Student Expense Track Database Management System

IST659 Project Implementation Report



Instructor: Prof. Hernando Hoyos

**CONTENTS**

1. Project Summary…………………………………………………………………………3
2. Entity and Attribute Table……………………………………………………………......5
3. Data Dictionary…………………………………………………………………………...9
4. Entity Relationship Diagram…………………………………………………………….13
5. Database System Architecture………………………………………………………...…14
6. SQL Scripts for creating tables and inserting data………………………………………15
7. Major Data Questions……………………………………………………………………32
8. Interfaces………………………………………………………………………………....37

**PROJECT SUMMARY**

Syracuse University is a private institution which consists of diverse students enrolled from all over the world. The university has a number of on campus stores wherein the students can purchase various commodities like books, clothing, grocery, electronics and other stuff which are required for educational purposes. One of the types of stores of paramount importance to a student’s academics are the bookstores.

The university has two bookstores, one at the Schine Center and another one at Slocum center. The name ‘bookstore’ is actually a misleading name for these stores, because they typically are a hub for everything that you require to get comfortable in your new home away from home. The bookstore has products ranging from hoodies and t-shirts to sweats, personal care to general mart items, electronics to stationary and a lot more. With such a plethora of stocked items, it gets a heavy footfall at any point of the day, which means a lot of purchase history is recorded in the bookstore database.

The bookstore is a repository of all the purchases of each student. Whenever a student makes a purchase and swipes his/her SUID, after getting categorized as a student, it is automatically added to his/her database history. Currently, the bookstore uses the SUID, transaction ID and the price of the product to store information pertaining to a student.

There isn’t any system which enables students to remotely keep track of their bookstore purchases. The student is not availed of such a facility. This becomes a hassle for the student who makes a purchase at the bookstore, sometimes unaware that the very item he/she is purchasing has already been purchased in the near past.

Here, we are going to build and implement a central system, from the student’s perspective, where the student can keep track of all his/her expenses made during the particular academic year. The database will primarily consist of the first and last names of students, their SUID, prices of products at every purchase and respective dates till that point of time. This system will help the student to avoid the redundancy of buying the same product in the future.

The students will be able to access their expenses by logging into the system with the help of their SU credentials. This will be very helpful to the students in terms of managing their expenses at the bookstore and would give them all the information they need on a single database. It would be as easy as the click of a button.

The system has two primary users:

* Student: Can enter the product details and track the expenses made during an academic year
* Parents: Has the functionality of viewing the student expense details

Following Business Rules were considered while building the system:

* Purchaser should be a student of Syracuse University
* He/she must have a valid SUID
* The purchase should only be made in the SU Bookstore
* Every transaction must be of minimum $5 every time
* Purchases must be made during the semester
* The mode of payment should be only cash or card
* Student can have one or two parents, every parent can be related to one or more students

**ENTITY AND ATTRIBUTE TABLES**

**STUDENT**

|  |  |
| --- | --- |
| Attributes | Description |
| SUID | Primary key  Uniquely identifies the student |
| SFirstName | First name of the student |
| SLastName | Last name of the student |
| SEmailID | Email ID of the student |

**PARENT**

|  |  |
| --- | --- |
| Attributes | Description |
| Parent ID | Primary key  Unique identifier for each set of parents |
| SUID | Foreign key  SUID which uniquely identifies their ward |
| PFirstName | First name of the parent |
| PLastName | Last name of the parent |

**PARENT\_STUDENT**

|  |  |
| --- | --- |
| Attributes | Description |
| ParentID | Composite Primary Key |
| SUID | Composite Primary Key |

**PURCHASE**

|  |  |
| --- | --- |
| Attributes | Description |
| Purchase ID | Primary key  Unique identification for each purchase |
| SUID | Composite Primary Key |
| ProductID | Foreign Key |
| PurchaseDate | Mentions the date of the purchase |
| PaymentMethod | Specifies the mode of payment |

**PRODUCTS**

|  |  |
| --- | --- |
| Attributes | Description |
| ProductID | Primary Key  Unique identifier of the product |
| DepartmentID | Foreign Key |
| ProductName | Name of the product |
| ProductPrice | Price of the product being entered |

**PURCHASE\_PRODUCTS**

|  |  |
| --- | --- |
| Attributes | Description |
| Purchase ID | Composite Primary Key |
| Product ID | Composite Primary Key |
| SUID | Composite Primary Key |
| Quantity | Quantity of products the student purchases |

**DEPARTMENT**

|  |  |
| --- | --- |
| Attributes | Description |
| DepartmentID | Primary Key  Unique value pertaining to each department |
| DepartmentName | Specifies the department name |

**BOOKSTORE**

|  |  |
| --- | --- |
| Attributes | Description |
| Bookstore ID | Primary Key  Unique identification for each of the two bookstores |
| Bookstore Name | Name of the bookstore |

**PRODUCTS\_BOOKSTORE**

|  |  |
| --- | --- |
| Attributes | Description |
| Product ID | Composite Primary Key |
| Bookstore ID | Composite Primary Key |

**DATA DICTIONARY**

**STUDENT Entity**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Attribute Name | Field Type | NULL or NOT NULL | Description |
| Primary Key | SUID | INTEGER | NOT NULL | Primary key, Unique identifier for each set of parents |
|  | SFirstName | VARCHAR(20) | NOT NULL | First name of the student |
|  | SLastName | VARCHAR(20) | NOT NULL | Last name of the student |
|  | SEmailID | VARCHAR(20) | NOT NULL | Email ID of the student |

**PARENT Entity**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Attribute Name | Field Type | NULL or NOT NULL | Description |
| Primary Key | ParentID | INTEGER | NOT NULL | Primary key, Unique identifier for each set of parents |
| Foreign Key | SUID | VARCHAR(20) | NOT NULL | Primary Key of STUDENT table |
|  | PFirstName | VARCHAR(20) | NOT NULL | First name of the parent |
|  | PLastName | INTEGER | NOT NULL | Last name of the parent |

**PARENT\_STUDENT Entity**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Attribute Name | Field Type | NULL or NOT NULL | Description |
| Composite Primary Key | ParentID | INTEGER | NOT NULL | Composite Primary Key |
| Composite Primary Key | SUID | INTEGER | NOT NULL | Composite Primary Key |

**PURCHASE Entity**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Attribute Name | Field Type | NULL or NOT NULL | Description |
| Primary Key | PurchaseID | INTEGER | NOT NULL | Primary key, Unique identification for each purchase |
| Composite Primary Key | SUID | INTEGER | NOT NULL | Composite Primary Key |
| Foreign Key | ProductID | INTEGER | NOT NULL | Primary Key of PRODUCTS table |
|  | PurchaseDate | DATETIME |  | Mentions the date of the purchase |
|  | PaymentMethod | VARCHAR(20) |  | Specifies the mode of payment |

**PRODUCTS Entity**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Attribute Name | Field Type | NULL or NOT NULL | Description |
| Primary Key | ProductID | INTEGER | NOT NULL | Primary Key, Unique identifier of the product |
| Foreign Key | DepartmentID | INTEGER |  | Primary Key of DEPARTMENT table |
|  | ProductName | VARCHAR(30) |  | Name of the product |
|  | ProductPrice | DECIMAL(4,2) |  | Price of the product being entered |

