CIS 9340 GROUP PROJECT
FINAL REPORT

Database Implementation for XYZ Gym

Group 2

Farwa Ismail, Trang Le, Hyunjik Lee, Yen-Ching Lin and Nishtha Ram

May 24, 2020

Contents

I. Project Proposal

1. Overview of the business

XYZ Gym is a New York City based chain of gyms founded in 2003 and expanding to 20 gyms across Manhattan, Brooklyn, Queens and the Bronx. It gives a wide range of membership opportunities at affordable prices. Prospective members can choose to have access to gym equipment at a lower rate or can go for some of the more premium options, including having a personalized trainer and training regimen. This flexibility has resulted in more growth than expected and the owners have brought in new management to sustain this growth.

The new management has proposed several areas of improvement, one of which includes the transformation of the current informal paper-based system to a formal database management system. This change has been necessitated by redundancies and events, such as a recent series of bookkeeping errors that caused the understocking of gym equipment, leading to disgruntled customers and lost sales. “Errors such as these are an inevitable consequence of overloaded paper based systems,” notes the incoming management report, “and changing to a computer database management system is of utmost importance.”

2. Information and user-specific requirements or special interests

The stakeholders of the database include gym staff (Role1: Manager, Role 2: Instructors, Role 3: Facility Staff, Role 4: Personal Trainers) and gym members. There are several requirements for this project:

(1) Providing the Manager with the ability to enter and view employees’ information such as basic personal information, performance, and salaries to keep track of the performance metrics of each branch such as revenue and staff performance.

(2) Providing the Facility Staff with the ability to view product information such as basic information of product, inventory, product supplier's information and the order record, and to keep a track of the maintenance and repair record of the equipment.

(3) Providing the instructors/trainers with the ability to view the schedule and client’s info.

(4) Providing members with the ability to view their profile, program, schedule and payment records.

Stakeholder	Requirements
Manager	• View and enter Employee basic information (employee ID, last name, first name, date of birth, phone number, address, salary etc.) • View Branch general information (branch ID, address, phone, etc.) • View Branch Total Payroll, Monthly Revenue • View Facility Staff performance
Member	Access their customer membership profiles which include: • General information (member ID, name, DOB, phone number, email, address,) • Membership plan (rate, level) Access the fitness class schedule and Trainer info: • Class info (date, time, room no.) • Personal Trainer Access their payment history: • Payment record (package, description, payment date, amount, status, method)
Instructors/ Trainers	Access their classes/clients, which include: • Class info (Class name, room, class schedule, equipment) • Client info (Name, contact info)
Facility Staff	Access equipment's information: • Equipment basic information (equipment-id, equipment name, equipment type, quantity, purchase date, service life) • Equipment maintenance and repair record (maintenance id, maintenance type, repair duration, cost, etc.) • View and enter product information: • View and enter product inventory stock number • View and enter product sales record

