Iowa State Bookstore

Project Parts 1, 2, & 3

MIS 320

Kaleb Faas

Table of Contents

Project Part 1:

Introduction

The Iowa State Bookstore is on the first floor of the Memorial Union. This store holds a variety of products including clothing items, textbooks, and class materials. The bookstore is owned by Iowa State University and allows them to sell merchandise tax-free. The ISU Bookstore prides itself on the variety of brand-name products it holds including Apple, Under Armor, Nike, and Champion. Their mission is to support student success and provide financial support for class materials. This store also holds a tech shop in the back that offers tech services and repairs for students' devices.

The bookstore has reached out to our team to improve its database system that will store an inventory of products, professor materials/courses, employees, suppliers, and customer data. To visualize the database more easily, we developed an entity relationship diagram (ERD). Currently, the Bookstore uses separate systems to track its inventory and customer data. Our goal is to pool all the store's data into one database. It will allow the employees to retrieve and analyze data more easily. Our goals of the system will be to track inventory needs and supplier information so that when stock is low employees will know when to order more. Most importantly this database will aid the bookstore to better help students while providing excellent customer service and improving the bookstore's overall efficiency.

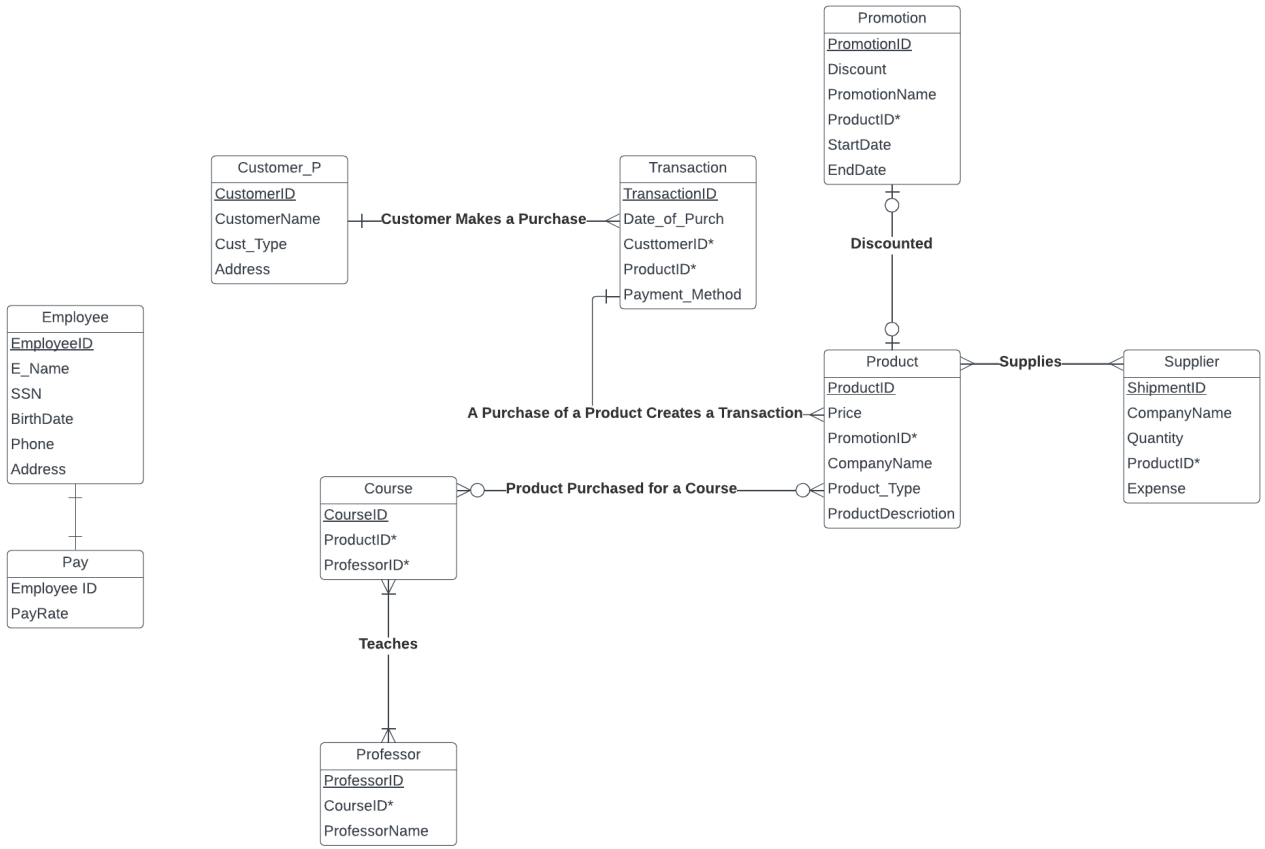
Limitations and Additions

- Limitations:
    - We didn't have access to the ISU Bookstore's actual data or database. Therefore, we were unable to accurately include the data that they collect when running the business.
    - We conducted this project from the point of view of a student/customer of the ISU Bookstore and not an employee. While it's not difficult to assume what tables and data are important to running the bookstore, there's no way to know for sure.
    - We did not have the experience/knowledge to actually implement our information and code to make it usable and functioning.
- Additions:
    - There are other tables that could be added to this project that would be relevant, however they weren't necessary for this project as it would become too complicated.
        - For example, we simplified our "Product" table by dividing it into three categories of "Apparel", "Technology", and "Textbook". However, there are obviously other categories of product that the bookstore sells that may not fall under these three.
    - Multiple products could be purchased in one transaction. To simplify our project and not make it too complex, we only allowed one product to be purchased per transaction.
        - For example, in a real-world situation someone may purchase more than one "Product" in a "Transaction". Therefore, this is an addition that could be made to this project.