**PURCHASE\_PRODUCTS Entity**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Attribute Name | Field Type | NULL or NOT NULL | Description |
| Composite Primary Key | PurchaseID | INTEGER | NOT NULL | Primary key of PURCHASE table |
| Composite Primary Key | ProductID | INTEGER | NOT NULL | Primary key of PRODUCT table |
| Composite Primary Key | SUID | INTEGER | NOT NULL | Primary key of STUDENT table |
|  | Quantity | INTEGER |  | Quantity of products the student purchases |

***DEPARTMENT Entity:***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Attribute Name | Field Type | NULL or NOT NULL | Description |
| Primary Key | DepartmentID | INTEGER | NOT NULL | Primary Key, Unique value pertaining to each department |
|  | DepartmentName | VARCHAR(20) |  | Specifies the department name |

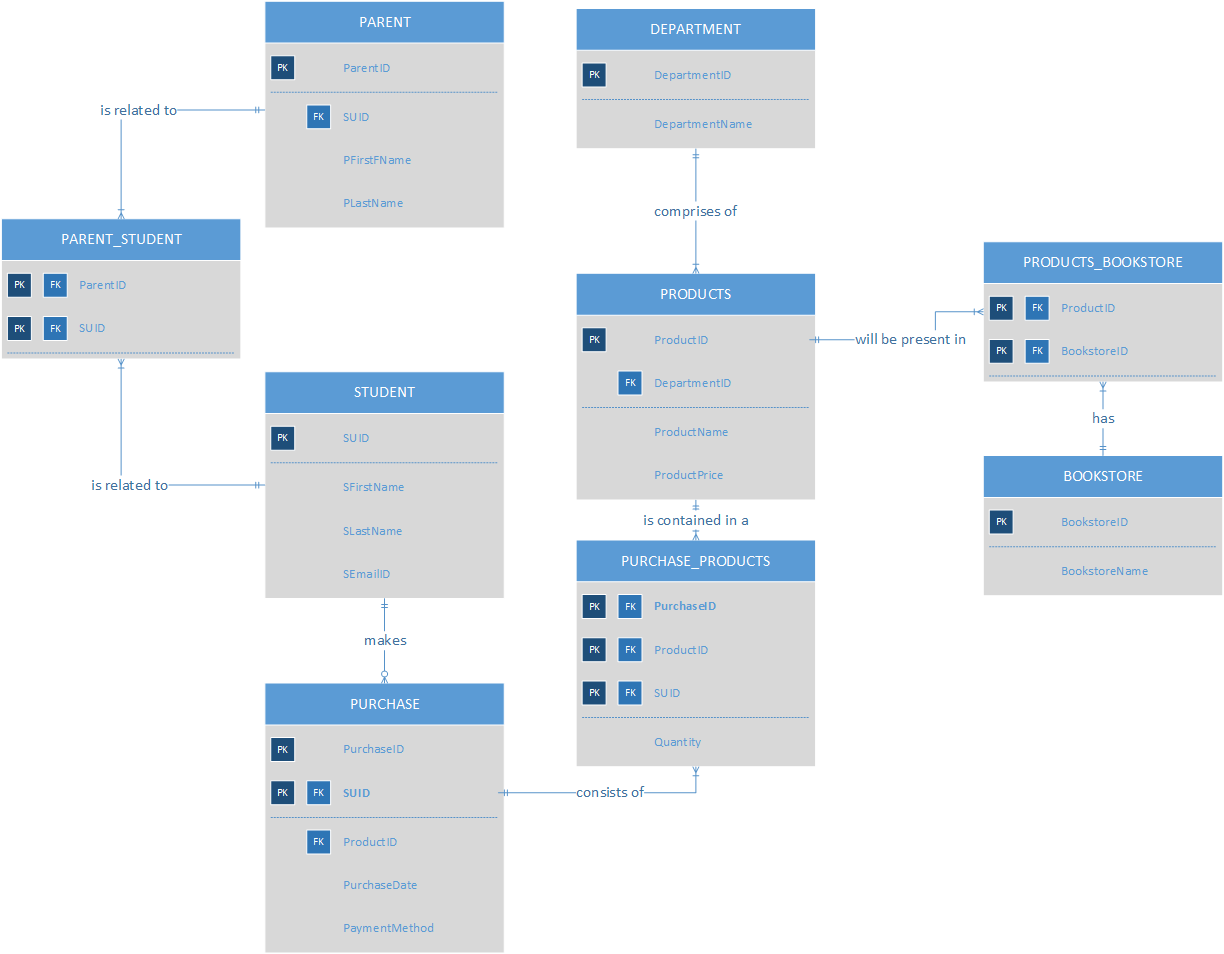
**BOOKSTORE Entity**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Attribute Name | Field Type | NULL or NOT NULL | Description |
| Primary Key | BookstoreID | INTEGER | NOT NULL | Primary Key, Unique identification for each of the two bookstores |
|  | BookstoreName | VARCHAR(30) |  | Name of the bookstore |

**PRODUCTS\_BOOKSTORE Entity**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Attribute Name | Field Type | NULL or NOT NULL | Description |
| Composite Primary Key | ProductID | INTEGER | NOT NULL | Composite Primary Key |
| Composite Primary Key | BookstoreID | INTEGER | NOT NULL | Composite Primary Key |

**ENTITY RELATIONSHIP DIAGRAM**

****

**DATABASE SYSTEM ARCHITECTURE**

The following tools were used by us to create the design and implement the project.

**1. MS Visio**

* We created entity relationship diagram using MS Visio. Here we created entities.
* We also specified their attributes along with the keys (primary & foreign).
* We also created Associative entity as well during Entity relationship diagram building process for a better data flow.
* We developed the relationships and dependencies between entities.

**2. SQL Server**

* We used SQL Server as the database that stored all the tables and their data.
* We created tables in the database using SQL queries.
* We used SQL server insert data into the tables and by using select query we also viewed the data.

**3. MS Access**

* We used MS Access to create the interface for the system. Using Access, we link our tables that were created in SQL Server by ODBC connection and imported all the tables from SQL Server along with data.
* After table extraction we created various forms that could take user input or display the necessary information to the users wish to obtain with the help of buttons.
* Based on the data, we used MS Access to generate reports for users to help them understand large data in a smaller and easy understandable format.

**SQL SCRIPTS FOR CREATING TABLE AND INSERTING DATA**

-------------------------------- Create Table Query-------------------------------------------

***--Student Table***

create table Student

(

SUID INTEGER NOT NULL,

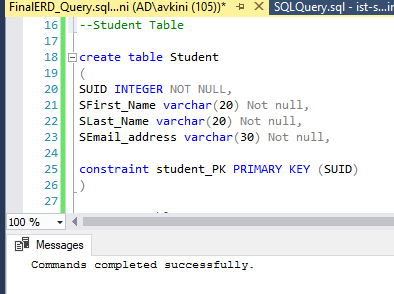
SFirstName varchar(20) Not null,

SLastName varchar(20) Not null,

SEmailID varchar(30) Not null,

constraint student\_PK PRIMARY KEY (SUID)

)



***--Parent Table***

create table Parent

(

ParentID INTEGER NOT NULL,

PFirst\_Name Varchar(20) NOT NULL,

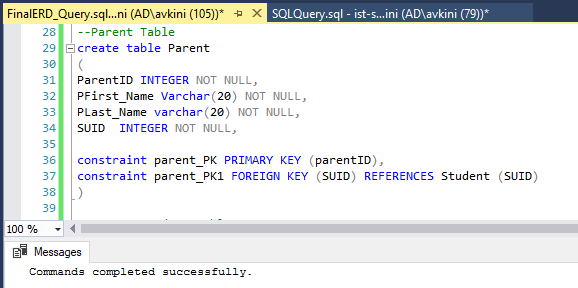
PLast\_Name varchar(20) NOT NULL,

SUID INTEGER NOT NULL,

constraint parent\_PK PRIMARY KEY (parentID),

constraint parent\_PK1 FOREIGN KEY (SUID) REFERENCES Student (SUID)