3. Objective for the database development

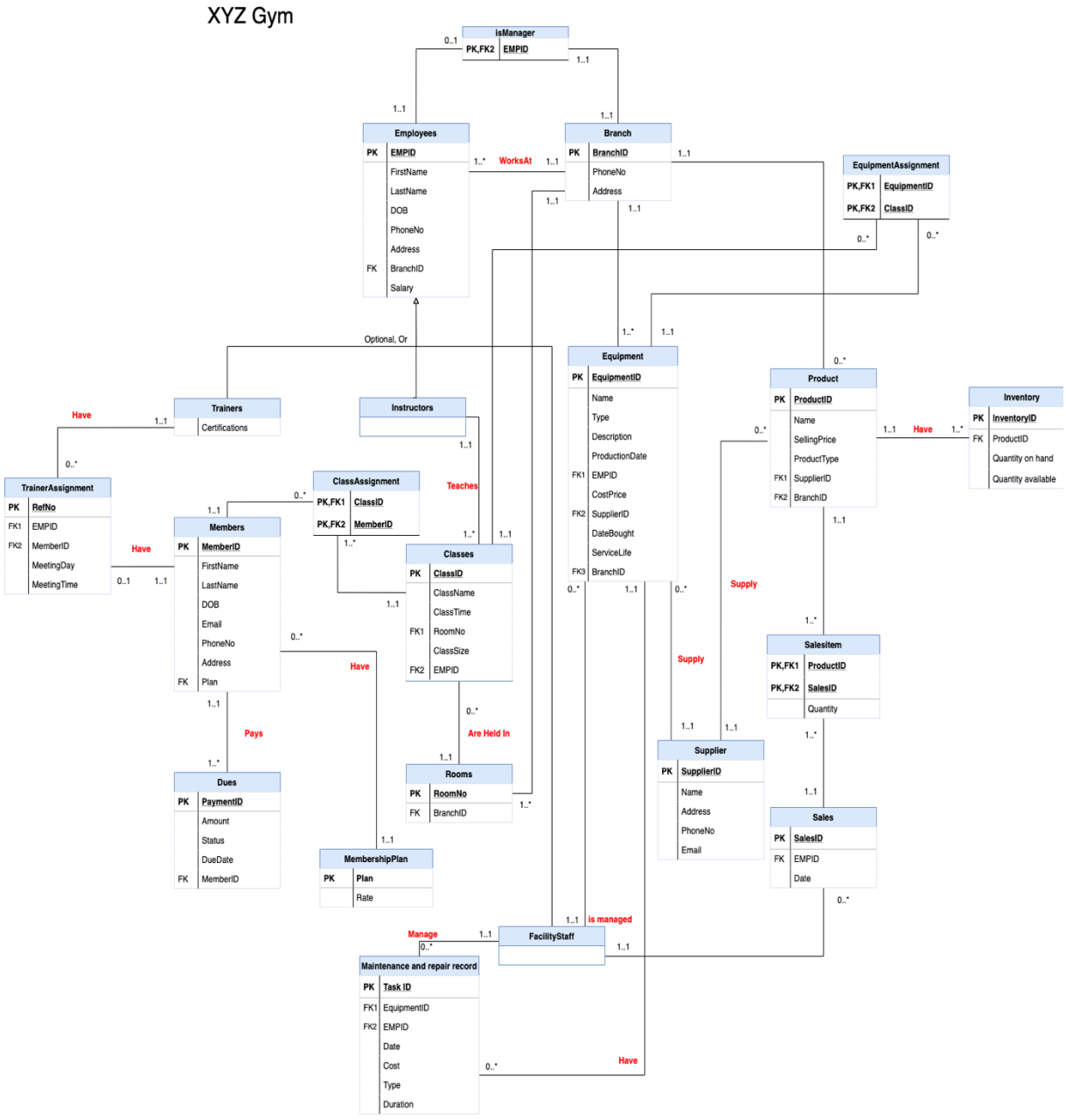
The database is designed to help the team at each XYZ gym, including managers, members, instructors/trainers and facility staffs, to improve the efficiency of daily operations by providing a full understanding of how the gym chain works. The objective of the database development is to efficiently operate XYZ gym in order to stay competitive in the market.

The goal will be achieved by:

- Allowing the manager to easily record and track instructors' and facility staff's personal information and salaries.
- Allowing the manager to access each branch's performance by recording Branch revenue and Staff performance
- Making a concise schedule for each session easily accessible to the instructors and staff.
- Allowing gym members to access class information, payment and plan information.
- Allowing the facility's staff to manage equipment, maintenance and repair record, product orders, and product inventory.

There are many factors contributing to the success or the failure of gym chain business. Managing the gym system is the most important factor. It's important to manage the gym session schedule and machines, and work on improving it. By utilizing this database system, we can not only make it more efficient, but also have XYZ gym provide better quality of service for its members.

II. Entity Relational Model



III. Relational Model and Normalization

Supplier (SupplierID char(5), Name varchar(40), PhoneNo char(10), Address varchar(150), Email varchar(30))

PK = {SupplierID}

Functional Dependency: SupplierID-> Name, PhoneNo, Address, Email

1NF: There is no repeating group structure in the relation. Therefore, it is in 1NF.

