**DATA ORGANIZATION FOR**

**E-SHOPPING**



ABSTRACT:

A Database Design for Online Shopping:

Internet has turned into an asset in current business, in the same scenario electronic shopping picked the criticalness from business person’s and client’s perspective. In business times, electronic shopping gives open doors for business and client which makes relative shopping conceivable. To an overview, many of the customers from online stores are not cautious and not settled on a choice to remain focused on webpage in the initial timings. The e-trade business is most utilized in world wide web and main objective is to offer products and administrations of web.

This manages building up an e-business website for online shopping. It enriches the client with index of many things accessed to buy in the store. The end goal is to encourage online shopping with basket to the client. The framework is executed utilized a 3-level technology with backend database, a main level of Microsoft Internet Information Services(IIS) and ASP.NET, web programming as front end customer. The end goal is to build up an electronic-trade site, technologies must be contemplated. The incorporate multi layered design, server and customer side scripting procedures, executing innovations like programming dialect, social databases like MySQL. So mainly this objective to develop a website where a consumer is provided with shopping cart application and known about technologies used to develop an application. This provides the discussion of each underlying technologies to create and implement an e-website.

# Contents

1.0 INTRODUCTION 3

[Objectives 3](#_TOC_250012)

[Scope 3](#_TOC_250011)

2.0 GENERAL DESCRIPTION 3

[Product Description 3](#_TOC_250010)

[Problem Statement 3](#_TOC_250009)

[Advantages of Computer based Library Management System 4](#_TOC_250008)

[Disadvantages 4](#_TOC_250007)

[New System 4](#_TOC_250006)

[Features](#_TOC_250005) 4

[Facilities](#_TOC_250004) 4

3.0 SYSTEM OBJECTIVES 4

[4.0 SYSTEM REQUIREMENTS](#_TOC_250003) 4

[Operational Requirements 6](#_TOC_250002)

Non Functional Requirements 6

Product Requirements 6

5.0 SYSTEM DESIGN 5

[Conceptual Design (ER Diagram)](#_TOC_250001) 5

6.0 NORMALISATION…………………………………………………………………………..9

7.0 PHYSICAL DATABASE DESIGN…………………………………………………………15

1. [CONCLUSION](#_TOC_250000) 27
2. **INTRODUCTION**

An online store is virtual store on internet where customers can browse catalog and select interest of products. The selected items can be collected in cart where at checkout time, the items in shopping cart will be represented as order. At the same time, more information will be needed to complete transaction. The customer will be asked to fill or select a billing address, a shipping address, a shipping option and payment information such as credit card number. An email notification is sent to customer as soon as order is placed to the email address of customer.

**Objectives**

1. To shop from your home without having to step out of door.
2. To easily save money and compare prices from website to website
3. Online resellers tend to sell at a lower price due to less overhead expenses.

**Scope**

Online shopping is a trend now a days. Everyone want to save money and also time and online business offers both promotion for users.

1. **GENERAL DESCRIPTION**

**Product Description**

E-shopping is computerized system which help user to manage customer daily activity in electronic format. It reduces the risk of time consuming, price etc. It can help user user to manage the transaction more effective and timesaving.

## **Advantages of online shopping**:

* + - Save Time –With just a couple of clicks of the mouse, you can purchase your shopping orders and instantly move to other important things, which can save time.
    - Save Fuel – The market of fuel industries battles from increasing and decreasing its cost every now and again, but no matter how much the cost of fuel is it does not affect your shopping errands. One of the advantages of shopping online is that there is no need for vehicles, so no purchase of fuel necessary.
    - Comparison of Prices – The advanced innovation of search engine allows you to easily check prices and compare with just a few clicks. It is very straightforward to conduct price comparisons from one online shopping website to another. This gives you the freedom to determine which online store offers the most affordable item you are going to buy.

## **Disadvantages**

Despite the success of purchasing through online shopping stores, there are still some disadvantages that most people complain about. These include:

• Personally, Check the Item – If you are one of those shoppers who want to touch, see, and test the product personally, at E-shopping, you are not able to do so. Online stores are only showing product description and photos of the merchandise, which can be a disadvantage for many online shoppers.

• Diminished Instant Satisfaction – Unlike buying at retail stores, you can use the product instantly after you buy it, which can be satisfying. However, online shopping requires patience to wait for the item to arrive at your door step about 2 to 3 days or even longer depending on the location you've ordered it from.

## **New System**:

* + - New features which will be incorporated with database design for online shopping which can handle all queries regarding shopping and maintain its process.

### **Features:**

* + - Details of each Customer is stored in database
    - Large volume of goods are easily handled
    - List of Customers with same category can be viewed
    - Could see the list of ordered products.
    - Borrowing of book become easy because in database librarian can find in seconds if this is book is issued to someone else or not
    - Customer has password and login details of database so as to secure process

## **Facilities:**

* + - **STORE** can find if the products are available or not, their description and how much the price is declared.
    - **Customer** can borrow & return goods/products easily.

1. **SYSTEM OBJECTIVES**

* Improvement sales in products
* Cost
* Time

1. **SYSTEM REQUIREMENTS**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Hardware |  |  | capacity |  |
|  | | |  | | |
| PROCESSOR | | | INTEL CORE PROCESSOR OR BETTER PERFORMANCE | | |
| HARD DISK SPACE | | | MINIMUM 3 GB FOR DATABASE USAGE FOR FUTURE | | |
| MEMORY | | | 1GB RAM OR MORE | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Software |  |  | Version |  |
|  | | |  | | |
| DATABASE | | | Oracle 11g | | |
| OPERATING SYSTEM | | | WINDOWS VISTA ,WINDOWS7, UBUNTU | | |
| SQL Developer | | | Open Source, Latest Version | | |

**Non Functional Requirements:**

Reliability Requirement: The system accurately performs registration, product search.

Usability Requirement: The system is designed user friendly environment so that user can add the product in cart and delete easily.

1. **SYSTEM DESIGN**

There are three phases in database design model

1.Conceptual

2.Logical

3.Physical

**Conceptual Design**: Entity-relationship model is a graphical representation of entities and their relationships to each other used in regard to organization of data within databases. It is cleared from discussion that main entities of online shopping are customer, products which store any queries. Using ER Design redundancy of many items can be reduced among entities.