Business Rules

- A **Customer** is identified by Customer ID with the attributes Customer Name, Customer Type, and Address. Customers can have one or many purchases and have a relationship with Transaction table.
- A **Professor** has the attributes Professor Name, Course ID, and Product ID. A Professor is identified with Professor ID. A Professor is linked to the Product table by Product ID, and Course by Course ID. A Professor teaches only one course and has only one Product required to be purchased for their course.
- An **Employee** is identified by their Employee ID, with the attributes Employee Name, SSN, Birth Date, and Phone. Employee has a relationship to the Pay table to denote how much they're paid.
- **Pay** is defined by the Employee ID and has the attribute PayRate, its related to the Employee table with a one-to-one relationship.
- **Product** is identified through their Product ID, with the attributes Price, Promotion ID, Company Name, Product Description, and Product Type. A Product can be supplied by only one supplier. A product can be purchased by multiple customers. And a Product has no Promotion or one Promotion.
- A **Promotion** is identified through the Promotion ID, with the attributes Product ID, the Promotion Name, Discount, Start Date, and End Date. Promotion has a relationship with the Product, there can be no Promotion tied to a Product or there can be only one.
- A **Supplier** is identified by the Shipment ID, with the attributes Company Name, Product ID, Expense, and Quantity. Suppliers have a relationship with the Product they supply to the bookstore, each Supplier can supply multiple Products but each Product is only supplied by one Supplier.
- A **Course** is identified by the Course ID, with the attributes Course Name, Product ID, Professor ID. A Course has a relationship with the Professor that teaches it and the Product that is required for students to buy to take the course.
- A **Transaction** is identified by the Transaction ID, with the attributes Date_of_Purch, Customer ID, Product ID, and Payment_Method. Transaction is related to the Customer table, there can be only one customer per transaction, but each customer can have multiple transactions. Transaction is also related to the Product table, a product can appear in multiple transactions, and a transaction only contains one product.

Entity Relationship Diagram


Promotion
PromotionID
Discount
PromotionName
ProductID*
StartDate
EndDate
Customer_P
CustomerID
CustomerName
Cust_Type
Address
Customer Makes a Purchase
Transaction
TransactionID
Date_of_Purch
CusttomerID*
ProductID*
Payment_Method
Discounted
Employee
EmployeeID
E_Name
SSN
BirthDate
Phone
Address
A Purchase of a Product Creates a Transaction
Product
ProductID
Price
PromotionID*
CompanyName
Product_Type
ProductDescriotion
Supplies
Supplier
ShipmentID
CompanyName
Quantity
ProductID*
Expense
Course
CourseID
ProductID*
ProfessorID*
Product Purchased for a Course
Pay
Employee ID
PayRate
Teaches
Professor
ProfessorID
CourseID*
ProfessorName

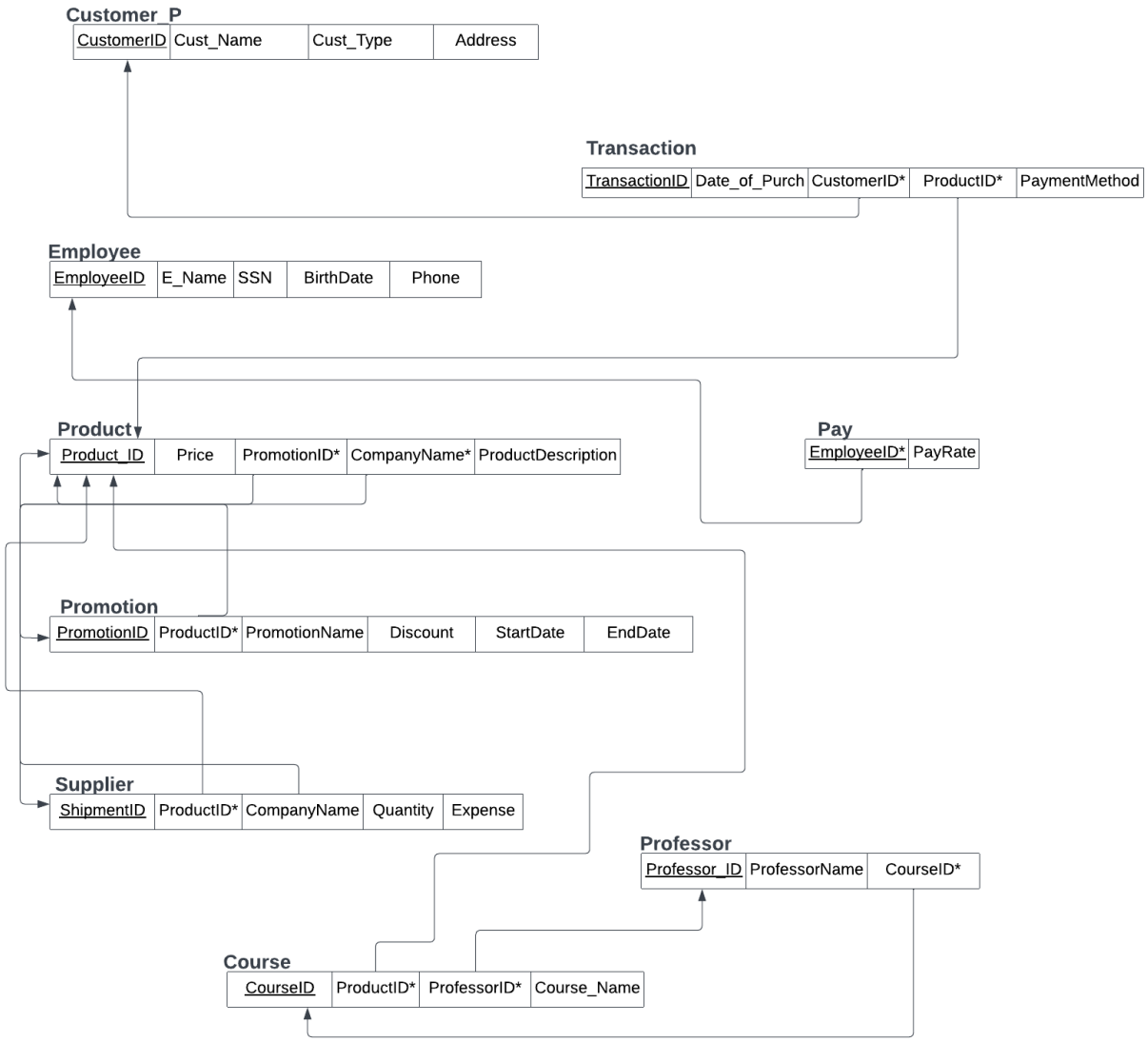
Project Part 2:

Data Types

- Transaction
  - TransactionID INT Not Null
  - Date_of_Purch DATE Not Null
  - CustomerID INT Not Null
  - ProductID INT
  - Payment_Method CHAR (11)
- Employee
  - EmployeeID INT Not Null
  - E_name CHAR (20) Not Null
  - SSN INT Not Null
  - BirthDate DATE Not Null
  - Phone CHAR (10) Not Null
- Pay
  - EmployeeID INT Not Null
  - Payrate NUMBER
- Customer_P
  - CustomerID INT Not Null
  - Cust_Name VARCHAR (30)
  - Cust_Type VARCHAR (30)
  - Address VARCHAR (30) Not Null
- Course
  - CourseID INT Not Null
  - Course_Name VARCHAR (30) Not Null
  - ProductID INT Not Null
  - ProfessorID INT Not Null
- Professor
  - ProfessorID INT Not Null
  - CourseID INT Not Null
  - ProfessorName CHAR (20) Not Null
  - ProductID CHAR (20) Not Null
- Product
  - ProductID INT Not Null
  - Price DECIMAL (7, 3)
  - PromotionID INT
  - CompanyName VARCHAR (20) Not Null