2NF: It is already in 1NF. Primary key has exactly one attribute. Therefore, it is in 2NF.

3NF: It is already in 2NF. There is no transitive functional dependency. Therefore, it is in 3NF.

BCNF: The only FD is due to the primary key. All determinants are candidate keys. Therefore, it is in BCNF.

4NF: There is no observable multi-valued dependency. Therefore, it is in 4NF.

Branch (BranchID char(3), PhoneNo char(10), Address varchar(150))

PK = {BranchID}

Functional Dependency: BranchID -> PhoneNo, Address

1NF: There is no repeating group structure in the relation. Therefore, it is in 1NF.

2NF: It is already in 1NF. Primary key has exactly one attribute. Therefore, it is in 2NF.

3NF: It is already in 2NF. There is no transitive functional dependency. Therefore, it is in 3NF.

BCNF: The only FD is due to the primary key. All determinants are candidate keys. Therefore, it is in BCNF.

4NF: There is no observable multi-valued dependency. Therefore, it is in 4NF.

Employees (EMPID char(5), FirstName varchar(40), LastName varchar(40) , DOB date, PhoneNo char(10), Address varchar(150), BranchID char(3), Salary CURRENCY)

PK = {EMPID}

FK={BranchID} references PK of Branch

Functional Dependency: EMPID-> FirstName, LastName, DOB, PhoneNo, Address, BranchID, Salary

1NF: There is no repeating group structure in the relation. Therefore, it is in 1NF.

2NF: It is already in 1NF. Primary key has exactly one attribute. Therefore, it is in 2NF.

3NF: It is already in 2NF. There is no transitive functional dependency. Therefore, it is in 3NF.

BCNF: The only FD is due to the primary key. All determinants are candidate keys. Therefore, it is in BCNF.

4NF: There is no observable multi-valued dependency. Therefore, it is in 4NF.

Product (ProductID char(5), Name varchar(40), SellingPrice CURRENCY, ProductType

varchar(30), SupplierID char(5), BranchID char(3))

PK = {ProductID}

FK1 ={SupplierID} references PK of Supplier

FK2 = {BranchID} references PK of Branch

Functional Dependency: ProductID -> Name, Description, SellingPrice, ProductType, SupplierID, BranchID

1NF: There is no repeating group structure in the relation. Therefore, it is in 1NF.

2NF: It is already in 1NF. Primary key has exactly one attribute. Therefore, it is in 2NF.

3NF: It is already in 2NF. There is no transitive functional dependency. Therefore, it is in 3NF.

BCNF: The only FD is due to the primary key. All determinants are candidate keys. Therefore, it is in BCNF.

4NF: There is no observable multi-valued dependency. Therefore, it is in 4NF.

FacilityStaff(EMPID char(5))

PK = {EMPID}

FK = {EMPID} references PK of Employees

Functional Dependency: EMPID->

1NF: There is no repeating group structure in the relation. Therefore, it is in 1NF.

2NF: It is already in 1NF. Primary key has exactly one attribute. Therefore, it is in 2NF.

3NF: It is already in 2NF. FacilityStaff has one attribute only, and it is not possible to see transitive functional dependency. Therefore, it is in 3NF.

BCNF: The only FD is due to the primary key. All determinants are candidate keys. Therefore, it is in BCNF.

4NF: There is no observable multi-valued dependency. Therefore, it is in 4NF.

Sales(SalesID char(5), EMPID char(5), Date date)

PK = {SalesID}

FK ={EMPID} references PK of FacilityStaff

Functional Dependency: SalesID -> EMPID, Date

1NF: There is no repeating group structure in the relation. Therefore, it is in 1NF.

2NF: It is already in 1NF. Primary key has exactly one attribute. Therefore, it is in 2NF.

3NF: It is already in 2NF. There is no transitive functional dependency. Therefore, it is in

3NF.

BCNF: The only FD is due to the primary key. All determinants are candidate keys. Therefore, it is in BCNF.