**Cardinality of Relationship:**

A customer can have many orders

A product can have many customers

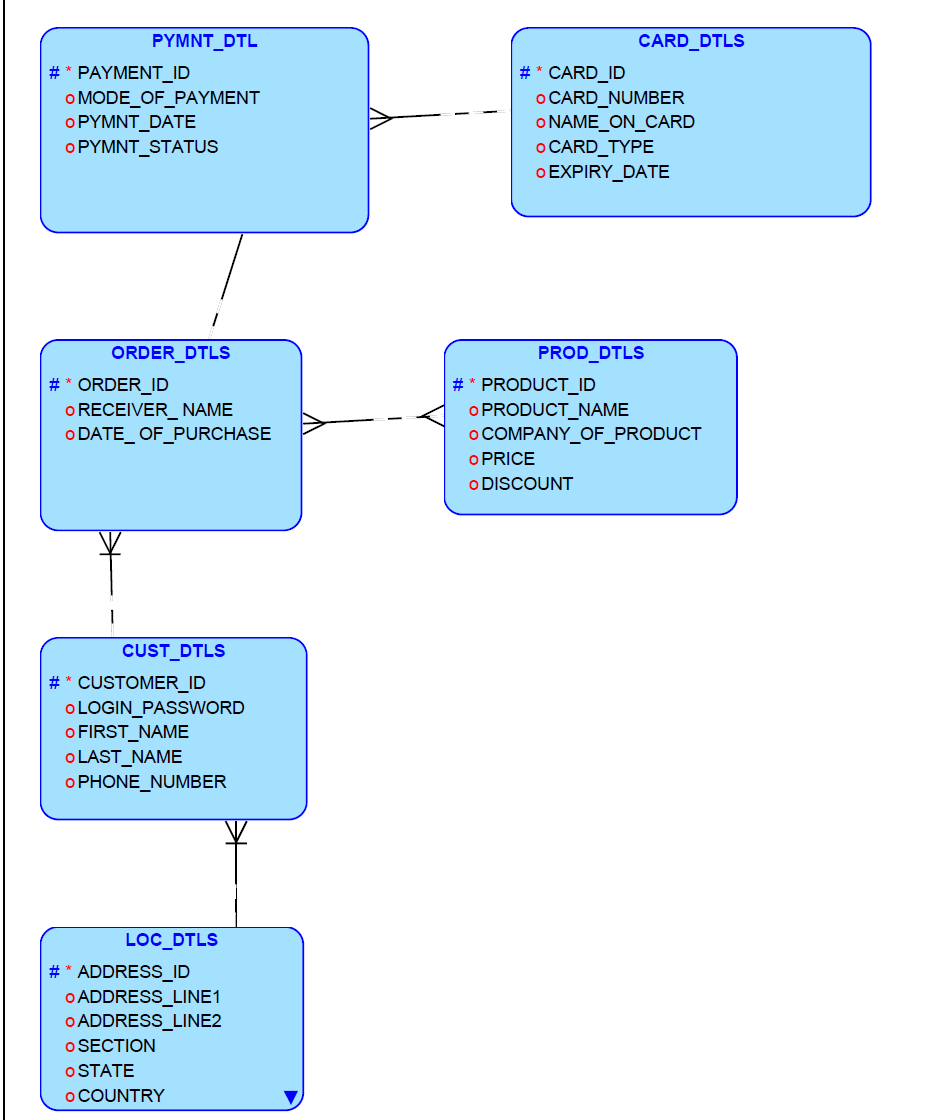
A product can have many categories

A customer can be associated with zero or one or many products.

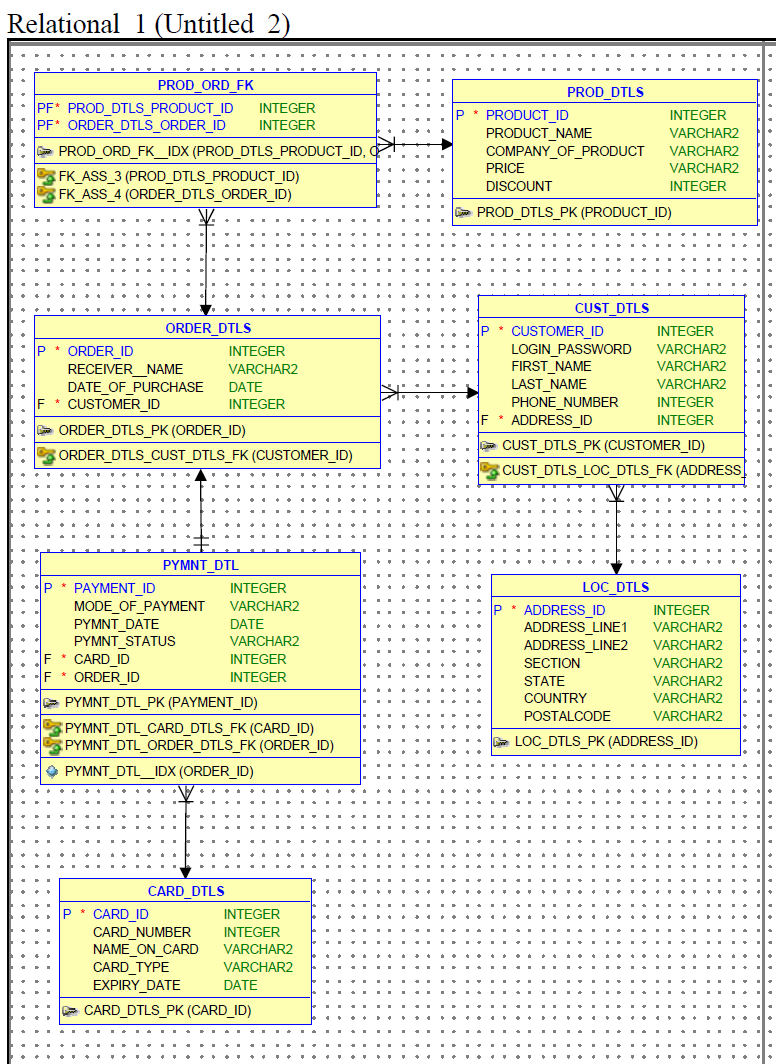
**EER DIAGRAM**:

Identifying relationship is when child object is dependent on parent object. Non identifying relationship is when primary key attributes of parent must not become primary attributes of child.

Logical Model:



The above ER model for E-shopping data management is given. The customer can place multiple orders to buy a product through online with payment mode like cash on delivery, debit or credit methods and where as model will store card details for further future references.



**Relational Database Schema:**

The entities, datatypes, constraints and relation definition are given:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **1.CUST\_DTL** |  |  |  |  |
| **Sno** | **Keys** | **Attribute Name** | **Data Type** | **Description** |
| 1 | PK | CUSTOMER\_ID | Integer |  |
| 2 | FK | ADDRESS\_ID | Integer |  |
| 3 |  | LOGIN\_PASSWORD | Varchar |  |
| 4 |  | FIRST\_NAME | Varchar |  |
| 5 |  | LAST\_NAME | Varchar |  |
| 6 |  | EMAIL\_ADDRESS | Varchar |  |
| 7 |  | PHONE\_NUMBER | Varchar |  |

2.PROD\_DLTS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sno. | Keys | Attribute Name | Data type | Description |
| 1 | PK | PRODUCT\_ID | Integer |  |
| 2 |  | PRODUCT\_NAME | Varchar |  |
| 3 |  | COMPANY\_OF\_PRODUCT | Varchar |  |
| 4 |  | PRICE | Decimal |  |
| 5 |  | DISCOUNT | Decimal |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 3.ORDR\_DLTS |  |  |  |  |
| Sno. | Keys | Attribute Name | Data Type | Description |
| 1 | PK | ORDER\_ID | Integer | Primary key for Order identification |
| 2 | FK | CUSTOMER\_ID | Char | Foreign key to Customer |
| 3 |  | RECEIVER\_NAME | Char | If order is to be sent to other address rather than to the customer. |
| 4 | FK | ADDRESS\_ID | Integer |  |
| 5 |  | DATE\_OF\_PURCHASE | Date |  |