    - ProductDescription VARCHAR (50) Not Null
    - Product_Type VARCHAR (20) Not Null
- Promotion
    - PromotionID INT Not Null
    - PromotionName CHAR (20) Not Null
    - DiscountRate NUMBER
    - ProductID INT Not Null
    - StartDate DATE Not Null
    - EndDate DATE Not Null
- Supplier
    - ShipmentID INT Not Null
    - CompanyName VARCHAR (20) Not Null
    - Quantity INT
    - Expense DECIMAL (7, 3) Not Null
    - ProductID INT Not Null

Relational Schema (Normalization)



Design Mock-Up

https://iowastatebookstore.edu

IOWA STATE UNIVERSITY BOOK STORE

Employees

Promotions

Products

Transactions

Customers Suppliers

https://iowastatebookstore.edu

IOWA STATE UNIVERSITY BOOK STORE

< Back

Customers

CustomerID	Cust_name	Cust_type	Address
90001	Max Russo	Current Student	222 Lincoln Way
90001	Troy Bolton	Alumni	515 Welch Ave
90001	Sharpay Evans	Current Student	455 S. Duff Ave
90001	Stefan Salvatore	Non-Student	424 Beach Ave
90001	Debby Ryan	Alumni	623 Universiy Blvd

Customers Suppliers

https://iowastatebookstore.edu

IOWA STATE UNIVERSITY BOOK STORE

< Back

Suppliers

ShipmentID	CompanyName	Quantity	Expense	Product ID
40001	Shirts Co	100	$2,500.00	28401
40001	Uni Acessories	200	$2,000.00	38401
40001	Champion	150	$2,250.00	83027
40001	Paper Products	150	$750.00	29472
40001	Nike	200	$6,000.00	10384

Customers Suppliers

https://iowastatebookstore.edu

IOWA STATE UNIVERSITY BOOK STORE

< Back

Products

ProductID	Price	PromotionID	CompanyName	ProdDesc	ProductType
28401	$35.00	001	Shirts Co	Vintage Cy Shirt	Apparel
38401	$20.00	002	Uni Accessories	ISU Hat	Apparel
83027	$25.00	003	Champion	Cy Bucket Hat	Apparel
29472	$15.00	004	Paper Products	Cardinal & Gold Planners	Misc
10384	$40.00	005	Nike	Vintage Cy Sweatshirt	Apparel

Add New Make Changes

Professors Courses

https://iowastatebookstore.edu

IOWA STATE UNIVERSITY BOOK STORE

< Back

Courses

CourseID	Course_Name	ProductID	ProfessorID
60001	History 101	15221	500638478
60002	Anthropoogy 102	55461	510628488
60003	Economics 101	67431	520758466
60004	Accounting 285	88900	523448596
60005	Int Studies 201	99281	588098654

Professors Courses

https://iowastatebookstore.edu

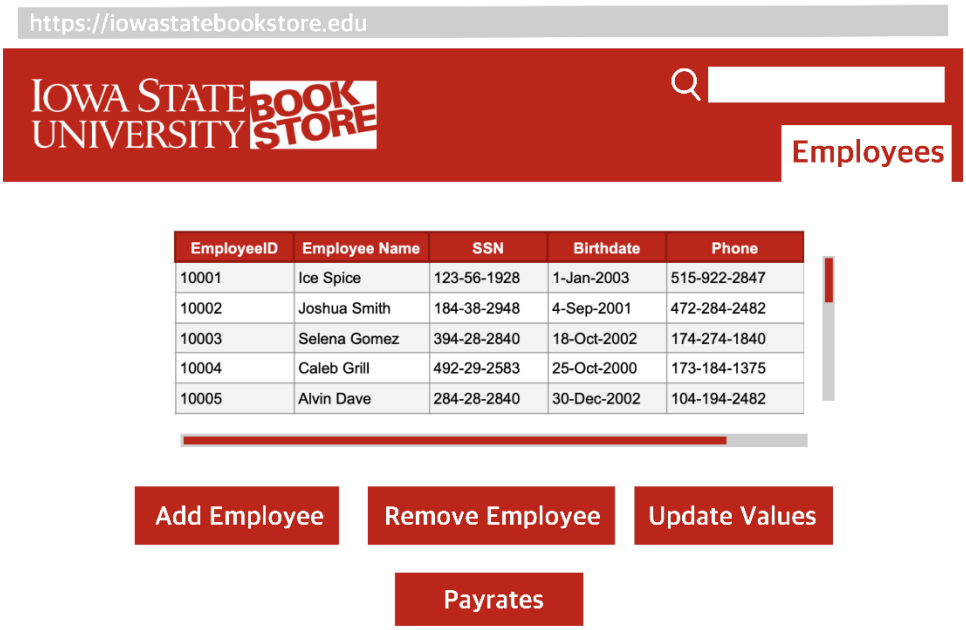
IOWA STATE UNIVERSITY BOOK STORE

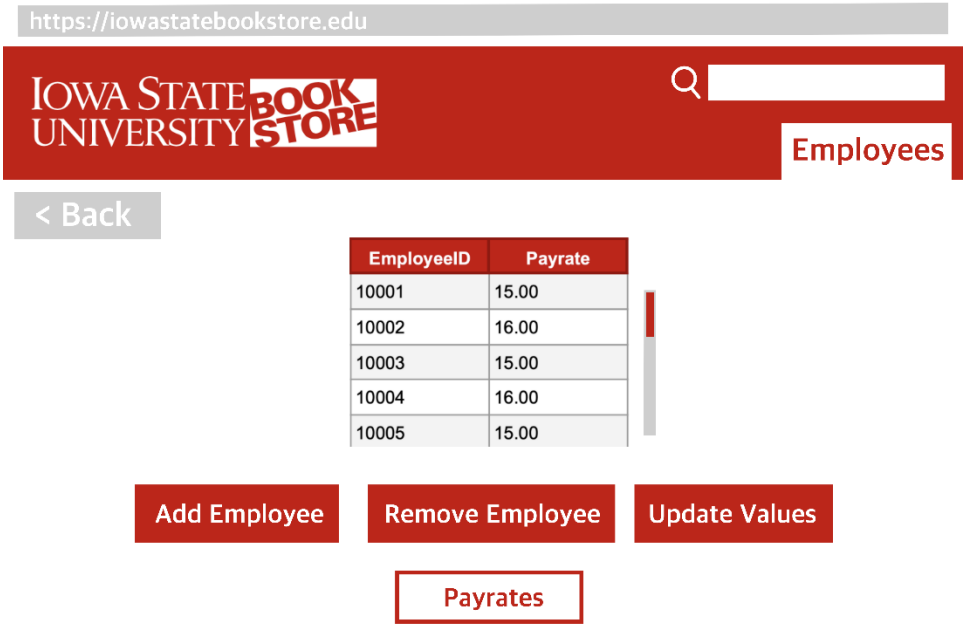
< Back

Professors

ProfessorID	CourseID	ProfessorName
500638478	60001	Joe Smith
510628488	60002	Maria Garcia
520758466	60003	James Johnson
523448596	60004	Dave Hernandez
588098654	60005	Michael Williams