4NF: There is no observable multi-valued dependency. Therefore, it is in 4NF.

SalesItem (ProductID char(5), SalesID char(5), Quantity INT)

PK = {ProductID, SalesID}

FK1 ={ProductID} references PK of Product

FK2 = {SalesID} references PK of Sales

Functional Dependency: ProductID, SalesID -> Quantity

1NF: There is no repeating group structure in the relation. Therefore, it is in 1NF.

2NF: It is already in 1NF. All non-key attributes depend on the full primary key. Therefore, it is in 2NF.

3NF: It is already in 2NF. There is no transitive functional dependency. Therefore, it is in 3NF.

BCNF: The only FD is due to the primary key. All determinants are candidate keys. Therefore, it is in BCNF.

4NF: There is no observable multi-valued dependency. Therefore, it is in 4NF.

Inventory (InventoryID char(5), ProductID char(5), Quantity_on_hand INT,
Quantity_available INT)

PK = {InventoryID}

FK1 ={ProductID} references PK of Product

Functional Dependency: InventoryID -> ProductID, Quantity_on_hand,
Quantity_available

1NF: There is no repeating group structure in the relation. Therefore, it is in 1NF.

2NF: It is already in 1NF. Primary key has exactly one attribute. Therefore, it is in 2NF.

3NF: It is already in 2NF. There is no transitive functional dependency. Therefore, it is in 3NF.

BCNF: The only FD is due to the primary key. All determinants are candidate keys. Therefore, it is in BCNF.

4NF: There is no observable multi-valued dependency. Therefore, it is in 4NF.

Equipment (EquipmentID char(5), Name varchar(40), Type varchar(30), Description varchar(150), ProductionDate date, EMPID char(5), CostPrice CURRENCY, SupplierID char(5), DateBought date , ServiceLife DOUBLE, BranchID char(3))

PK = {EquipmentID}

FK1 ={EMPID} references PK of FacilityStaff

FK2 ={SupplierID} references PK of Supplier

FK3 = {BranchID} references PK of Branch

Functional Dependency: EquipmentID -> Name, Type, Description, ProductionDate, EMPID, CostPrice, SupplierID, DateBought , ServiceLife, BranchID

1NF: There is no repeating group structure in the relation. Therefore, it is in 1NF.

2NF: It is already in 1NF. Primary key has exactly one attribute. Therefore, it is in 2NF.

3NF: It is already in 2NF. There is no transitive functional dependency. Therefore, it is in 3NF.

BCNF: The only FD is due to the primary key. All determinants are candidate keys. Therefore, it is in BCNF.

4NF: There is no observable multi-valued dependency. Therefore, it is in 4NF.

Maintenance_and_repair_report (TaskID char(5), EquipmentID char(5), EMPID char(5), Date date, Cost CURRENCY, Type varchar(30), Duration DOUBLE)

PK = {TaskID}

FK1 ={EquipmentID} references PK of Equipment

FK2 ={EMPID} references PK of FacilityStaff

Functional Dependency: TaskID -> EquipmentID, EMPID, Date, Cost, Type, Duration

1NF: There is no repeating group structure in the relation. Therefore, it is in 1NF.

2NF: It is already in 1NF. Primary key has exactly one attribute. Therefore, it is in 2NF.

3NF: It is already in 2NF. There is no transitive functional dependency. Therefore, it is in 3NF.

BCNF: The only FD is due to the primary key. All determinants are candidate keys. Therefore, it is in BCNF.

4NF: There is no observable multi-valued dependency. Therefore, it is in 4NF.

Rooms (RoomNo char(3), BranchID char(3))

PK = {RoomNo}

FK={BranchID} references PK of Branch

Functional Dependency: RoomNo -> BranchID

1NF: There is no repeating group structure in the relation. Therefore, it is in 1NF.

2NF: It is already in 1NF. Primary key has exactly one attribute. Therefore, it is in 2NF.

3NF: It is already in 2NF. Rooms has two attributes only, and it is not possible to see transitive functional dependency. Therefore, it is in 3NF.