)



***--Parent\_Student table***

create table Parent\_Student

(

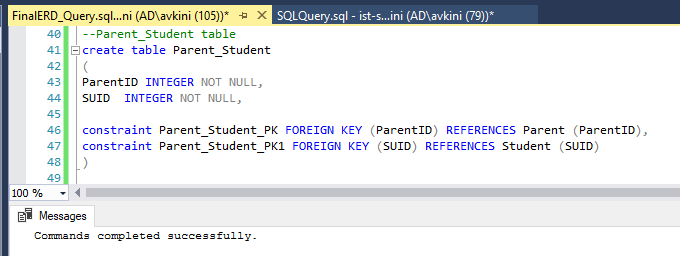
ParentID INTEGER NOT NULL,

SUID INTEGER NOT NULL,

constraint Parent\_Student\_PK FOREIGN KEY (ParentID) REFERENCES Parent (ParentID),

constraint Parent\_Student\_PK1 FOREIGN KEY (SUID) REFERENCES Student (SUID)

)

****

***--Products table***

CREATE TABLE Products

(

ProductID INTEGER NOT NULL,

ProductName VARCHAR(30),

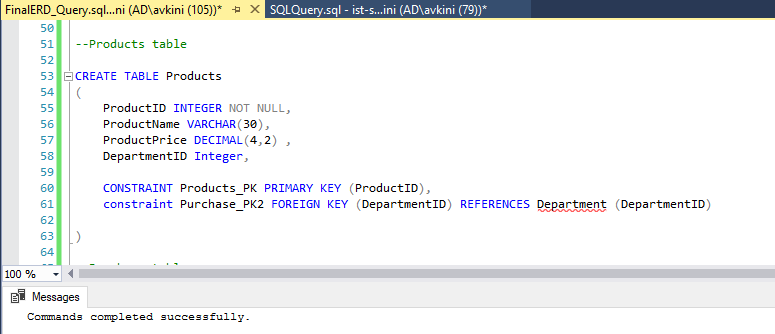
ProductPrice DECIMAL(4,2) ,

DepartmentID Integer,

CONSTRAINT Products\_PK PRIMARY KEY (ProductID),

constraint Purchase\_PK2 FOREIGN KEY (DepartmentID) REFERENCES Department (DepartmentID)

)



***--Purchase table***

create table Purchase

(

SUID INTEGER NOT NULL,

PurchaseID INTEGER NOT NULL,

ProductID INTEGER NOT NULL,

PurchaseDate DATETIME,

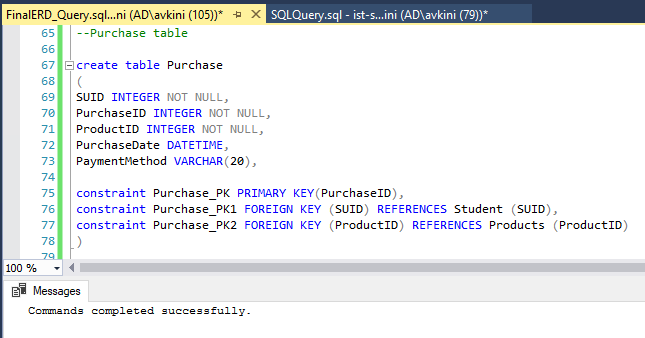
PaymentMethod VARCHAR(20),

constraint Purchase\_PK PRIMARY KEY(PurchaseID),

constraint Purchase\_PK1 FOREIGN KEY (SUID) REFERENCES Student (SUID),

constraint Purchase\_PK2 FOREIGN KEY (ProductID) REFERENCES Products (ProductID)

)



***--Purchase Products table***

create table Purchase\_Products

(

SUID INTEGER NOT NULL,

PurchaseID INTEGER NOT NULL,

ProductID INTEGER NOT NULL,

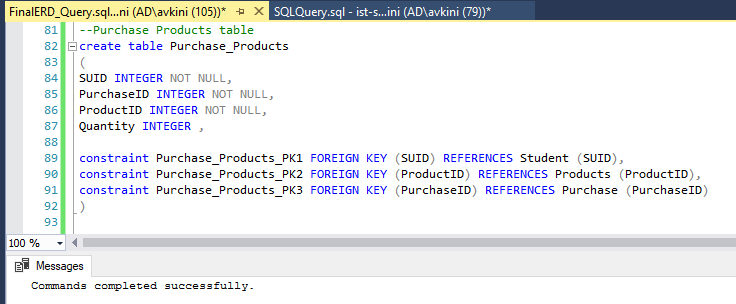
Quantity INTEGER ,

constraint Purchase\_Products\_PK1 FOREIGN KEY (SUID) REFERENCES Student (SUID),

constraint Purchase\_Products\_PK2 FOREIGN KEY (ProductID) REFERENCES Products (ProductID),

constraint Purchase\_Products\_PK3 FOREIGN KEY (PurchaseID) REFERENCES Purchase (PurchaseID)

)



***--Department table***

create table Department

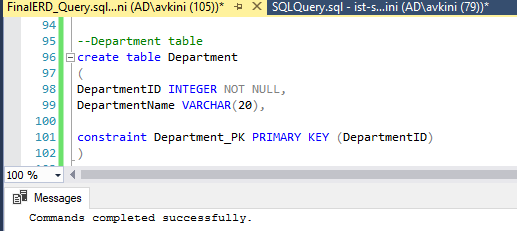
(

DepartmentID INTEGER NOT NULL,

DepartmentName VARCHAR(20),

constraint Department\_PK PRIMARY KEY (DepartmentID)

)



***--Products\_bookstore table***

create table Products\_Bookstore

(

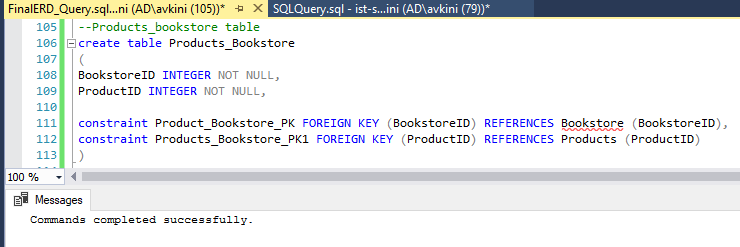
BookstoreID INTEGER NOT NULL,

ProductID INTEGER NOT NULL,

constraint Product\_Bookstore\_PK FOREIGN KEY (BookstoreID) REFERENCES Bookstore (BookstoreID),

constraint Products\_Bookstore\_PK1 FOREIGN KEY (ProductID) REFERENCES Products (ProductID)

)



***--Bookstore Table***

CREATE TABLE Bookstore

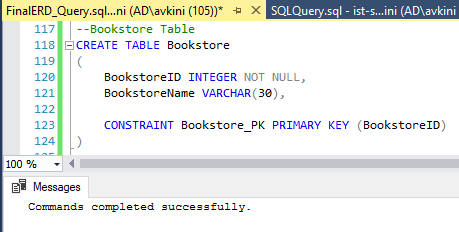
(

BookstoreID INTEGER NOT NULL,

BookstoreName VARCHAR(30),

CONSTRAINT Bookstore\_PK PRIMARY KEY (BookstoreID)