4.PAYMENT\_DTLS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| Sno | Keys | Attribute Name | Data Type | Description |
| 1 | PK | PAYMENT\_ID | Integer |  |
| 2 |  | MODE\_OF\_PAYMENT | Varchar |  |
| 3 |  | PAYMENT\_DATE | Date |  |
| 4 |  | PAYMENT\_STATUS | Varchar |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 5.CARD\_DTLS |  |  |  |  |
| Sno | Keys | Attribute Name | Data Type | Description |
| 1 | PK | CARD\_ID | Integer |  |
| 2 | FK | CUSTOMER\_ID | Integer | Foreign key to Customer |
| 3 | FK | PAYMENT\_ID | Integer | Foreign key to Customer |
| 4 |  | NAME\_ON\_CARD | Varchar | Primary key for Customer Identification |
| 5 |  | CARD\_NUMBER | Varchar |  |
| 6 |  | CARD\_TYPE | Varchar | Master Card, Visa, Discover |
| 7 |  | CVV\_NUMBER | Integer | Number present on the back of the card for extra security |
| 8 |  | EXPIRY\_DATE | Date |  |

6.LOC\_DTLS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| Sno | Keys | Attribute Name | Data Type | Description |
| 1 | PK | ADDRESS\_ID | Integer |  |
| 2 |  | ADDRESS\_LINE1 | Varchar |  |
| 3 |  | ADDRESS\_LINE2 | Varchar |  |
| 4 |  | SECTION | Varchar |  |
| 5 |  | STATE | Varchar |  |
| 6 |  | COUNTRY | Varchar |  |
| 7 |  | POSTAL\_CODE | Integer |  |

1. **NORMALISATION**

The tables in database are designed for third normal form(3NF). The third normal form must satisty the requirement of first and second normalization i.e 1NF and 2NF. Most of the tables in databases have been designed to keep optimization. We use normalization to remove update, delete and insert anomalies.

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
| Before Normalization | | |
| Key | Attribute Names | Comments |
|  | CUSTOMER\_NAME |  |
|  | LOGIN\_PASSWORD |  |
|  | EMAIL\_ADDRESS |  |
|  | PHONE\_NUMBER |  |
|  | ADDRESS\_LINE1 |  |
|  | ADDRESS\_LINE2 |  |
|  | SECTION |  |
|  | STATE |  |
|  | COUNTRY |  |
|  | POSTAL\_CODE |  |
|  | RECEIVER\_NAME1 |  |
|  | DATE\_OF\_PURCHASE1 |  |
|  | MODE\_OF\_PAYMENT1 |  |
|  | PAYMENT\_DATE1 |  |
|  | PAYMENT\_STATUS1 |  |
|  | NAME\_ON\_CARD1 |  |
|  | CARD\_NUMBER1 |  |
|  | CARD\_TYPE1 |  |
|  | CVV\_NUMBER1 |  |
|  | EXPIRY\_DATE1 |  |
|  | RECEIVER\_NAME2 |  |
|  | DATE\_OF\_PURCHASE2 |  |
|  | MODE\_OF\_PAYMENT2 |  |
|  | PAYMENT\_DATE2 |  |
|  | PAYMENT\_STATUS2 |  |
|  | NAME\_ON\_CARD2 |  |
|  | CARD\_NUMBER2 |  |
|  | CARD\_TYPE2 |  |
|  | CVV\_NUMBER2 |  |
|  | EXPIRY\_DATE2 |  |
|  | PRODUCT\_NAME |  |
|  | COMPANY\_OF\_PRODUCT |  |
|  | PRICE |  |
|  | DISCOUNT |  |

1NF

By the 1NF there should be no multiple values against any field and create a separate table for each set of related data with unique primary key.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | |  | |  | |
|  | |  | |  | |
|  | |  | |  | |
| 1stNormal Form | | | | | |
| Key | | Attribute Name | | Comments | |
| PK | | CUSTOMER\_ID | |  | |
|  | | FIRST\_NAME | |  | |
|  | | SECOND\_NAME | |  | |
|  | | LOGIN\_PASSWORD | |  | |
|  | | EMAIL\_ADDRESS | |  | |
|  | | PHONE\_NUMBER | |  | |
|  | | ADDRESS\_LINE1 | |  | |
|  | | ADDRESS\_LINE2 | |  | |
|  | | SECTION | |  | |
|  | | STATE | |  | |
|  | | COUNTRY | |  | |
|  | | POSTAL\_CODE | |  | |
|  | |  | |  | |
|  | |  | |  | |
| PK | | CUSTOMER\_ID | |  | |
|  | | RECEIVER\_FIRST\_NAME | |  | |
|  | | RECEIVER\_LAST\_NAME | |  | |
|  | | DATE\_OF\_PURCHASE | |  | |
|  | | MODE\_OF\_PAYMENT | |  | |
|  | | PAYMENT\_DATE | |  | |
|  | | PAYMENT\_STATUS | |  | |
|  | | NAME\_ON\_CARD | |  | |
|  | | CARD\_NUMBER | |  | |
|  | | CARD\_TYPE | |  | |
|  | | CVV\_NUMBER | |  | |
|  | | EXPIRY\_DATE | |  | |
|  | | PRODUCT\_NAME | |  | |
|  | | COMPANY\_OF\_PRODUCT | |  | |
|  | | PRICE | |  | |
|  | | DISCOUNT | |  | |

2NF

For second normalization form the non-key attributes should be fully dependent on composite primary key, partial dependency is not allowed.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| 2nd Normal Form | | |  |
| Key | Attribute Name | Comments |  |
| PK | CUSTOMER\_ID |  |  |
|  | FIRST\_NAME |  |  |
|  | SECOND\_NAME |  |  |
|  | LOGIN\_PASSWORD |  |  |
|  | EMAIL\_ADDRESS |  |  |
|  | PHONE\_NUMBER |  |  |
|  | ADDRESS\_LINE1 |  |  |
|  | ADDRESS\_LINE2 |  |  |
|  | SECTION |  |  |
|  | STATE |  |  |
|  | COUNTRY |  |  |
|  | POSTAL\_CODE |  |  |
|  |  |  |  |
|  |  |  |  |
| PK | CUSTOMER\_ID |  |  |
|  | PAYMENT\_ID |  |  |
|  | RECEIVER\_FIRST\_NAME |  |  |
|  | RECEIVER\_LAST\_NAME |  |  |
|  | DATE\_OF\_PURCHASE |  |  |
|  | MODE\_OF\_PAYMENT |  |  |
|  | PAYMENT\_DATE |  |  |
|  | PAYMENT\_STATUS |  |  |
|  | NAME\_ON\_CARD |  |  |
|  | CARD\_NUMBER |  |  |
|  | CARD\_TYPE |  |  |
|  | CVV\_NUMBER |  |  |
|  | EXPIRY\_DATE |  |  |
|  | PRODUCT\_NAME |  |  |
|  | COMPANY\_OF\_PRODUCT |  |  |
|  | PRICE |  |  |
|  | DISCOUNT |  |  |

3NF

To achieve this normal form eliminate the non key values which are not fully functionally dependent on primary key of table.