Professors Courses


https://iowastatebookstore.edu
IOWA STATE UNIVERSITY BOOK STORE
Employees
EmployeeID | Employee Name | SSN | Birthdate | Phone
10001 | Ice Spice | 123-56-1928 | 1-Jan-2003 | 515-922-2847
10002 | Joshua Smith | 184-38-2948 | 4-Sep-2001 | 472-284-2482
10003 | Selena Gomez | 394-28-2840 | 18-Oct-2002 | 174-274-1840
10004 | Caleb Grill | 492-29-2583 | 25-Oct-2000 | 173-184-1375
10005 | Alvin Dave | 284-28-2840 | 30-Dec-2002 | 104-194-2482
Add Employee
Remove Employee
Update Values
Payrates


https://iowastatebookstore.edu
IOWA STATE UNIVERSITY BOOK STORE
Employees
< Back
EmployeeID | Payrate
10001 | 15.00
10002 | 16.00
10003 | 15.00
10004 | 16.00
10005 | 15.00
Add Employee
Remove Employee
Update Values
Payrates

https://iowastatebookstore.edu

IOWA STATE UNIVERSITY BOOK STORE

< Back

Promotions

PromotionID	PromtionName	Discount	ProductID	StartDate	EndDate
001	Back To School	25%	28401	24-Aug-2023	31-Aug-2023
002	Game Day	15%	38401	13-Sep-2023	15-Sep-2023
003	20 Off Bucket Hats	20%	83027	30-Sep-2023	7-Oct-2023
004	BOGO Planners	100%	29472	12-Oct-2023	31-Oct-2023
005	50 Off Ames Day	50%	10384	15-Feb-2023	15-Feb-2023