BCNF: The only FD is due to the primary key. All determinants are candidate keys. Therefore, it is in BCNF.

4NF: There is no observable multi-valued dependency. Therefore, it is in 4NF.

Instructors (EMPID char(5))

PK = {EMPID}

FK = {EMPID} references PK of Employees

Functional Dependency: EMPID ->

1NF: There is no repeating group structure in the relation. Therefore, it is in 1NF.

2NF: It is already in 1NF. Primary key has exactly one attribute. Therefore, it is in 2NF.

3NF: It is already in 2NF. FacilityStaff has one attribute only, and it is not possible to see transitive functional dependency. Therefore, it is in 3NF.

BCNF: The only FD is due to the primary key. All determinants are candidate keys. Therefore, it is in BCNF.

4NF: There is no observable multi-valued dependency. Therefore, it is in 4NF.

Classes (ClassID char(5), ClassName varchar(100), ClassTime varchar(40), RoomNo char(3), ClassSize INT, EMPID char(5))

PK = {ClassID}

FK1 ={EMPID} references PK of Instructors

FK2 ={RoomNo} references PK of Rooms

Functional Dependency: ClassID-> ClassName, ClassTime, RoomNo, ClassSize, EMPID

1NF: There is no repeating group structure in the relation. Therefore, it is in 1NF.

2NF: It is already in 1NF. Primary key has exactly one attribute. Therefore, it is in 2NF.

3NF: It is already in 2NF. There is no transitive functional dependency. Therefore, it is in 3NF.

BCNF: The only FD is due to the primary key. All determinants are candidate keys. Therefore, it is in BCNF.

4NF: There is no observable multi-valued dependency. Therefore, it is in 4NF.

EquipmentAssignment (EquipmentID char(5), ClassID char(5))

PK = {EquipmentID, ClassID}

FK1 ={EquipmentID} references PK of Equipment

FK2 = {ClassID} references PK of Classes

Functional Dependency: EquipmentID, ClassID ->

1NF: There is no repeating group structure in the relation. Therefore, it is in 1NF.

2NF: It is already in 1NF. It's not possible to see partial functional dependency due to that no other functional dependencies exist except the functional dependency for the primary key. Therefore, it is in 2NF.

3NF: It is already in 2NF. EquipmentAssignment has two attributes only, and it is not possible to see transitive functional dependency. Therefore, it is in 3NF.

BCNF: The only FD is due to the primary key. All determinants are candidate keys. Therefore, it is in BCNF.

4NF: There is no observable multi-valued dependency. Therefore, it is in 4NF.

MembershipPlan (Plan varchar(15), Rate CURRENCY)

PK = {Plan}

Functional Dependency: Plan-> Rate

1NF: There is no repeating group structure in the relation. Therefore, it is in 1NF.

2NF: It is already in 1NF. Primary key has exactly one attribute. Therefore, it is in 2NF.

3NF: It is already in 2NF. MembershipPlan has two attributes only, and it is not possible to see transitive functional dependency. Therefore, it is in 3NF.

4NF: There is no observable multi-valued dependency. Therefore, it is in 4NF.

Members (MemberID char(5), FirstName varchar(40), LastName varchar(40) , DOB date, Email varchar(30), PhoneNo char(10), Address varchar(150), Plan varchar(15))

PK = {MemberID}

FK={Plan} references PK of MembershipPlan {Plan}

Functional Dependency: MemberID-> FirstName, LastName, DOB, Email, PhoneNo,

Address, Plan

1NF: There is no repeating group structure in the relation. Therefore, it is in 1NF.

2NF: It is already in 1NF. Primary key has exactly one attribute. Therefore, it is in 2NF.

3NF: It is already in 2NF. There is no transitive functional dependency. Therefore, it is in 3NF.

BCNF: The only FD is due to the primary key. All determinants are candidate keys. Therefore, it is in BCNF.

4NF: There is no observable multi-valued dependency. Therefore, it is in 4NF.