)



-------------------------------- Insert Table Query-------------------------------------------

***--Student Table***

Insert into Student values

('991983467','Mick','Rogers','mrogers@syr.edu'),

('996578452','Aditya','Kini','avkini@syr.edu'),

('854675921','Rajvee','Shah','rshah@syr.edu'),

('996587412','Prathamesh','Datar','pdatar@syr.edu'),

('993254675','Anushi','Doshi','adoshi@syr.edu'),

('997853214','Pranjal','Sondkar','psondka@syr.edu'),

('857469146','Steven','Bendric','sbendr@syr.edu'),

('889412753','Hyun','Mia','hmia@syr.edu'),

('995328756','Josh','Muller','jmuller@syr.edu'),

('975864318','Mark','Smith','jmuller@syr.edu')

A screenshot of a cell phone

Description automatically generated

***--Parent Table***

insert into Parent values

('1','Peter','John','854675921'),

('2','Abe', 'Kelly','857469146'),

('3','Adam','Sultz','889412753'),

('4','Elaine', 'Thomas','975864318'),

('5','Carl', 'Lyle','991983467'),

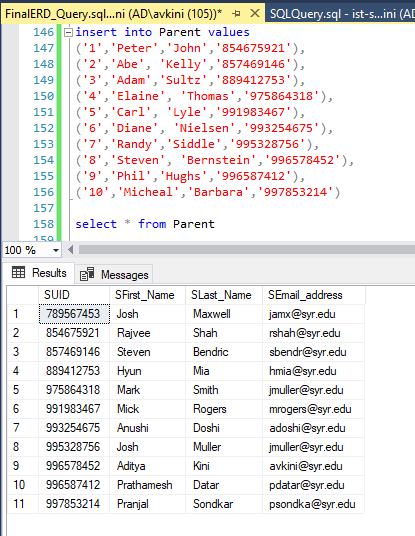
('6','Diane', 'Nielsen','993254675'),

('7','Randy','Siddle','995328756'),

('8','Steven', 'Bernstein','996578452'),

('9','Phil','Hughs','996587412'),

('10','Micheal','Barbara','997853214')



***--Parent\_Student table***

Insert into Parent\_Student values

('1','854675921'),

('2','857469146'),

('3','889412753'),

('4','975864318'),

('5','991983467'),

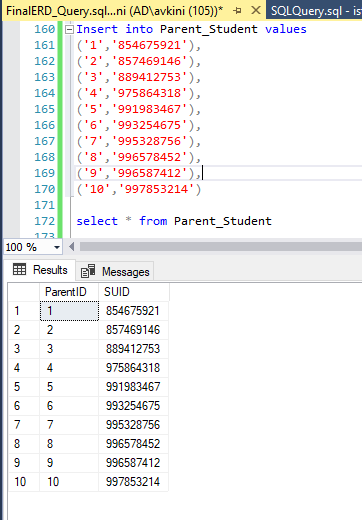
('6','993254675'),

('7','995328756'),

('8','996578452'),

('9','996587412'),

('10','997853214')



***--Products Table***

insert into Products values

('9001','Syracuse Cap','23.99','4001'),

('9002','Syracuse Tshirt','10','4001'),

('9003','Otto Small','8.99','4001'),

('9004','Syracuse Umbrella','17','4002'),

('9005','Syracuse Bottle','15','4002'),

('9006','Syracuse Sunglasses','20','4003'),

('9007','Brush set','4','4004')

('9008','Textbook','50.99','4005'),

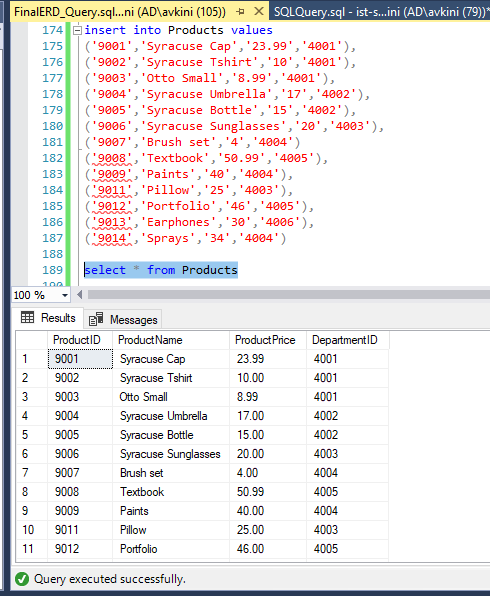
('9009','Paints','40','4004'),

('9011','Pillow','25','4003'),

('9012','Portfolio','46','4005'),

('9013','Earphones','30','4006'),

('9014','Sprays','34','4004')



***--Purchase Table***

insert into Purchase values

('991983467','1001','9002',GETDATE(),'Cash'),

('996578452','1002','9004','2019-11-25 00:00:00','Credit'),

('854675921','1003','9001','2019-10-12 00:00:00','Debit'),

('996587412','1004','9005','2019-09-03 00:00:00','Cash'),

('993254675','1005','9006','2019-11-23 00:00:00','Credit'),

('997853214','1006','9003','2019-11-11 00:00:00','Cash'),

('857469146','1007','9001','2019-11-25 00:00:00','Debit'),

('889412753','1008','9007','2019-10-26 00:00:00','Cash'),

('995328756','1009','9004','2019-10-16 00:00:00','Debit'),

('991983467','1011','9007','2019-10-15 00:00:00','Credit'),

('996578452','1013','9008','2019-11-29 00:00:00','Cash'),

('854675921','1012','9009','2019-09-02 00:00:00','Debit'),

('996587412','1014','9011','2019-09-12 00:00:00','Cash'),

('993254675','1015','9009','2019-10-09 00:00:00','Cash'),

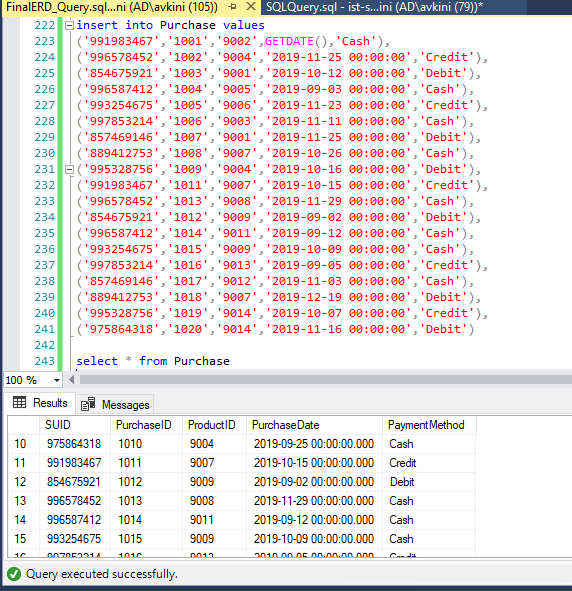
('997853214','1016','9013','2019-09-05 00:00:00','Credit'),

('857469146','1017','9012','2019-11-03 00:00:00','Cash'),

('889412753','1018','9007','2019-12-19 00:00:00','Debit'),

('995328756','1019','9014','2019-10-07 00:00:00','Credit'),

('975864318','1020','9014','2019-11-16 00:00:00','Debit')