Add New Make Changes

SQL Script/Code

```
DROP TABLE PROMOTION;
CREATE TABLE PROMOTION
(
PromotionID INT PRIMARY KEY Not Null,
PromotionName CHAR (30) Not Null,
DiscountRate NUMBER,
ProductID INT Not Null,
StartDate DATE Not Null,
EndDate DATE Not Null
);
INSERT INTO PROMOTION VALUES('001','Back to School','.25','28401','24-Aug-2023','24-Aug-2023');
INSERT INTO PROMOTION VALUES('002','Game Day','.15','38401','13-Sep-2023','15-Sep-2023');
INSERT INTO PROMOTION VALUES('003','20 Off Bucket Hats','.20','83027','30-Sep-2023','7-Oct-2023');
INSERT INTO PROMOTION VALUES('004','BOGO Planners','1','29472','12-Oct-2023','31-Oct-2023');
INSERT INTO PROMOTION VALUES('005','50 Off Ames Day','.50','10384', '15-Feb-2023','15-Feb-2023');
INSERT INTO PROMOTION VALUES('006','Cardinal and Gold','.15','75543','6-Nov-2023','13-Nov-2023');
INSERT INTO PROMOTION VALUES('007','Homecoming','.35','18665','20-Oct-2023','25-Oct-2023');
INSERT INTO PROMOTION VALUES('008','Bundle Up','.15','23367','5-Dec-2023','10-Dec-2023');
INSERT INTO PROMOTION VALUES('009','Notebook Sale','.10','56478','11-Sep-2023','16-Sep-2023');
INSERT INTO PROMOTION VALUES('010','Cy Day','.15','33256','13-Jan-2023','18-Jan-2023');
INSERT INTO PROMOTION VALUES('011','Tech Day','.05','15567','9-Sep-2023','14-Sep-2023');
INSERT INTO PROMOTION VALUES('012','Game Day','.20','22567', '20-Feb-2023','20-Feb-2023');
INSERT INTO PROMOTION VALUES('013','Tech Day','.05','77659','1-Mar-2023','6-Mar-2023');
INSERT INTO PROMOTION VALUES('014','Spring Sale','.15','44568','15-Apr-2023','20-Apr-2023');
INSERT INTO PROMOTION VALUES('015','Graduation','.30','56745','10-May-2023','15-May-2023');

DROP TABLE PAY;
CREATE TABLE PAY
(
```

```
EmployeeID INT PRIMARY KEY,
Payrate NUMBER
);
INSERT INTO PAY VALUES('10001','15.00');
INSERT INTO PAY VALUES('10002','16.00');
INSERT INTO PAY VALUES('10003','15.00');
INSERT INTO PAY VALUES('10004','16.00');
INSERT INTO PAY VALUES('10005','15.00');
INSERT INTO PAY VALUES('10006','16.00');
INSERT INTO PAY VALUES('10007','17.00');
INSERT INTO PAY VALUES('10008','15.00');
INSERT INTO PAY VALUES('10009','17.00');
INSERT INTO PAY VALUES('10010','17.00');
INSERT INTO PAY VALUES('10011','16.00');
INSERT INTO PAY VALUES('10012','16.00');
INSERT INTO PAY VALUES('10013','15.00');
INSERT INTO PAY VALUES('10014','17.00');
INSERT INTO PAY VALUES('10015','16.00');

DROP TABLE EMPLOYEE;
CREATE TABLE EMPLOYEE
(
Employee_ID INT PRIMARY KEY,
E_name CHAR (20) NOT NULL,
SSN INT NOT NULL,
BirthDate DATE NOT NULL,
Phone CHAR (12) NOT NULL
);
INSERT INTO EMPLOYEE VALUES('10001','Ice Spice','123561928','1-Jan-2003','515-922-
2847');
INSERT INTO EMPLOYEE VALUES('10002','Joshua Smith','184382948','4-Sep-2001','472-
284-2482');
INSERT INTO EMPLOYEE VALUES('10003','Selena Gomez','394282840','18-Oct-2002','174-
274-1840');
INSERT INTO EMPLOYEE VALUES('10004','Caleb Grill','492292583','25-Oct-2000','173-
184-1375');
INSERT INTO EMPLOYEE VALUES('10005','Alvin Dave','284282840','30-Dec-2002','104-
194-2482');
INSERT INTO EMPLOYEE VALUES('10006','Kaylin Klomhaus','325662784','13-Mar-
2002','156-234-3567');
INSERT INTO EMPLOYEE VALUES('10007','Annie Chu','887286534','31-Oct-2001','178-
434-2678');
INSERT INTO EMPLOYEE VALUES('10008','Ava Feeney','345677890','3-Apr-2002','657-
889-7745');
INSERT INTO EMPLOYEE VALUES('10009','Sophia Damani-Ladha','135767480','3-Aug-
2002','515-178-3256');
```

```
INSERT INTO EMPLOYEE VALUES('10010','Kaleb Faas','546282034','12-Feb-2003','789-432-2322');
INSERT INTO EMPLOYEE VALUES ('10011','Cy Clone','778902212','7-Nov-2000','515-789-8843');
INSERT INTO EMPLOYEE VALUES ('10012','Hannah Montana','321874412','3-Jun-2002','670-854-9906');
INSERT INTO EMPLOYEE VALUES ('10013','John Smith','346121143','18-Jul-2001','889-221-3336');
INSERT INTO EMPLOYEE VALUES ('10014','Pedro Pascal','333695678','2-Apr-2000','515-432-4467');
INSERT INTO EMPLOYEE VALUES ('10015','Peeta Mellark','113765532','12-Mar-2002','414-778-0909');

DROP TABLE COURSE;
CREATE TABLE COURSE
(
COURSEID INT PRIMARY KEY,
COURSE_NAME VARCHAR (50),
PRODUCT_ID INT,
PROFESSORID INT
);
INSERT INTO COURSE VALUES('60001','History 101','15221','500638478');
INSERT INTO COURSE VALUES('60002','Anthropology 102','55461','510628488');
INSERT INTO COURSE VALUES('60003','Economics 101','67431','520758466');
INSERT INTO COURSE VALUES('60004','Accounting 285','88900','523448596');
INSERT INTO COURSE VALUES('60005','International Studies 201','99281','588098654');
INSERT INTO COURSE VALUES('60006','Math 150','88703','521333898');
INSERT INTO COURSE VALUES('60007','English 250','22190','576999785');
INSERT INTO COURSE VALUES('60008','Finance 301','66671','584235888');
INSERT INTO COURSE VALUES('60009','Management 372','57321','540551321');
INSERT INTO COURSE VALUES('60010','Statistics 226','83333','591555763');
INSERT INTO COURSE VALUES('60011','Geology 101','33321','531990002');
INSERT INTO COURSE VALUES('60012','Management 371','22100','511911003');
INSERT INTO COURSE VALUES('60013','Biology 101','00178','560932887');
INSERT INTO COURSE VALUES('60014','Marketing 340','11230','571022310');
INSERT INTO COURSE VALUES('60015','Management Information Systems 301','87611','501477909');

DROP TABLE PRODUCT;
CREATE TABLE PRODUCT
(
ProductID INT PRIMARY KEY NOT NULL,
Price DECIMAL (7, 3),
PromotionID INT,
CompanyName VARCHAR (30) NOT NULL,
ProductDescription VARCHAR (50) NOT NULL,
```

```
Product_Type VARCHAR (20) NOT NULL
);
 INSERT INTO PRODUCT VALUES('28401','35.00','001','Shirts Co.','Vintage Cy
Shirt','Apparel');
INSERT INTO PRODUCT VALUES('38401','20.00','002','Uni Accessories','ISU Hat','Apparel');
INSERT INTO PRODUCT VALUES('83027','25.00','003','Champion', 'Cy Bucket
Hat','Apparel');
INSERT INTO PRODUCT VALUES('29472','15.00','004','Paper Products','Cardinal & Gold
Planners','Miscellaneous');
INSERT INTO PRODUCT VALUES('10384','40.00','005','Nike','Vintage Cy
Sweatshirt','Apparel');
INSERT INTO PRODUCT VALUES('56745','35.00','015','Shirts Co. ','Alumni T-
Shirt','Apparel');
INSERT INTO PRODUCT VALUES('75543','50.00','006','Nike','Red Cy Sweatshirt','Apparel');
INSERT INTO PRODUCT VALUES('18665','45.00','007','Champion','Cy Sweatshirt','Apparel');
INSERT INTO PRODUCT VALUES('23367','25.00','008','Under Armor','ISU
Beanie','Apparel');
INSERT INTO PRODUCT VALUES('56478','15.00','009','Paper products','Cy
Notebook','Miscellaneous');
INSERT INTO PRODUCT VALUES('33256','20.00','010','Uni Accessories','Vintage Cy
Mug','Miscellaneous');
INSERT INTO PRODUCT VALUES('15567','1300.00','011','Apple','MacBook','Technology');
INSERT INTO PRODUCT VALUES('22567','40.00','012','Nike','Vintage Cy Basketball
Sweatshirt','Apparel');
INSERT INTO PRODUCT VALUES('77890','190.00','','Apple','Airpods','Technology');