ClassAssignment (ClassID char(5), MemberID char(5))

PK = {ClassID, MemberID}

FK1 = {ClassID} references PK of Classes

FK2 ={MemberID} references PK of Members

Functional Dependency: ClassID, Members->

1NF: There is no repeating group structure in the relation. Therefore, it is in 1NF.

2NF: It is already in 1NF. It's not possible to see partial functional dependency due to that no other functional dependencies exist except the functional dependency for the primary key. Therefore, it is in 2NF.

3NF: It is already in 2NF. ClassAssignment has two attributes only, and it is not possible to see transitive functional dependency. Therefore, it is in 3NF.

BCNF: The only FD is due to the primary key. All determinants are candidate keys. Therefore, it is in BCNF.

4NF: There is no observable multi-valued dependency. Therefore, it is in 4NF.

Trainers (EMPID char(5), Certifications varchar(150))

PK = {EMPID}

FK={EMPID} references PK of Employees

Functional Dependency: EMPID-> Certifications

1NF: There is no repeating group structure in the relation. Therefore, it is in 1NF.

2NF: It is already in 1NF. Primary key has exactly one attribute. Therefore, it is in 2NF.

3NF: It is already in 2NF. Trainers has two attributes only, and it is not possible to see transitive functional dependency. Therefore, it is in 3NF.

BCNF: The only FD is due to the primary key. All determinants are candidate keys.

Therefore, it is in BCNF.

4NF: There is no observable multi-valued dependency. Therefore, it is in 4NF.

TrainerAssignment (RefNo char(5), EMPID char(5), MemberID char(5), MeetingDay date, MeetingTime time)

PK = {RefNo}

FK1 ={EMPID} references PK of Trainers

FK2 = {MemberID} references PK of Members

Functional Dependency: RefNo->MemberID, EMPID, MeetingDay, MeetingTime

1NF: There is no repeating group structure in the relation. Therefore, it is in 1NF.

2NF: It is already in 1NF. Primary key has exactly one attribute. Therefore, it is in 2NF.

3NF: It is already in 2NF. There is no transitive functional dependency. Therefore, it is in 3NF.

BCNF: The only FD is due to the primary key. All determinants are candidate keys. Therefore, it is in BCNF.

4NF: There is no observable multi-valued dependency. Therefore, it is in 4NF.

Dues (PaymentID char(7), Amount CURRENCY, Status varchar(20), DueDate date, MemberID char(5))

PK = {PaymentID}

FK={MemberID} references PK of Members

Functional Dependency: PaymentID-> Amount, Status, DueDate, MemberID

1NF: There is no repeating group structure in the relation. Therefore, it is in 1NF.

2NF: It is already in 1NF. Primary key has exactly one attribute. Therefore, it is in 2NF.

3NF: It is already in 2NF. There is no transitive functional dependency. Therefore, it is in 3NF.

BCNF: The only FD is due to the primary key. All determinants are candidate keys. Therefore, it is in BCNF.

4NF: There is no observable multi-valued dependency. Therefore, it is in 4NF.

---

isManager (EMPID char(5))