****

***--Purchase\_Products Table***

insert into Purchase\_Products values

('991983467','1001','9002','3'),

('996578452','1002','9004','6'),

('854675921','1003','9001','4'),

('996587412','1004','9005','1'),

('993254675','1005','9006','9'),

('997853214','1006','9003','8'),

('857469146','1007','9001','7'),

('889412753','1008','9007','6'),

('995328756','1009','9004','5'),

('975864318','1010','9004','3'),

('991983467','1011','9007', '2'),

('996578452','1013','9008', '2'),

('854675921','1012','9009', '1'),

('996587412','1014','9011', '2'),

('993254675','1015','9009', '2'),

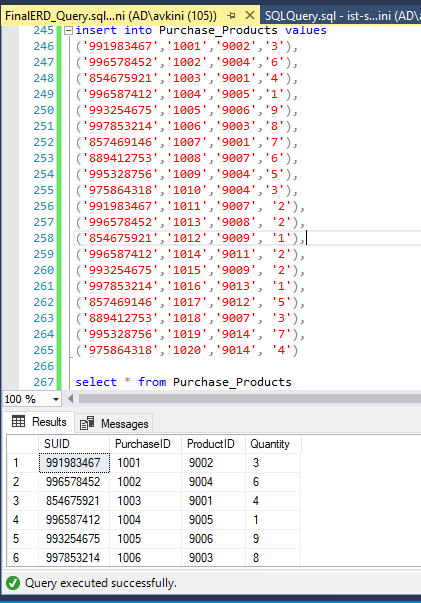
('997853214','1016','9013', '1'),

('857469146','1017','9012', '5'),

('889412753','1018','9007', '3'),

('995328756','1019','9014', '7'),

('975864318','1020','9014', '4')

****

***--Department Table***

insert into Department values

('4001','Clothing'),

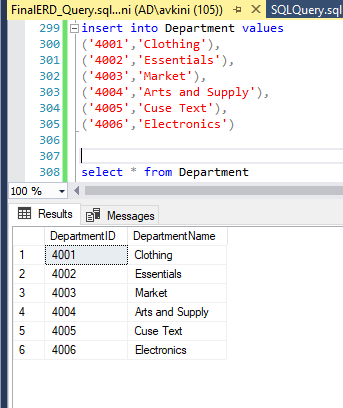
('4002','Essentials'),

('4003','Market'),

('4004','Arts and Supply'),

('4005','Cuse Text'),

('4006','Electronics')

****

***--Products\_Bookstore Table***

insert into Products\_Bookstore values

('111','9001'),

('222','9002'),

('222','9003'),

('111','9004'),

('222','9005'),

('111','9006'),

('222','9007'),

('111','9008'),

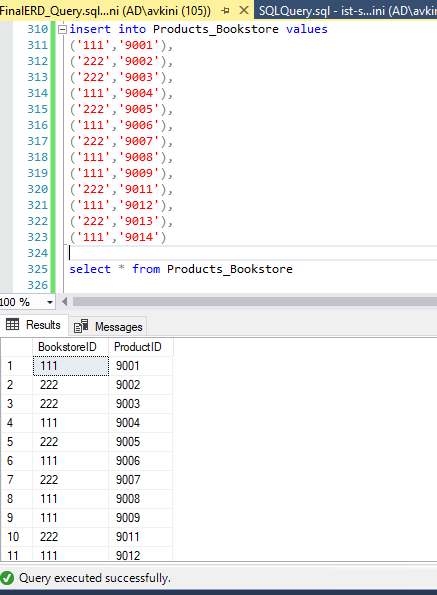
('111','9009'),

('222','9011'),

('111','9012'),

('222','9013'),

('111','9014')

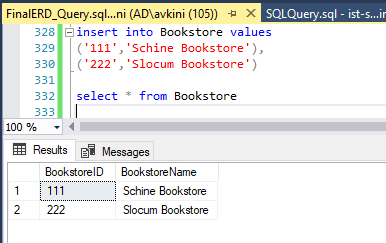
****

***--Bookstore Table***

insert into Bookstore values

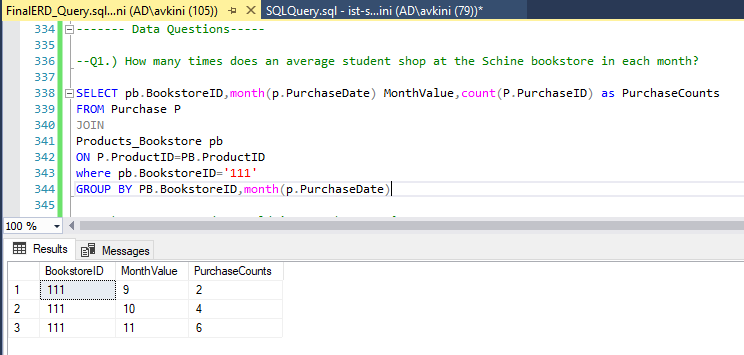
('111','Schine Bookstore'),

('222','Slocum Bookstore')

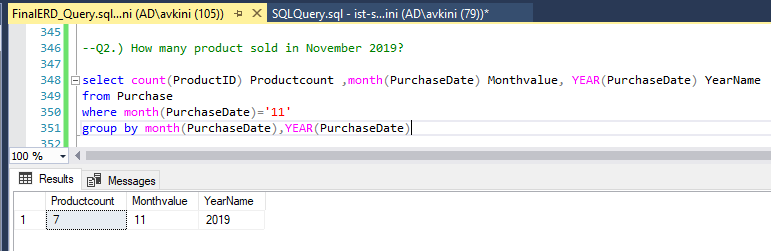
****

**MAJOR DATA QUESTIONS**

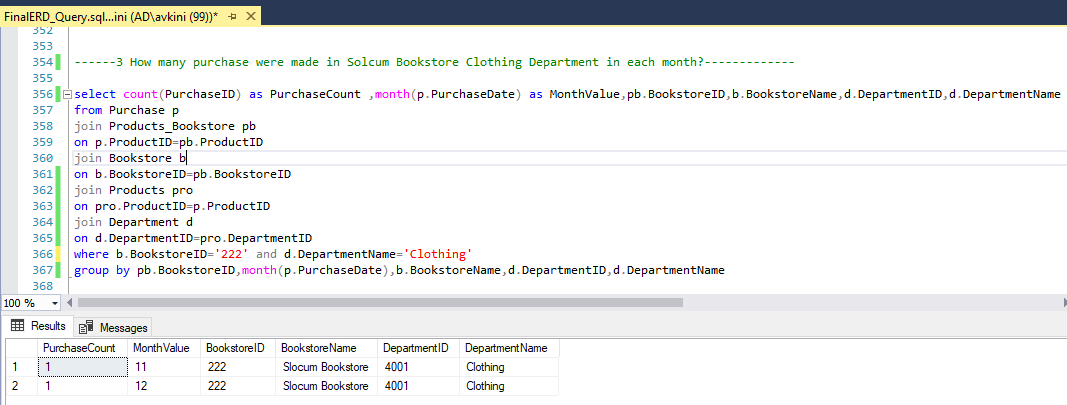
**Q1.) How many times does an average student shop at the Schine bookstore in each month?**