INSERT INTO PRODUCT VALUES('44568','35.00','014','Shirts Co.','Cy T-Shirt','Apparel');
INSERT INTO PRODUCT VALUES('15221','80.00','','McGraw Hill','History of
Europe','Textbook');
INSERT INTO PRODUCT VALUES('67431','76.00','','McGraw Hill', 'Economics
101','Textbook');
INSERT INTO PRODUCT VALUES('88900','54.00','','Immediate Access','Managerial
Accounting','Textbook');
INSERT INTO PRODUCT VALUES('77659','1000.00','013','Dell','Laptop','Technology');
INSERT INTO PRODUCT VALUES('55461','94.00','','Immediate Access','Introduction to
Anthropology','Textbook');

DROP TABLE CUSTOMER_P;
CREATE TABLE CUSTOMER_P
(
CustomerID INT PRIMARY KEY,
Cust_name VARCHAR (30),
Cust_Type VARCHAR (30),
Address VARCHAR (30) NOT NULL
);
INSERT INTO CUSTOMER_P VALUES('90001','Max Russo','Current Student','222 Lincoln
Way');
```

```
INSERT INTO CUSTOMER_P VALUES('90002','Troy Bolton','Alumni','515 Welch Ave');
INSERT INTO CUSTOMER_P VALUES('90003','Sharpay Evans','Current Student','455 S. Duff
Ave.');
INSERT INTO CUSTOMER_P VALUES('90004','Stefan Salvatore','Non-Student','424 Beach
Ave.');
INSERT INTO CUSTOMER_P VALUES('90005','Debby Ryan','Alumni','623 University
Blvd.');
INSERT INTO CUSTOMER_P VALUES('90006','Meredith Grey','Alumni','274 Lincoln Way');
INSERT INTO CUSTOMER_P VALUES('90007','Elaine Benes','Non-Student','345 West St.');
INSERT INTO CUSTOMER_P VALUES('90008','Malibu Barbie','Current Student','678 Hyland
Ave.');
INSERT INTO CUSTOMER_P VALUES('90009','Alison DiLaurentis','Non-Student','213
Sheldon Ave.');
INSERT INTO CUSTOMER_P VALUES('90010','Penn Badgley','Alumni','778 Stanton Ave.');
INSERT INTO CUSTOMER_P VALUES('90011','Percy Jackson','Current Student','425 Welch
Ave.');
INSERT INTO CUSTOMER_P VALUES('90012','Drew Barrymore','Alumni','764 Sheldon
Ave.');
INSERT INTO CUSTOMER_P VALUES('90013','Florence Pugh','Alumni','223 Welch Ave.');
INSERT INTO CUSTOMER_P VALUES('90014','Andrew Garfield','Current Student','123
Stanton Ave.');
INSERT INTO CUSTOMER_P VALUES('90015','Dylan OBrien','Current Student','425 Welch
Ave.');

DROP TABLE TRANSACTION;
CREATE TABLE TRANSACTION
(
   TransactionID INT PRIMARY KEY,
   Date_of_Purch DATE,
   CustomerID INT,
   ProductID INT,
   PaymentMethod CHAR (11)
);
INSERT INTO TRANSACTION VALUES('50001','15-Feb-2023','90001','28401','U-Bill');
INSERT INTO TRANSACTION VALUES('50002','17-Jan-2023','90002','38401','Credit Card');
INSERT INTO TRANSACTION VALUES('50003','6-Mar-2023','90003','29472','U-Bill');
INSERT INTO TRANSACTION VALUES('50004','3-Mar-2023','90004','10384','Debit Card');
INSERT INTO TRANSACTION VALUES('50005','17-Feb-2023','90005','83027','Credit Card');
INSERT INTO TRANSACTION VALUES('50006','10-Jan-2023','90006','56745','Debit Card');
INSERT INTO TRANSACTION VALUES('50007','10-Jan-2023','90007','75543','Cash');
INSERT INTO TRANSACTION VALUES('50008','6-Mar-2023','90008','18665','U-Bill');
INSERT INTO TRANSACTION VALUES('50009','6-Mar-2023','90009','23367','Cash');
INSERT INTO TRANSACTION VALUES('50010','15-Feb-2023','90010','56478','Debit Card');
INSERT INTO TRANSACTION VALUES('50011','8-Apr-2023','90011','33256','U-Bill');
INSERT INTO TRANSACTION VALUES('50012','8-Apr-2023','90012','15567','Credit Card');
INSERT INTO TRANSACTION VALUES('50013','16-Apr-2023','90013','22567','Cash');
```

```
INSERT INTO TRANSACTION VALUES('50014','16-Apr-2023','90014','77890','U-Bill');
INSERT INTO TRANSACTION VALUES('50015','20-Apr-2023','90015','44568','U-Bill');
INSERT INTO TRANSACTION VALUES('50016','24-Apr-2023','90001','15221','U-Bill');
INSERT INTO TRANSACTION VALUES('50017','29-Apr-2023','90003','67431','U-Bill');
INSERT INTO TRANSACTION VALUES('50018','30-Apr-2023','90011','88900','U-Bill');
INSERT INTO TRANSACTION VALUES('50019','1-May-2023','90013','77659','Credit Card');
INSERT INTO TRANSACTION VALUES('50020','5-May-2023','90008','55461','U-Bill');

DROP TABLE PROFESSOR;
CREATE TABLE PROFESSOR
(
ProfessorID INT PRIMARY KEY,
CourseID INT NOT NULL,
ProfessorName CHAR (30) NOT NULL
);
INSERT INTO PROFESSOR VALUES('500638478','60001','Jane Smith');
INSERT INTO PROFESSOR VALUES('510628488','60002','Maria Garcia');
INSERT INTO PROFESSOR VALUES('520758466','60003','James Johnson');
INSERT INTO PROFESSOR VALUES('523448596','60004','Dave Hernandez');
INSERT INTO PROFESSOR VALUES('588098654','60005','Michael Williams');
INSERT INTO PROFESSOR VALUES('521333898','60006','Brandon Miller');
INSERT INTO PROFESSOR VALUES('576999785','60007','Zach Davis');
INSERT INTO PROFESSOR VALUES('584235888','60008','Jennifer Jones');
INSERT INTO PROFESSOR VALUES('540551321','60009','Sue Anderson');
INSERT INTO PROFESSOR VALUES('591555763','60010','Amy Young');
INSERT INTO PROFESSOR VALUES('531990002','60011','Emily Nguyen');
INSERT INTO PROFESSOR VALUES('511911003','60012','Jacob Hill');
INSERT INTO PROFESSOR VALUES('560932887','60013','Anna Flores');
INSERT INTO PROFESSOR VALUES('571022310','60014','Alex Mitchell');
INSERT INTO PROFESSOR VALUES('501477909','60015','Ben Roberts');

DROP TABLE SUPPLIER;
CREATE TABLE SUPPLIER
(
ShipmentID INT PRIMARY KEY Not Null,
CompanyName VARCHAR (20) Not Null,
Quantity INT,
Expense DECIMAL (9, 3) Not Null,
ProductID INT Not Null
);
INSERT INTO SUPPLIER VALUES('40001','Shirts Co.','100','2500.00','28401');
INSERT INTO SUPPLIER VALUES('40002','Uni Accessories','200','2000.00','38401');
INSERT INTO SUPPLIER VALUES('40003','Champion','150','2250.00','83027');
INSERT INTO SUPPLIER VALUES('40004','Paper Products','150','750.00','29472');
INSERT INTO SUPPLIER VALUES('40005','Nike','200','6000.00','10384');
INSERT INTO SUPPLIER VALUES('40006','Apple','100','110000.00','15567');
```

```
INSERT INTO SUPPLIER VALUES('40007','Shirts Co.','150','3750.00','56745');
INSERT INTO SUPPLIER VALUES('40008','Nike','100','4000.00','75543');
INSERT INTO SUPPLIER VALUES('40009','Champion','100','3500.00','18665');
INSERT INTO SUPPLIER VALUES('40010','Under Armor','100','1500.00','23367');
INSERT INTO SUPPLIER VALUES('40011','Paper Products','150','750.00','56478');
INSERT INTO SUPPLIER VALUES('40012','Uni Accessories','100','1000.00','33256');
INSERT INTO SUPPLIER VALUES('40013','Nike','100','3000.00','22567');
INSERT INTO SUPPLIER VALUES('40014','Apple','50','9000.00','77890');
INSERT INTO SUPPLIER VALUES('40015','Shirts Co.','100','2500.00','44568');
INSERT INTO SUPPLIER VALUES('40016','McGraw Hill','200','1400.00','15221');
INSERT INTO SUPPLIER VALUES('40017','McGraw Hill','200','13200.00','67431');
INSERT INTO SUPPLIER VALUES('40018','Immediate Access','200','8800.00','88900');
INSERT INTO SUPPLIER VALUES('40019','Immediate Access','200','16800.00','55461');
INSERT INTO SUPPLIER VALUES('40020','Dell','100','90000.00','77659');
```