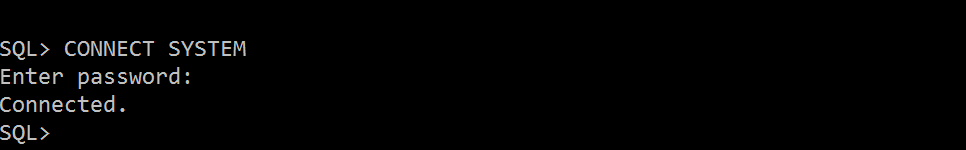
|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| 3rd Normal Form | | |  |
| Key | Attribute Name | Comments |  |
| PK | CUSTOMER\_ID |  |  |
|  | FIRST\_NAME |  |  |
|  | SECOND\_NAME |  |  |
|  | LOGIN\_PASSWORD |  |  |
|  | PHONE\_NUMBER |  |  |
|  | ADDRESS\_ID |  |  |
|  |  |  |  |
|  |  |  |  |
| FK | ADDRESS\_ID |  |  |
|  | ADDRESS\_LINE1 |  |  |
|  | ADDRESS\_LINE2 |  |  |
|  | SECTION |  |  |
|  | STATE |  |  |
|  | COUNTRY |  |  |
|  | POSTAL\_CODE |  |  |
|  |  |  |  |
| PK | ORDER\_ID |  |  |
| FK | CUSTOMER\_ID |  |  |
|  | RECEIVER\_FIRST\_NAME |  |  |
|  | RECEIVER\_LAST\_NAME |  |  |
|  | DATE\_OF\_PURCHASE |  |  |
|  |  |  |  |
|  |  |  |  |
| PK | PAYMENT\_ID |  |  |
| FK | CARD\_ID |  |  |
| FK | ORDER\_ID |  |  |
|  | MODE\_OF\_PAYMENT |  |  |
|  | PAYMENT\_DATE |  |  |
|  | PAYMENT\_STATUS |  |  |
|  |  |  |  |
| PK | CARD\_ID |  |  |
|  | NAME\_ON\_CARD |  |  |
|  | CARD\_NUMBER |  |  |
|  | CARD\_TYPE |  |  |
|  | CVV\_NUMBER |  |  |
|  | EXPIRY\_DATE |  |  |
|  |  |  |  |
| PK | PRODUCT\_ID |  |  |
|  | PRODUCT\_NAME |  |  |
|  | COMPANY\_OF\_PRODUCT |  |  |
|  | PRICE |  |  |
|  | DISCOUNT |  |  |
|  |  |  |  |
| PK | PRODUCT\_ID |  |  |
| PK | ORDER\_ID |  |  |