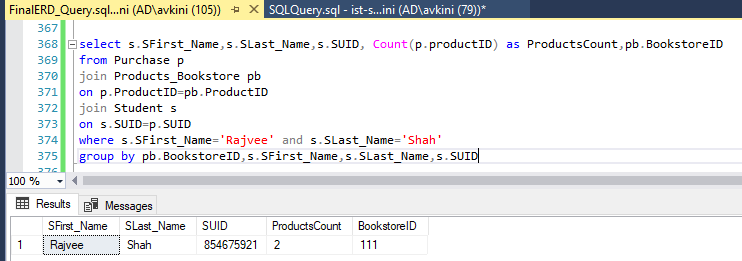
**Q2.) How many product sold in November 2019?**



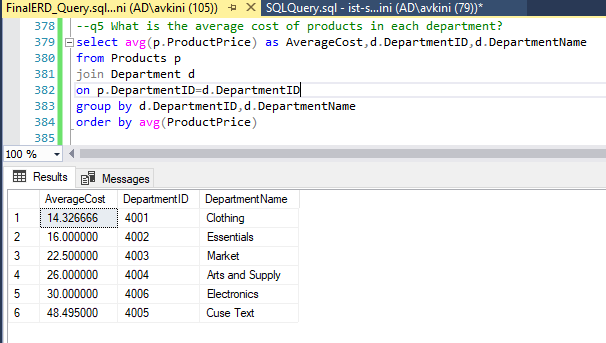
**Q3.) How many purchase were made in Solcum Bookstore Clothing Department in each month?**



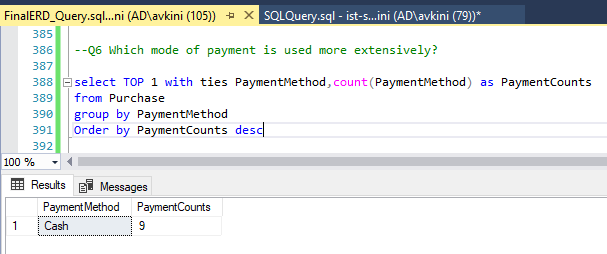
**Q4.) How many products are purchased by Rajvee Shah from each bookstore?**



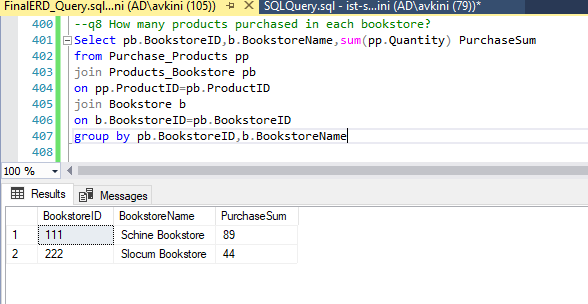
**Q5.) What is the average cost of products in each department?**



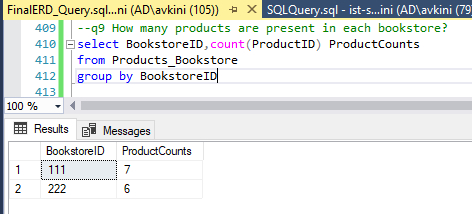
**Q6.) Which mode of payment is used more extensively?**



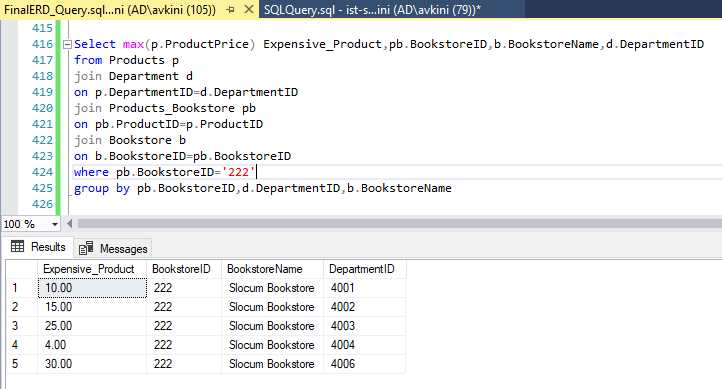
**Q7.) How many products purchased in each bookstore?**



**Q8.) How many products are present in each bookstore?**



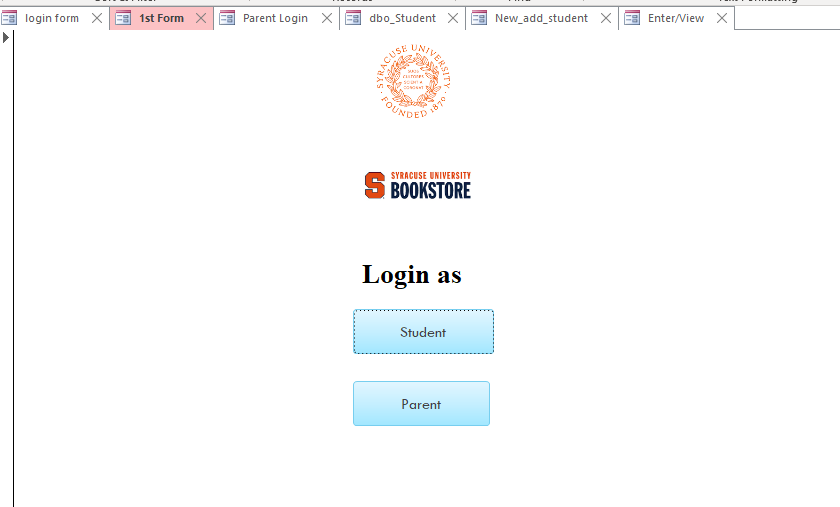
**Q9.) Which are the expensive product prices in Slocum Bookstore each department?**



**INTERFACES**

**STEP 1**

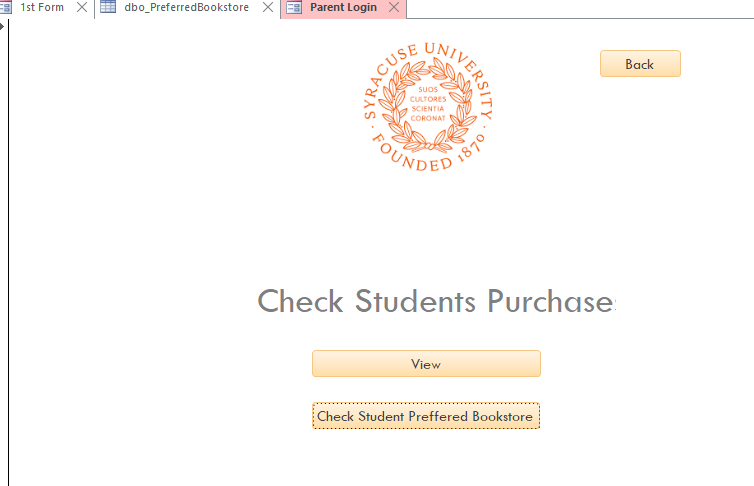
This step of the system provides two options: Student login and Parent Login. The form was created for the same and provided two buttons each which will redirect them to the next form for further use.



