Dependency**  UPC , name  -> upc , name  ( trivial dependency ) |
| **Author(name)**  Represents – Entity  Set  – Entity Author  Primary Key –name  Foreign Key  -  Constraints –  Each author must have authored one or more books  **Functional Dependency**  UPC , name  -> upc , name  ( trivial dependency ) |
| **Score ( UPC ,  title , publisher )**  Represents – Entity  Set  – Entity score  Primary Key – UPC , name  Foreign Key  - UPC references item table  Constraints –  N/A  **Functional Dependency**  UPC - > title , publisher |
| **Hascomposer ( UPC , name)**  Represents – Relationship  Set  – Entity score  and Relationship set : composer  Primary Key – UPC , name  Foreign Key  - UPC references item table , name references composer table  Constraints –  Each score must have one or more composer and each composer mush have composed one or more music sheet .  **Functional Dependency**  Trivial dependency |
| **Composer ( UPC , name)**  Represents – Entity  Set  – Entity composer  Primary Key –name  Foreign Key  - N /A  Constraints –  each composer mush have composed one or more music sheet .  **Functional Dependency**  Trivial dependency |
| **CDorDVD(UPC, title , company , year , type , category )**  Represents – Entity  Set – Entity CDorDVD  Primary Key – UPC  Foreign Key - UPC references item table  Constraints – N/A .  **Functional Dependency**  Upc -> title , company , year , type , category  Ad2 : Category -> company  BCNF DECOMPOSITION  CDorDVD (**Upc**,title,year , type , category ) and   Cat\_company (**category** , company) |
| **Hassong (UPC , title )**  Represents – Relationship and Entity  Set – Entity CDorDVD Relationship set : HasSong  Primary Key – UPC , title  Foreign Key - UPC references item table , title references song table  Constraints – Each cd or dvd has one or more songs and each must be present in atleast one cd or dvd  **Functional Dependency**  Trivial dependency |
| **song (Title)**  Represents – Entity  Set – Entity Song  Primary Key –title  Foreign Key – N/A  Constraints – N/A  **Functional Dependency**  Trivial dependency |
| **Leadsinger( UPC,name )**  Represents – Relationship and Entity  Set – Entity CDorDVD Relationship set : Leading singer  Primary Key – UPC , name  Foreign Key - UPC references item table , name references singer table  Constraints – Each cd or dvd has one or more singer and each cd or dvd must have at leat one singer  **Functional Dependency**  Trivial dependency |
| **Singer(name )**  Represents –Entity  Set – Entity singer  Primary Key –name  Foreign Key –N/A  Constraints –N/A  **Functional Dependency**  Trivial dependency |
| **Shipment*(*sid,date , sup\_name,store\_name)**  Represents –Entity  Set – Entity shipment  Primary Key – Sid  Foreign Key – sup\_name references supplier table and store\_name references store table  Constraints – Each shipment must have a supplier name and each shipment must to shipped to only one store  **Functional Dependency**  **Sid ->** date , sup\_name,store\_name  AD1 : Sid -> sup\_name , store name |
| **Supplier( name,status,address,city )**  Represents –Entity  Set – Entity supplier  Primary Key – name  Foreign Key – N/A  Constraints – N/A  **Functional Dependency**  Name - > status, address , city  AD3 : City , name -> status, address |
| **Shipitem(sid,upc,quantity)**  Represents –relationship  Set – shipitem  Primary Key – sid,upc  Foreign Key – sid references shipment table and upc references item table  Constraints – each shipment must contain one or more items  **Functional Dependency**  **Sid** , UPC -> quantity  Sid -> upc ,quantity |
| **Supplies (name,UPC, supprice)**  Represents –relationship  Set – supplies  Primary Key – name,UPC  Foreign Key – upc references item table and name references supplier table  Constraints – each item must be supplied atleast by one supplier and supplier might supply zero or more items  **Functional Dependency**  Name , UPC -> supprice |
| **Customer (cid,password , name , address , city , phone )**  Represents –Entity  Set – customer  Primary Key – cid  Foreign Key – N/A  Constraints  - N/A  **Functional Dependency**  cid -> password , name , address , city , phone |
| **Order(receiptId,card#,expire,delivereddate,cid,storename)**  Represents –relationship  Set – custorder  Primary Key – receiptid  Foreign Key – receipted references order table ,cid references purchase table  Constraints – Each custorder must be related to only one customer  **Functional Dependency**  **Receiptid - >** card#, expire , deliverydate , cid , storename , ordertype |
| **Purchase (receipted,date**)  Represents –entity  Set – purchase  Primary Key – receiptid  Foreign Key – N/A  Constraints – N/A  **Functional Dependency**  Receiptid -> date |
| **cardPurchase(receiptId, card# ,expire)**  Represents –entity  Set – cardpurchase  Primary Key – receiptid  Foreign Key – receipted references purchase table  Constraints – N/A  **Functional Dependency**  Receiptid ->  card# , expire |
| **Instore (receipted,storeName,purchasetype)**  Represents –Entity  Set – purchasestore , instore  Primary Key – receiptid  Foreign Key – receiptid references purchase table , name references store table  Constraints – each instore purchase must be related to only one regular store  **Functional Dependency**  Receiptid -> storename , purchasetype  Question  Did we create new table for cash purchase entity ??? |
| **purchaseItem(receiptId,UPC,quantity**)  Represents –relationship  Set – purchaseitem  Primary Key – receiptid,UPC  Foreign Key – receiptid references purchase table and UPC references item table  Constraints – each purchase should have atleast one item  **Functional Dependency**  Receiptid , UPC -> quantity |
| **ReturnItem(UPC,retId,quantity)**  Represents –relationship  Set – returnitem  Primary Key – retid , upc  Foreign Key – retid references return table and upc references item table  Constraints – Return should include one or more item  **Functional Dependency**  Upc , retid -> quantity |
| **Return(retid,date,purchaseid ,storename)**  Represents –entity and relationship  Set – return , basedon , returnstore  Primary Key – retid  Foreign Key – receiptid,name  Constraints – each return should be based on only one purchase and each return should be made only at one store  **Functional Dependency**  Retid ->date , purchaseid , store name |