1. **PHYSICAL DATABASE DESIGN**

The SQL Queries for online shopping are

--connect to database as system

*CONNECT SYSTEM(ENTER PASSWORD)*



*CREATE TABLESPACE sample*

*DATAFILE 'sample.dbf'*

*SIZE 500M AUTOEXTEND ON*

*EXTENT MANAGEMENT LOCAL;*

*CREATE TEMPORARY TABLESPACE TEMP\_sample*

*TEMPFILE 'temp\_sample.dbf'*

*SIZE 200M AUTOEXTEND ON*

*EXTENT MANAGEMENT LOCAL;*

*CREATE USER sampleusr*

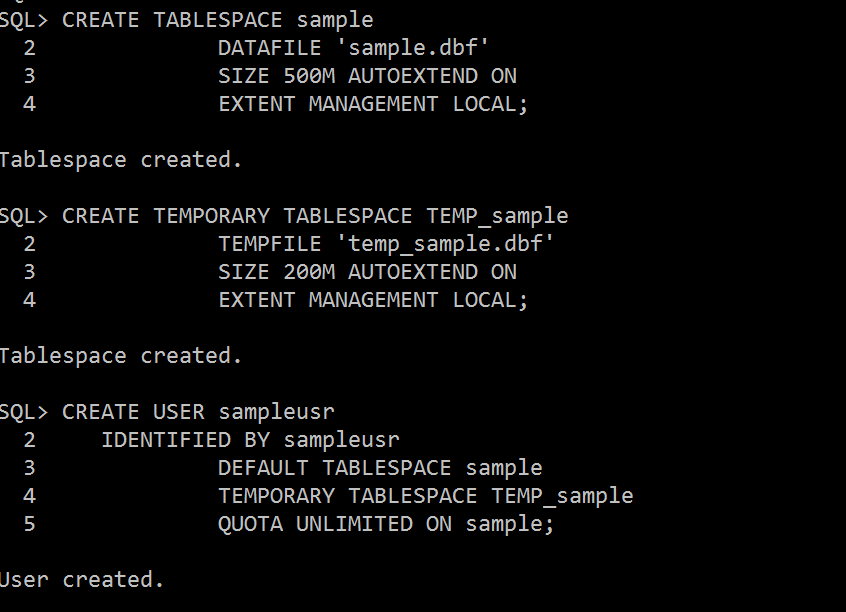
*IDENTIFIED BY sampleusr*

*DEFAULT TABLESPACE sample*

*TEMPORARY TABLESPACE TEMP\_sample*

*QUOTA UNLIMITED ON sample;*

*GRANT ALL PRIVILEGES TO sampleusr WITH ADMIN OPTION;*



*CREATE TABLE CARD\_DTLS*

*(*

*CARD\_ID INTEGER NOT NULL ,*

*CARD\_NUMBER INTEGER ,*

*NAME\_ON\_CARD VARCHAR2(20) ,*

*CARD\_TYPE VARCHAR2(20),*

*EXPIRY\_DATE DATE*

*) ;*

*ALTER TABLE CARD\_DTLS ADD CONSTRAINT CARD\_DTLS\_PK PRIMARY KEY ( CARD\_ID ) ;*

*CREATE TABLE CUST\_DTLS*

*(*

*CUSTOMER\_ID INTEGER NOT NULL ,*

*LOGIN\_PASSWORD VARCHAR2(100) ,*

*FIRST\_NAME VARCHAR2(100) ,*

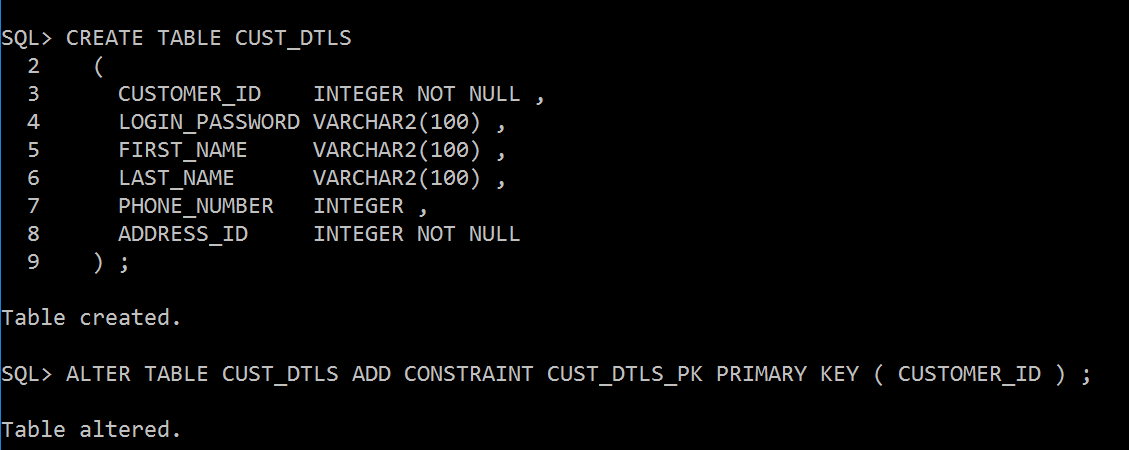
*LAST\_NAME VARCHAR2(100) ,*

*PHONE\_NUMBER INTEGER ,*

*ADDRESS\_ID INTEGER NOT NULL*

*) ;*

*ALTER TABLE CUST\_DTLS ADD CONSTRAINT CUST\_DTLS\_PK PRIMARY KEY ( CUSTOMER\_ID ) ;*



*CREATE TABLE LOC\_DTLS*

*(*

*ADDRESS\_ID INTEGER NOT NULL ,*

*ADDRESS\_LINE1 VARCHAR2(100) ,*

*ADDRESS\_LINE2 VARCHAR2(100) ,*

*SECTION VARCHAR2(100) ,*

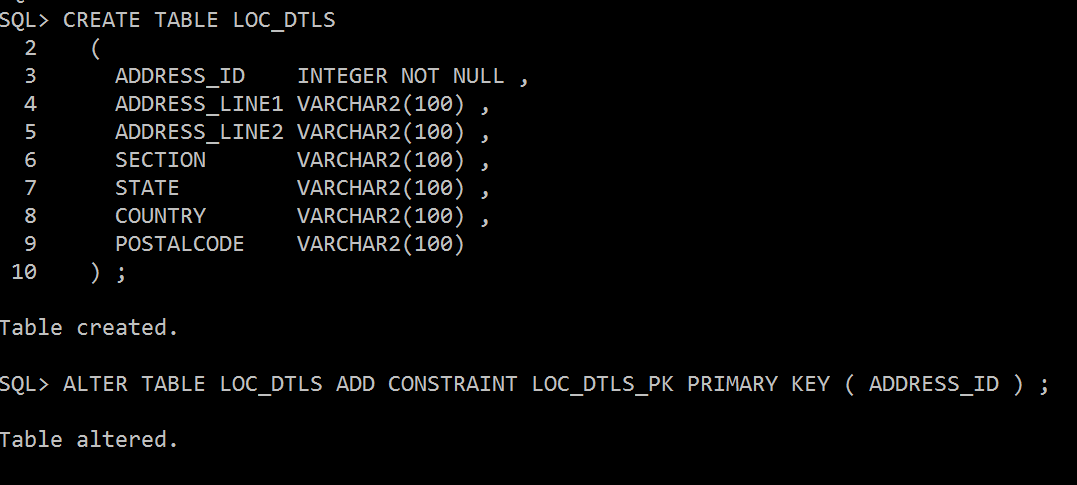
*STATE VARCHAR2(100) ,*

*COUNTRY VARCHAR2(100) ,*

*POSTALCODE VARCHAR2(100)*

*) ;*

*ALTER TABLE LOC\_DTLS ADD CONSTRAINT LOC\_DTLS\_PK PRIMARY KEY ( ADDRESS\_ID ) ;*



*CREATE TABLE ORDER\_DTLS*

*(*

*ORDER\_ID INTEGER NOT NULL ,*

*RECEIVER\_FIRST\_NAME VARCHAR2(100) ,*

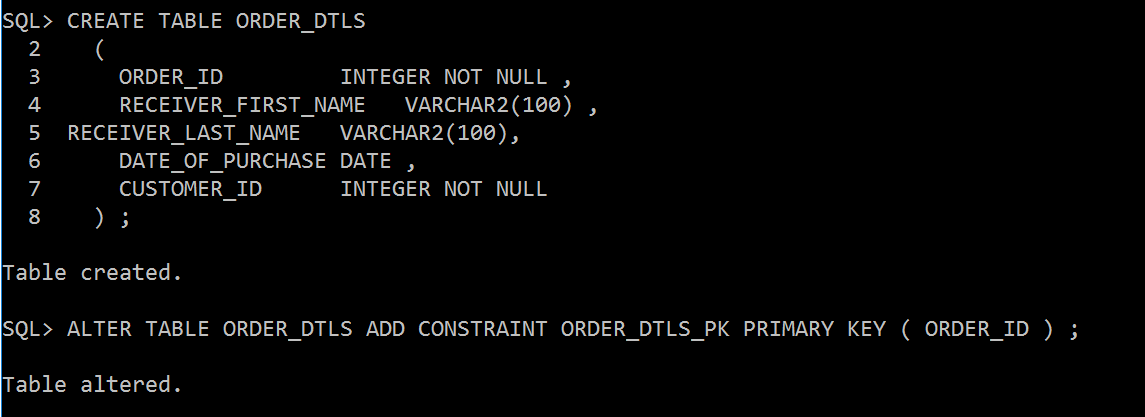
*RECEIVER\_LAST\_NAME VARCHAR2(100),*

*DATE\_OF\_PURCHASE DATE ,*

*CUSTOMER\_ID INTEGER NOT NULL*

*) ;*

*ALTER TABLE ORDER\_DTLS ADD CONSTRAINT ORDER\_DTLS\_PK PRIMARY KEY ( ORDER\_ID ) ;*