**PARENT INTERFACE**

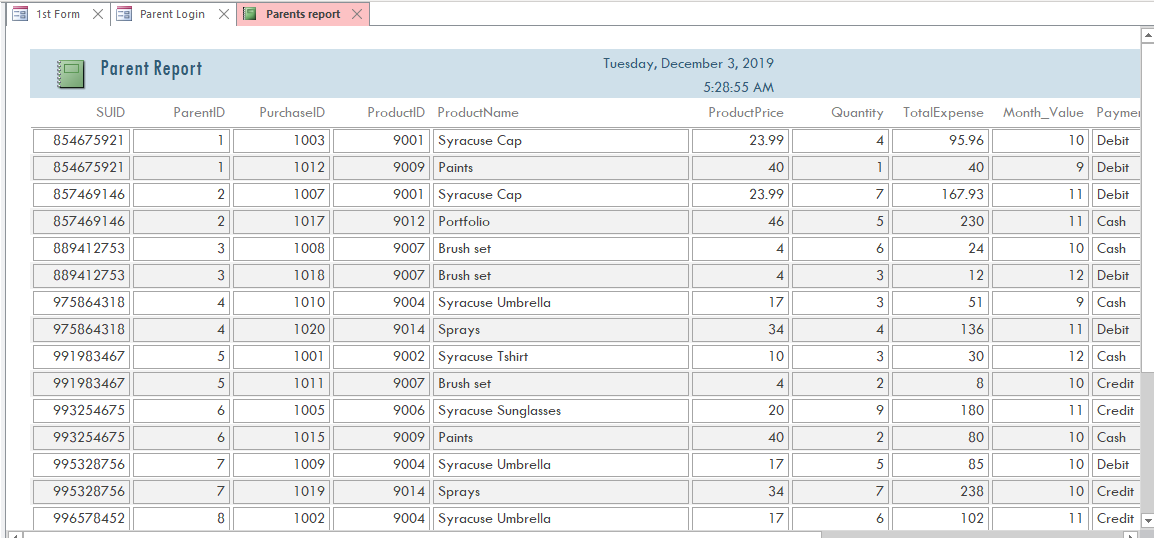
If the user click on Parent option in the above step he will be redirected to a Parent Login form which basically provides Parent with two options. One in the form of View button and other as check student preferred bookstore button.

Also a back button helps the user to traverse back to Step 1.



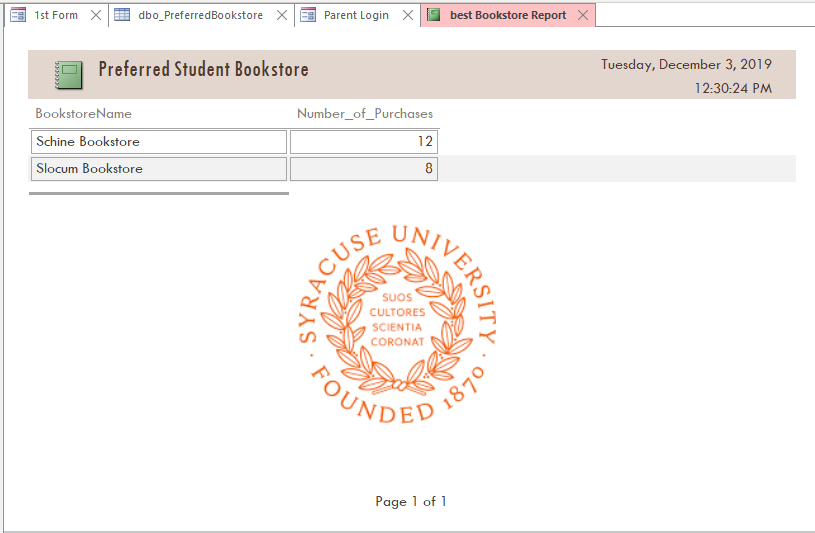
**VIEW OPTION**

When the parent click on the View button in the previous step he or she will be able to view a report which consists of Total expenses made by his/her child in the bookstore by using which payment mode and for which products.



**Check student preferred Bookstore Option**

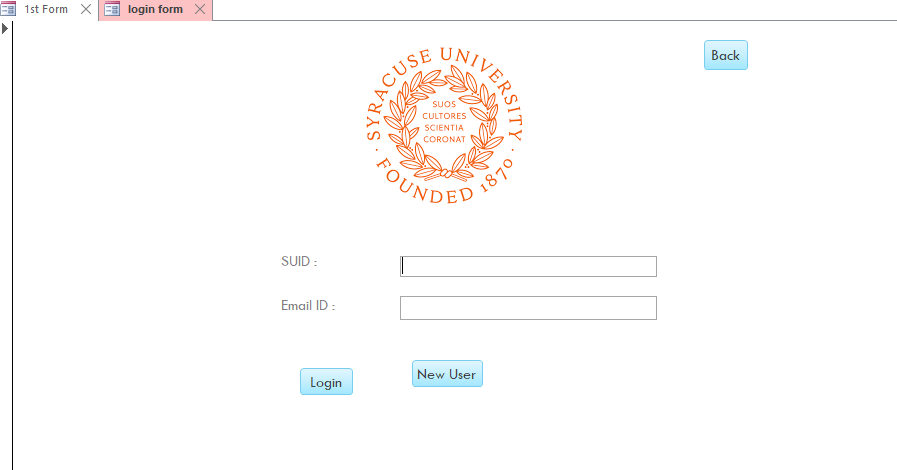
When the user clicks on the other option i.e Check student preferred bookstore in the Parent login form he/she can view a report which can help them understand which bookstore is widely used by students based on number of purchases.

****

**Student Login**

When the user selects Student option in Step 1, he/she will be redirected to the Student Login form. In this form the student can login using SUID and email address.   
Also in the form a New User option is also available which can help a new student to register into the system.

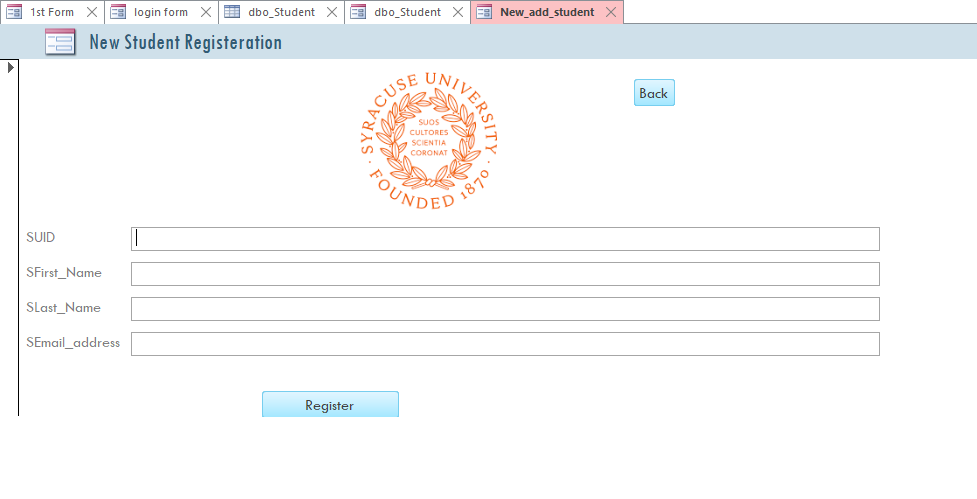
Along with this a Back option is also provided to go back to Step 1.

****

**New Student Form**

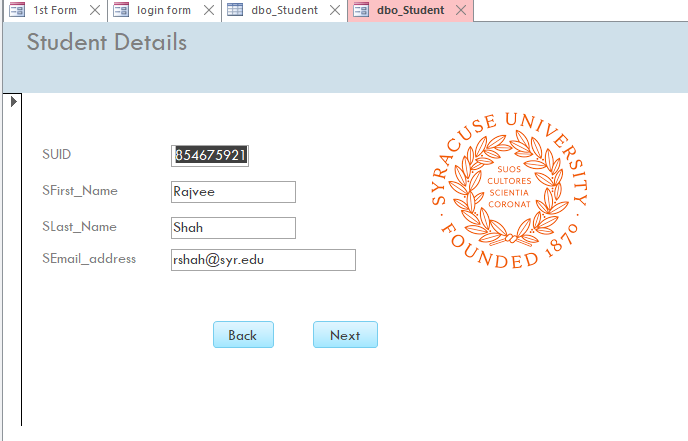
When the user selects New User option he/she will be redirected to this form wherein they can add their details SUID, First Name,Last Name and Email address and register into the system.