Functions and Descriptions

1. Give a list of all the Customer information within the database.

```
SELECT *
FROM CUSTOMER_P;
```

CUSTOMERID	CUST_NAME	CUST_TYPE	ADDRESS
90001	Max Russo	Current Student	222 Lincoln Way
90002	Troy Bolton	Alumni	515 Welch Ave
90003	Sharpay Evans	Current Student	455 S. Duff Ave.
90004	Stefan Salvatore	Non-Student	424 Beach Ave.
90005	Debby Ryan	Alumni	623 University Blvd.
90006	Meredith Grey	Alumni	274 Lincoln Way
90007	Elaine Benes	Non-Student	345 West St.
90008	Malibu Barbie	Current Student	678 Hyland Ave.
90009	Alison DiLaurentis	Non-Student	213 Sheldon Ave.

2. Retrieve the Names of the Current Students who have purchased Apparel.

```
SELECT CUST_NAME
FROM CUSTOMER_P,PRODUCT,TRANSACTION
WHERE CUSTOMER_P.CUSTOMERID=TRANSACTION.CUSTOMERID
AND TRANSACTION.PRODUCTID=PRODUCT.PRODUCTID
AND CUST_TYPE = 'Current Student'
AND  PRODUCT.PRODUCT_TYPE = 'Apparel';
```

CUST_NAME
Max Russo
Malibu Barbie
Dylan OBrien

3. Retrieve Transaction IDs that occurred on April 16th 2023.

```
SELECT TRANSACTIONID
FROM TRANSACTION
WHERE DATE_OF_PURCH = ‘16-Apr-2023' ;
```

TRANSACTIONID
50014
50013

4. Retrieve a list of all Suppliers and their Products

```
SELECT CompanyName, ProductID
FROM SUPPLIER ;
```

COMPANYNAME	PRODUCTID
Shirts Co.	28401
Uni Accessories	38401
Champion	83027
Paper Products	29472
Nike	10384
Apple	15567
Shirts Co.	56745
Nike	75543
Champion	18665
Under Armor	23367

5. Retrieve all Transaction IDs from the Transaction table for Customers that are the Customer Type "Alumni"

```
SELECT TRANSACTIONID
FROM TRANSACTION,CUSTOMER_P
WHERE TRANSACTION.CUSTOMERID = CUSTOMER_P.CUSTOMERID
AND CUST_TYPE='Alumni';
```

TRANSACTIONID
50002
50005
50006
50010
50012
50013
50019

6. Retrieve the Name of all Employees with a Pay Rate of "15.00".

```
SELECT E_NAME
FROM EMPLOYEE,PAY
WHERE EMPLOYEE.EMPLOYEEID = PAY.EMPLOYEEID
AND PAYRATE='15.00';
```

E_NAME
Ice Spice
Selena Gomez
Alvin Dave
Ava Feeney
John Smith

7. Retrieve the Professor and Course Name for the Product with the ProductID "88900".

```
SELECT PROFESSORNAME,COURSE_NAME
FROM PROFESSOR,COURSE,PRODUCT
WHERE PRODUCT.PRODUCTID=COURSE.PRODUCT_ID
AND COURSE.COURSEID=PROFESSOR.COURSEID
AND COURSE.PRODUCT_ID = '88900';
```

PROFESSORNAME	COURSE_NAME
Dave Hernandez	Accounting 285

1 rows returned in 0.01 seconds Download

8. Retrieves Employee ID, Name and Pay Rate.

```
SELECT EMPLOYEE.EMPLOYEEID, EMPLOYEE.E_NAME, PAY.PAYRATE
```

```
FROM EMPLOYEE
INNER JOIN PAY ON EMPLOYEE.EMPLOYEEID = PAY.EMPLOYEEID;
```

EMPLOYEEID	E_NAME	PAYRATE
10001	Ice Spice	15
10002	Joshua Smith	16
10003	Selena Gomez	15
10004	Caleb Grill	16
10005	Alvin Dave	15
10006	Kaylin Klomhaus	16
10007	Annie Chu	17
10008	Ava Feeney	15
10009	Sophia Damani-Ladha	17

9. Retrieve Professor information and Textbook relating to their Course sorted by Professor ID.

```
SELECT P.PROFESSORID, PROFESSORNAME, C.COURSEID, COURSE_NAME
FROM COURSE C
LEFT JOIN PROFESSOR P ON P.COURSEID = C.COURSEID
ORDER BY P.PROFESSORID;
```

PROFESSORID	PROFESSORNAME	COURSEID	COURSE_NAME
500638478	Jane Smith	60001	History 101
501477909	Ben Roberts	60015	Management Information Systems 301
510628488	Maria Garcia	60002	Anthropology 102
511911003	Jacob Hill	60012	Management 371
520758466	James Johnson	60003	Economics 101
521333898	Brandon Miller	60006	Math 150
523448596	Dave Hernandez	60004	Accounting 285
531990002	Emily Nguyen	60011	Geology 101
540551321	Sue Anderson	60009	Management 372

10. Retrieve all Transaction information of Customers who put their purchases on their U-Bill.

```
SELECT *
FROM TRANSACTION
WHERE PAYMENTMETHOD IN (SELECT PAYMENTMETHOD FROM TRANSACTION
WHERE PAYMENTMETHOD = 'U-Bill')
```

Results | Explain | Describe | Saved SQL | History

TRANSACTIONID	DATE_OF_PURCH	CUSTOMERID	PRODUCTID	PAYMENTMETHOD
50020	05-May-2023	90008	55461	U-Bill
50018	30-Apr-2023	90011	88900	U-Bill
50017	29-Apr-2023	90003	67431	U-Bill
50016	24-Apr-2023	90001	15221	U-Bill
50015	20-Apr-2023	90015	44568	U-Bill
50014	16-Apr-2023	90014	77890	U-Bill
50011	08-Apr-2023	90011	33256	U-Bill
50008	06-Mar-2023	90008	18665	U-Bill
50003	06-Mar-2023	90003	29472	U-Bill