*CREATE TABLE PROD\_DTLS*

*(*

*PRODUCT\_ID INTEGER NOT NULL ,*

*PRODUCT\_NAME VARCHAR2(100) ,*

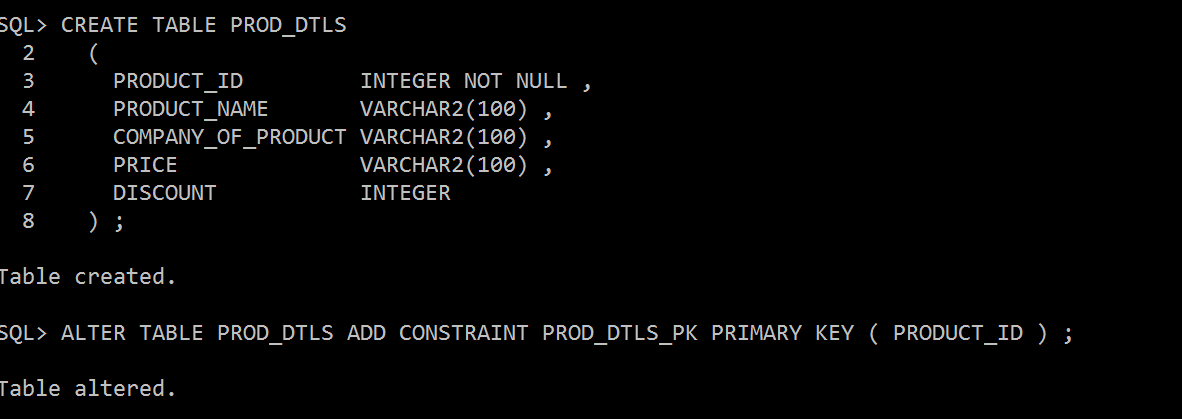
*COMPANY\_OF\_PRODUCT VARCHAR2(100) ,*

*PRICE VARCHAR2(100) ,*

*DISCOUNT INTEGER*

*) ;*

*ALTER TABLE PROD\_DTLS ADD CONSTRAINT PROD\_DTLS\_PK PRIMARY KEY ( PRODUCT\_ID ) ;*



*CREATE TABLE PROD\_ORD\_FK*

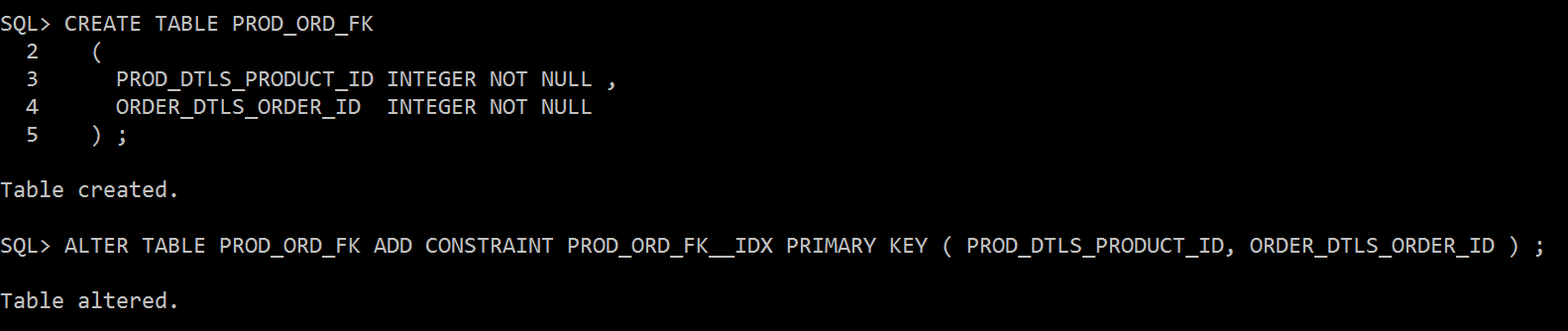
*(*

*PROD\_DTLS\_PRODUCT\_ID INTEGER NOT NULL ,*

*ORDER\_DTLS\_ORDER\_ID INTEGER NOT NULL*

*) ;*

*ALTER TABLE PROD\_ORD\_FK ADD CONSTRAINT PROD\_ORD\_FK\_\_IDX PRIMARY KEY ( PROD\_DTLS\_PRODUCT\_ID, ORDER\_DTLS\_ORDER\_ID ) ;*



*CREATE TABLE PYMNT\_DTL*

*(*

*PAYMENT\_ID INTEGER NOT NULL ,*

*MODE\_OF\_PAYMENT VARCHAR2(100) ,*

*PYMNT\_DATE DATE ,*

*PYMNT\_STATUS VARCHAR2(100) ,*

*CARD\_ID INTEGER NOT NULL ,*

*ORDER\_ID INTEGER NOT NULL*

*) ;*

*CREATE UNIQUE INDEX PYMNT\_DTL\_\_IDX ON PYMNT\_DTL*

*(*

*ORDER\_ID ASC*

*) ;*

*ALTER TABLE PYMNT\_DTL ADD CONSTRAINT PYMNT\_DTL\_PK PRIMARY KEY ( PAYMENT\_ID ) ;*

*ALTER TABLE CUST\_DTLS ADD CONSTRAINT CUST\_DTLS\_LOC\_DTLS\_FK FOREIGN KEY ( ADDRESS\_ID ) REFERENCES LOC\_DTLS ( ADDRESS\_ID ) ;*

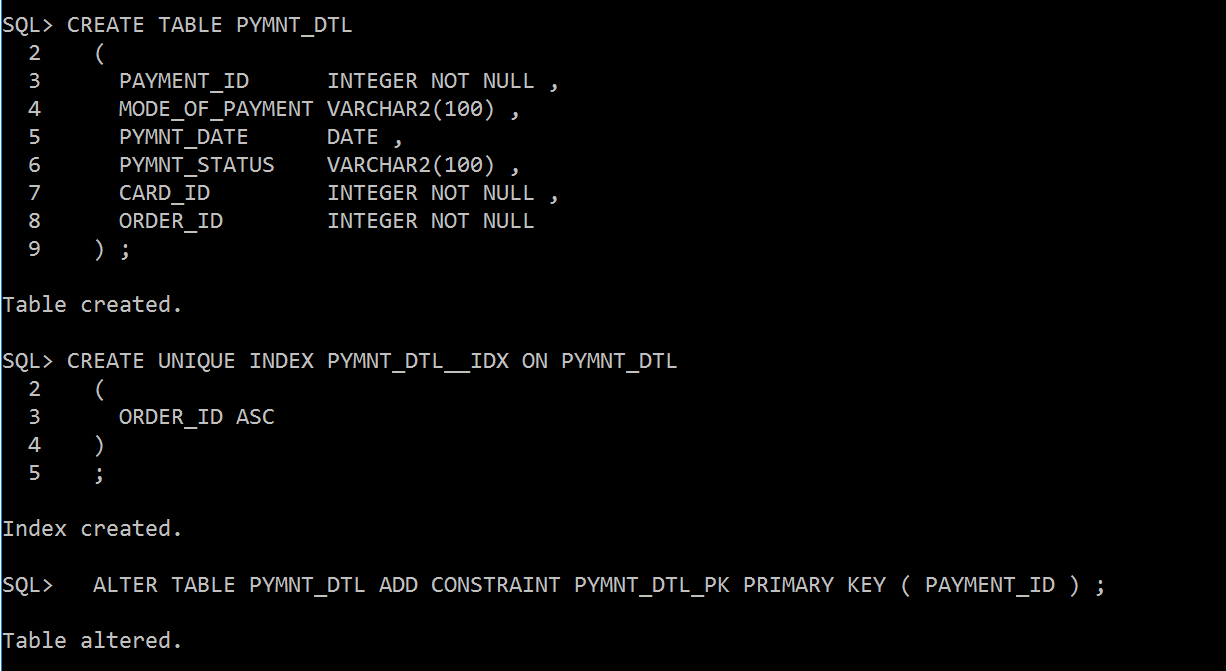
*ALTER TABLE PROD\_ORD\_FK ADD CONSTRAINT FK\_ASS\_3 FOREIGN KEY ( PROD\_DTLS\_PRODUCT\_ID ) REFERENCES PROD\_DTLS ( PRODUCT\_ID ) ;*

*ALTER TABLE PROD\_ORD\_FK ADD CONSTRAINT FK\_ASS\_4 FOREIGN KEY ( ORDER\_DTLS\_ORDER\_ID ) REFERENCES ORDER\_DTLS ( ORDER\_ID ) ;*

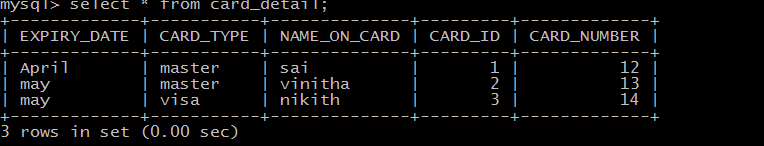
*ALTER TABLE ORDER\_DTLS ADD CONSTRAINT ORDER\_DTLS\_CUST\_DTLS\_FK FOREIGN KEY ( CUSTOMER\_ID ) REFERENCES CUST\_DTLS ( CUSTOMER\_ID ) ;*

*ALTER TABLE PYMNT\_DTL ADD CONSTRAINT PYMNT\_DTL\_CARD\_DTLS\_FK FOREIGN KEY ( CARD\_ID ) REFERENCES CARD\_DTLS ( CARD\_ID ) ;*

*ALTER TABLE PYMNT\_DTL ADD CONSTRAINT PYMNT\_DTL\_ORDER\_DTLS\_FK FOREIGN KEY ( ORDER\_ID ) REFERENCES ORDER\_DTLS ( ORDER\_ID ) ;*