A Back option helps the student to go back to the Student Login page. A register button is provided which load the new data into the database.

****

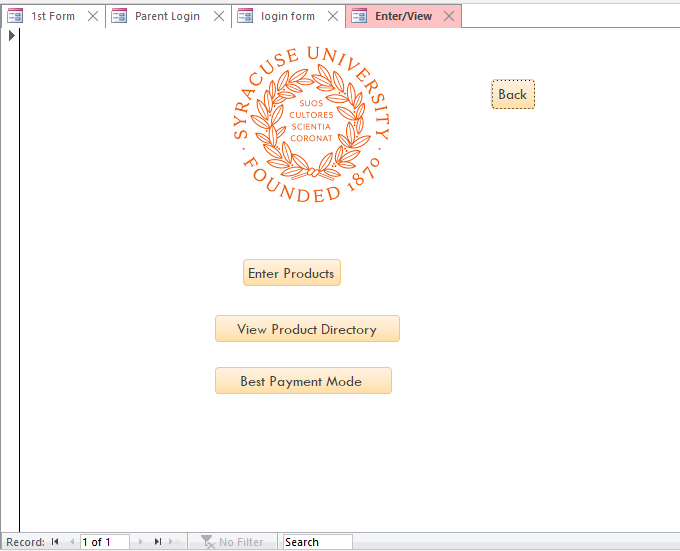
**Student details**

Upon successful login credentials the student will be directed to this form in which he can see his details. A Back option is provided which helps the user to back to the Student login form. Also a Next option is provided which helps them to move ahead to Enter/View form.

****

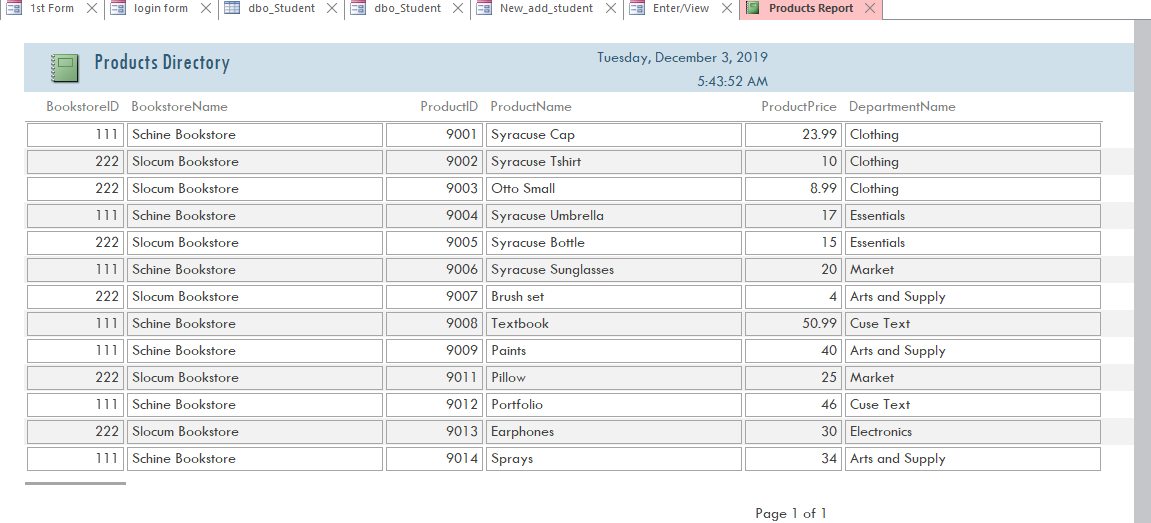
**Enter/View Form**

When the user selects the Next Option in the previous step he/she is directed to the Enter/View Form. In this form the student has 3 options. First is to add new products,second is to view the cost of each product in each bookstore and in which department.

****

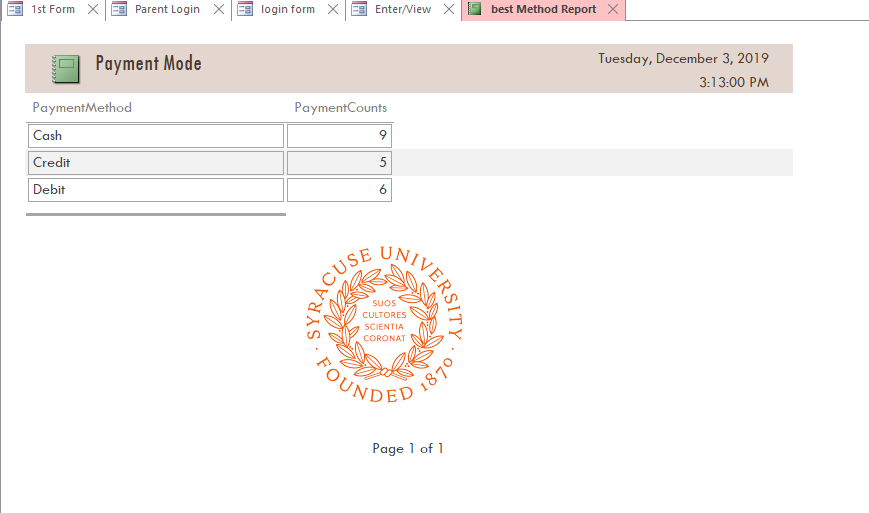
**View Product Directory Button**

When the user selects this button this helps them to view the list of products in both bookstore along with the price, name and department of the product.

****

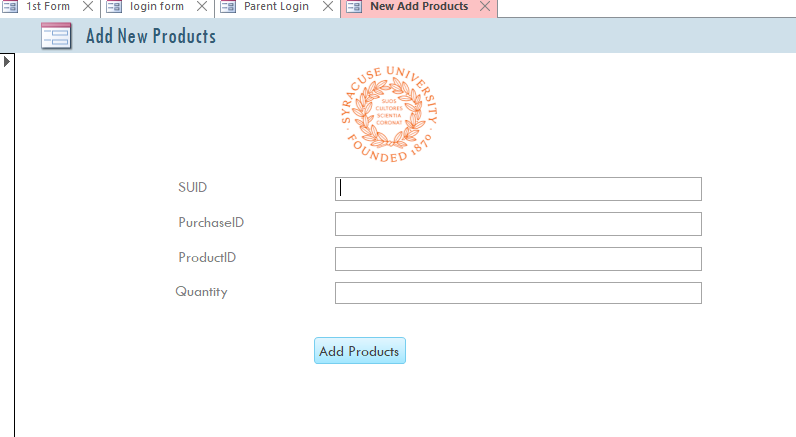
**Best Payment Mode Button**

When the user clicks on this option he/she will be able to view a report which it shows the most widely used Payment Mode by the students which can help them to decide which mode to be used as an alternative option.

****

**Enter Products button**

When the user clicks on this button he/she can be able to add the new details like SUID, PurchaseID, ProductID and Quantity. An Add products button is provided which loads the new data into the database.

****