11. Retrieve Transaction ID, Product ID, and Date of Purchase, ordered by Date of Purchase.

```
SELECT TRANSACTIONID, PRODUCT.PRODUCTID,DATE_OF_PURCH
FROM TRANSACTION,PRODUCT
WHERE TRANSACTION.PRODUCTID=PRODUCT.PRODUCTID
ORDER BY DATE_OF_PURCH;
```

TRANSACTIONID	PRODUCTID	DATE_OF_PURCH
50006	56745	10-Jan-2023
50007	75543	10-Jan-2023
50002	38401	17-Jan-2023
50001	28401	15-Feb-2023
50010	56478	15-Feb-2023
50005	83027	17-Feb-2023
50004	10384	03-Mar-2023

12. Retrieve Customer and Transaction information ordered by Customer ID.

```
SELECT C.CUSTOMERID, CUST_NAME, CUST_TYPE, ADDRESS, TRANSACTIONID,
DATE_OF_PURCH, PAYMENTMETHOD
FROM CUSTOMER_P C
INNER JOIN TRANSACTION T ON T.CUSTOMERID = C.CUSTOMERID
ORDER BY C.CUSTOMERID;
```

CUSTOMERID	CUST_NAME	CUST_TYPE	ADDRESS	TRANSACTIONID	DATE_OF_PURCH	PAYMENTMETHOD
90001	Max Russo	Current Student	222 Lincoln Way	50016	24-Apr-2023	U-Bill
90001	Max Russo	Current Student	222 Lincoln Way	50001	15-Feb-2023	U-Bill
90002	Troy Bolton	Alumni	515 Welch Ave	50002	17-Jan-2023	CreditCard
90003	Sharpay Evans	Current Student	455 S. Duff Ave.	50017	29-Apr-2023	U-Bill
90003	Sharpay Evans	Current Student	455 S. Duff Ave.	50003	06-Mar-2023	U-Bill

13. Retrieve a list of the Customer Name, Product Description, and Date of Purchase.

```
SELECT CUST_NAME,PRODUCTDESCRIPTION,DATE_OF_PURCH
FROM PRODUCT,TRANSACTION,CUSTOMER_P
WHERE PRODUCT.PRODUCTID =TRANSACTION.PRODUCTID
AND TRANSACTION.CUSTOMERID=CUSTOMER_P.CUSTOMERID;
```

CUST_NAME	PRODUCTDESCRIPTION	DATE_OF_PURCH
Max Russo	Vintage Cy Shirt	15-Feb-2023
Troy Bolton	ISU Hat	17-Jan-2023
Debby Ryan	Cy Bucket Hat	17-Feb-2023
Sharpay Evans	Cardinal & Gold Planners	06-Mar-2023
Stefan Salvatore	Vintage Cy Sweatshirt	03-Mar-2023
Meredith Grey	Alumni T-Shirt	10-Jan-2023
Elaine Benes	Red Cy Sweatshirt	10-Jan-2023
Malibu Barbie	Cy Sweatshirt	06-Mar-2023

14. Retrieve Transaction information only for “Current Students”.

```
SELECT
TRANSACTIONID,DATE_OF_PURCH,TRANSACTION.CUSTOMERID,TRANSACTION.PRODUCTID,PAYMENTMETHOD
FROM PRODUCT,TRANSACTION,CUSTOMER_P
WHERE PRODUCT.PRODUCTID =TRANSACTION.PRODUCTID
AND TRANSACTION.CUSTOMERID=CUSTOMER_P.CUSTOMERID
AND CUSTOMER_P.CUST_TYPE='Current Student';
```

TRANSACTIONID	DATE_OF_PURCH	CUSTOMERID	PRODUCTID	PAYMENTMETHOD
50001	15-Feb-2023	90001	28401	U-Bill
50003	06-Mar-2023	90003	29472	U-Bill
50008	06-Mar-2023	90008	18665	U-Bill
50011	08-Apr-2023	90011	33256	U-Bill
50014	16-Apr-2023	90014	77890	U-Bill
50015	20-Apr-2023	90015	44568	U-Bill
50016	24-Apr-2023	90001	15221	U-Bill
50017	29-Apr-2023	90003	67431	U-Bill

15. Retrieve Product information, Supplier information and Expense that had a Shipment of 100 items.

```
SELECT PRODUCT.PRODUCTID, PRICE, SUPPLIER.COMPANYNAME,
PRODUCTDESCRIPTION, PRODUCT_TYPE, QUANTITY, EXPENSE
FROM PRODUCT, SUPPLIER
WHERE PRODUCT.PRODUCTID = SUPPLIER.PRODUCTID
AND PRODUCT.PRODUCTID IN (SELECT PRODUCTID FROM SUPPLIER WHERE
QUANTITY = 100);
```

PRODUCTID	PRICE	COMPANYNAME	PRODUCTDESCRIPTION	PRODUCT_TYPE	QUANTITY	EXPENSE
28401	35	Shirts Co.	Vintage Cy Shirt	Apparel	100	2500
18665	45	Champion	Cy Sweatshirt	Apparel	100	3500
23367	25	Under Armor	ISU Beanie	Apparel	100	1500
33256	20	Uni Accessories	Vintage Cy Mug	Miscellaneous	100	1000
77659	1000	Dell	Laptop	Technology	100	90000
15567	1300	Apple	MacBook	Technology	100	110000
75543	50	Nike	Red Cy Sweatshirt	Apparel	100	4000

Conclusion

In conclusion, for this project, we wanted to improve the database system for the Iowa State Bookstore. Iowa State Bookstore sells a variety of products including clothing items, textbooks, and class materials. We had many roadblocks during the implementation of data because it made us realize we had to update previous tables to keep our proposed system up to date. Another issue we had was setting up regular meetings with everyone's schedule. After this our group was able to collaborate effectively and efficiently, resulting in us being a compatible team.

In conclusion, this project has highlighted several issues and lessons that can be applied to future work from both a teamwork and database knowledge standpoint. We noticed some issues from the viewpoint of database knowledge, including data discrepancies, problems organizing data, and a lack of uniformity across diverse sources. These issues show that data administration has to be simplified and data quality control needs to be given more focus. On the other hand, the project demonstrated the value of good collaboration, communication, and task delegation from the perspective of teamwork. The group also discovered the importance of having clear expectations and objectives, checking in frequently, and being adaptable to changes. Overall, this project has provided valuable insights that will help improve future projects. By addressing the challenges identified, and applying the lessons learned, we can enhance our database knowledge and teamwork skills, leading to more successful projects in the future.