*CARD TABLE;*

**

*INSERT INTO CARD DTLS(EXPIRY DATE,CARD\_TYPE,NAME\_ON\_CARD,CARD\_ID,CARD\_NUMBER)*

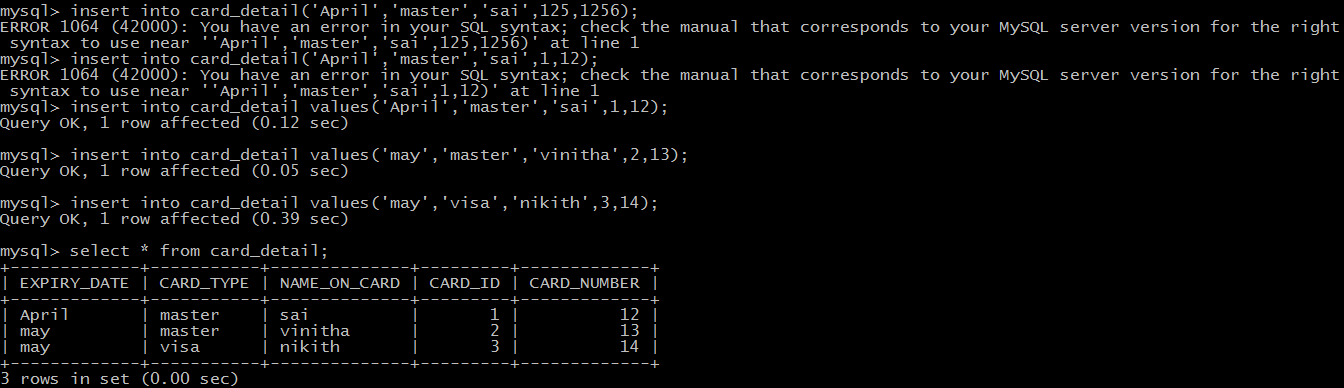
*VALUES(‘APRIL’,’MASTER’,’SAI’,1,12);*

*INSERT INTO CARD DTLS(EXPIRY DATE,CARD\_TYPE,NAME\_ON\_CARD,CARD\_ID,CARD\_NUMBER)*

*VALUES(‘MAY,’MASTER’,’VINITHA’,2,13);*

*INSERT INTO CARD DTLS(EXPIRY DATE,CARD\_TYPE,NAME\_ON\_CARD,CARD\_ID,CARD\_NUMBER)*

*VALUES(‘MAY’,’VISA’,’NIKITH’,3,14);*

**

*LOCATION DTLS*

*INSERT INTO LOC\_DTLS(ADDRESS ID, ADDRESS\_LINE1, ADDRESS\_LINE2,SECTION,STATE,COUNTR,POSTALCODE)*

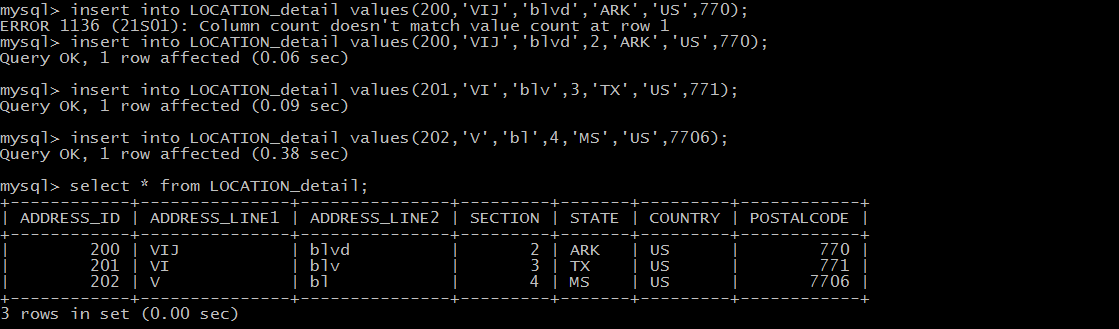
*VALUES(200,’VIJ’,’BLVD’,2,’ARK’,’US’,770);*

*INSERT INTO LOC\_DTLS(ADDRESS ID, ADDRESS\_LINE1, ADDRESS\_LINE2,SECTION,STATE,COUNTR,POSTALCODE)*

*VALUES(201,’VI’,’BLV’,3,TX,’US’,771);*

*INSERT INTO LOC\_DTLS(ADDRESS ID, ADDRESS\_LINE1, ADDRESS\_LINE2,SECTION,STATE,COUNTR,POSTALCODE)*

*VALUES(202,’V’,’BL’,4,MS,’US’,7706);*

**

*CUSTOMER\_DTLS*

*INSERT INTO CUST\_DTLS(CUSTOMER ID,FIRST NAME,LAST\_NAME,LOGIN\_PASSWORD,PHONE\_NUMBER,ADDRESS\_ID)*

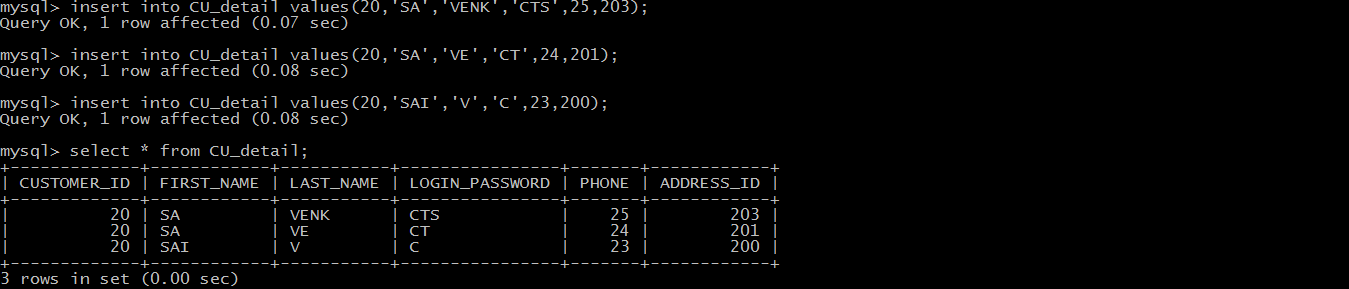
*VALUES(20,’SAI’,’V’,’C’,23,200);*

*INSERT INTO CUST\_DTLS(CUSTOMER ID,FIRST NAME,LAST\_NAME,LOGIN\_PASSWORD,PHONE\_NUMBER,ADDRESS\_ID)*

*VALUES(21,’SAI’,’VE’,’CT’,24,201);*

*INSERT INTO CUST\_DTLS(CUSTOMER ID,FIRST NAME,LAST\_NAME,LOGIN\_PASSWORD,PHONE\_NUMBER,ADDRESS\_ID)*

*VALUES(22,’SA’,’VENK’,’CTS’,25,203);*

**

*ORDER\_DTLS*

*INSERT INTO ORDER\_DTLS*

*(ORDER ID,CUSTOMER ID,RECEIVER\_FIRST\_NAME,RECEIVER\_LAST\_NAME,DATE\_OF\_PURCHASE)*

*VALUES(301,100,’SAI’,’KUMAR’,’APRIL’ );*

*INSERT INTO ORDER\_DTLS*

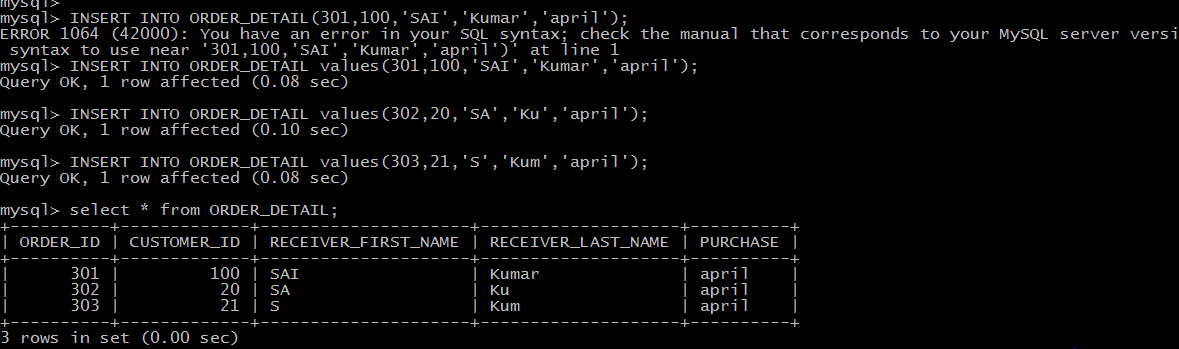
*(ORDER ID,CUSTOMER ID,RECEIVER\_FIRST\_NAME,RECEIVER\_LAST\_NAME,DATE\_OF\_PURCHASE)*

*VALUES(302,20,’SA’,’KU’,APRIL );*

*INSERT INTO ORDER\_DTLS*

*(ORDER ID,CUSTOMER ID,RECEIVER\_FIRST\_NAME,RECEIVER\_LAST\_NAME,DATE\_OF\_PURCHASE)*

*VALUES( 303,21,’S’,’KUM’,’APRIL’);*

**

*PYMNT\_DTL*

*INSERT INTO*

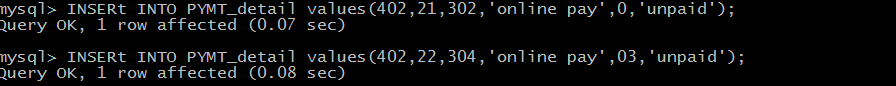
*PYMNT\_DTL(PAYMENT ID, CARD ID,ORDER\_ID,MODE\_OF\_PAYMENT,PYMNT\_DATE,PYMNT\_STATUS)*

*VALUES(402,21,302,’ONLINE PAY’,0,’UNPAID’ );*

*INSERT INTO*

*PYMNT\_DTL(PAYMENT ID, CARD ID,ORDER\_ID,MODE\_OF\_PAYMENT,PYMNT\_DATE,PYMNT\_STATUS)*

*VALUES(402,22,304,’ONLINE PAY’,03,’UNPAID’ );*

**

*PRODUCT DTLS:*

*INSERT INTO PROD\_DTLS(PRODUCT ID,PRODUCT NAME,COMPANY\_OF\_PRODUCT,PRICE,DISCOUNT)*

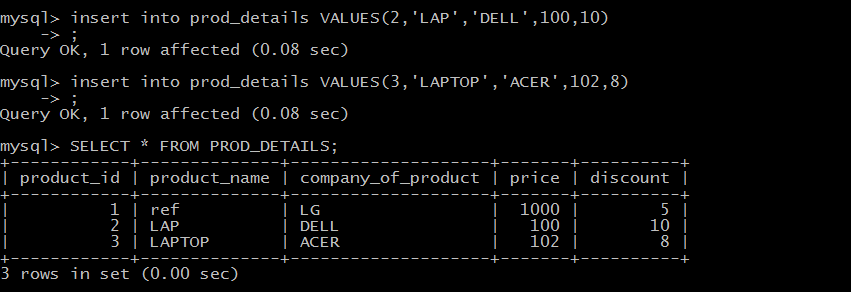
*VALUES(1’,REF’,’LG’,1000,5 );*

*INSERT INTO PROD\_DTLS(PRODUCT ID,PRODUCT NAME,COMPANY\_OF\_PRODUCT,PRICE,DISCOUNT)*

*VALUES(2,LAP,DELL,100,10 );*

*INSERT INTO PROD\_DTLS(PRODUCT ID,PRODUCT NAME,COMPANY\_OF\_PRODUCT,PRICE,DISCOUNT)*

*VALUES(3,’LAPTOP’,’ACER’,102,8 );*

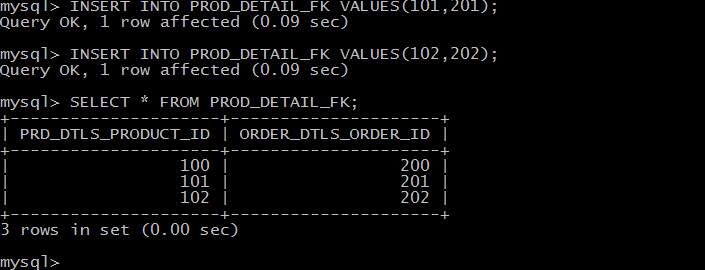
**

*PROD\_ORDER DTLS:*

*INSERT INTO PROD\_ORD\_FK (PROD\_DTLS\_PRODUCT\_ID,ORDER\_DTLS\_ORDER\_ID) VALUES (100,200);*

*INSERT INTO PROD\_ORD\_FK (PROD\_DTLS\_PRODUCT\_ID,ORDER\_DTLS\_ORDER\_ID) VALUES (101,201);*

*INSERT INTO PROD\_ORD\_FK (PROD\_DTLS\_PRODUCT\_ID,ORDER\_DTLS\_ORDER\_ID) VALUES (102,202);*

**

SELECT C.FIRST\_NAME,P.PRODUCT\_NAME,P.PRICE,PY.PYMNT\_DATE,PY.PYMNT\_STATUS FROM SAMPLEUSR.CUST\_DTLS C  
INNER JOIN LOC\_DTLS  L  
ON C.ADDRESS\_ID=L.ADDRESS\_ID  
INNER JOIN ORDER\_DTLS O    
ON C.CUSTOMER\_ID=O.CUSTOMER\_ID  
INNER JOIN PROD\_ORD\_FK PO  
ON PO.ORDER\_DTLS\_ORDER\_ID=O.ORDER\_ID  
INNER JOIN PROD\_DTLS P  
ON P.PRODUCT\_ID=PO.PROD\_DTLS\_PRODUCT\_ID  
INNER JOIN PYMNT\_DTL PY  
ON PY.ORDER\_ID=O.ORDER\_ID  
WHERE PYMNT\_STATUS='PAID'

SELECT C.FIRST\_NAME,P.PRODUCT\_NAME,P.PRICE,PY.PYMNT\_DATE,PY.PYMNT\_STATUS FROM SAMPLEUSR.CUST\_DTLS C  
INNER JOIN LOC\_DTLS  L  
ON C.ADDRESS\_ID=L.ADDRESS\_ID  
INNER JOIN ORDER\_DTLS O    
ON C.CUSTOMER\_ID=O.CUSTOMER\_ID  
INNER JOIN PROD\_ORD\_FK PO  
ON PO.ORDER\_DTLS\_ORDER\_ID=O.ORDER\_ID  
INNER JOIN PROD\_DTLS P  
ON P.PRODUCT\_ID=PO.PROD\_DTLS\_PRODUCT\_ID  
INNER JOIN PYMNT\_DTL PY  
ON PY.ORDER\_ID=O.ORDER\_ID  
WHERE PYMNT\_STATUS='NOT PAID'

1. **CONCLUSION**

The conclusion that can be drawn is therefore that Library Management database system has been designed for college library used to store data regarding books, members of library, update, delete and add or modify new records. It is flexible implementing in any library and moreover it can be modified into new technology. We can establish online library system by using database design at backend additionally.