PK = {EMPID}

FK1 = {EMPID} references PK of Employees

IV. Database Implementation

The following SQL statements were used to create tables based on the Relational Model. The SQL statements that used to create data record are attached in appendix.

```
CREATE TABLE Branch (
BranchID char(3) PRIMARY KEY,
PhoneNo char(10),
Address varchar (150)
);

CREATE TABLE Employees (
EMPID char(5) PRIMARY KEY,
FirstName varchar(40),
LastName varchar(40),
DOB date,
PhoneNo char(10),
Address varchar(150),
Salary CURRENCY,
BranchID char(3),
FOREIGN KEY(BranchID) references Branch(BranchID)
);

CREATE TABLE isManager (
EMPID char(5) PRIMARY KEY,
FOREIGN KEY(EMPID) REFERENCES Employees(EMPID)
);

CREATE TABLE FacilityStaff (
EMPID char(5) PRIMARY KEY,
FOREIGN KEY(EMPID) REFERENCES Employees(EMPID)
);

CREATE TABLE Instructors (
```

```
EMPID char(5) PRIMARY KEY,
FOREIGN KEY(EMPID) REFERENCES Employees(EMPID)
);

CREATE TABLE Trainers (
EMPID char(5) primary key,
Certifications varchar(150),
FOREIGN KEY(EMPID) REFERENCES Employees(EMPID)
);

CREATE TABLE ClassAssignment (
ClassID char(5),
MemberID char(5),
Primary key(ClassID, MemberID),
foreign key(ClassID) references Classes(ClassID),
foreign key(MemberID) references Members(MemberID)
);

CREATE TABLE Classes (
ClassID char(5) primary key,
ClassName varchar(40),
ClassTime varchar(40),
RoomNo char(3),
ClassSize INT,
EMPID char(5),
foreign key(RoomNo) references Rooms(RoomNo),
foreign key(EMPID) references Instructors(EMPID)
);

CREATE TABLE MembershipPlan (
Plan varchar(15) primary key,
Rate CURRENCY
);
```

```
CREATE TABLE Rooms (
RoomNo char(3) primary key,
BranchID char(3),
foreign key(BranchID) references Branch(BranchID)
);
CREATE TABLE Supplier (
SupplierID char(5) primary key,
Name varchar(40),
PhoneNo char(10),
Address varchar(150),
Email varchar(30)
);

CREATE TABLE Equipment(
EquipmentID char(5) primary key,
Name varchar(40),
Type varchar(30),
Description varchar(150),
ProductionDate date,
EMPID char(5),
CostPrice CURRENCY,
SupplierID char(5),
DateBought date ,
ServiceLife DOUBLE,
BranchID char(3),
foreign key(EMPID) references FacilityStaff(EMPID),
foreign key(SupplierID) references Supplier(SupplierID)
);

CREATE TABLE Maintenance_and_repair_report(
TaskID char(5) primary key,
EquipmentID char(5),
```

```
EMPID char(5),
TaskDate date,
Cost CURRENCY,
TaskType varchar(30),
Duration DOUBLE,
foreign key(EMPID) references FacilityStaff(EMPID),
foreign key(EquipmentID) references Equipment(EquipmentID)
);

CREATE TABLE Members (
MemberID char(5) primary key,
FirstName varchar(40),
LastName varchar(40),
DOB date,
Email varchar(30),
PhoneNo char(10),
Address varchar(150),
Plan varchar(15),
FOREIGN KEY(Plan) references MembershipPlan(Plan)
);

CREATE TABLE Dues(
PaymentID char(7) primary key,
Amount CURRENCY,
Status varchar(20),
DueDate date,
MemberID char(5),
FOREIGN KEY(MemberID) references Members(MemberID)
);

CREATE TABLE TrainerAssignment(
RefNo char(5),
EMPID char(5),
```

```
MemberID char(5),
MeetingDay date,
MeetingTime time,
PRIMARY KEY(RefNo),
FOREIGN KEY(EMPID) references Trainers(EMPID),
FOREIGN KEY(MemberID) references Members(MemberID)
);

CREATE TABLE EquipmentAssignment
(EquipmentID char(5) primary key,
ClassID char(5),
FOREIGN KEY(EquipmentID) REFERENCES Equipment(EquipmentID),
FOREIGN KEY(ClassID) REFERENCES Classes(ClassID)
);

CREATE TABLE Product
(ProductID char(5) PRIMARY KEY,
Name varchar(40),
SellingPrice CURRENCY,
ProductType varchar(30),
SupplierID char(5),
BranchID char(3),
FOREIGN KEY(SupplierID) REFERENCES Supplier(SupplierID),
FOREIGN KEY(BranchID) REFERENCES Branch(BranchID)
);

CREATE TABLE Inventory
(InventoryID char(5) PRIMARY KEY,
ProductID char(5),
Quantity_on_hand INT,
Quantity_available INT,
FOREIGN KEY(ProductID) REFERENCES Product(ProductID)
);
```

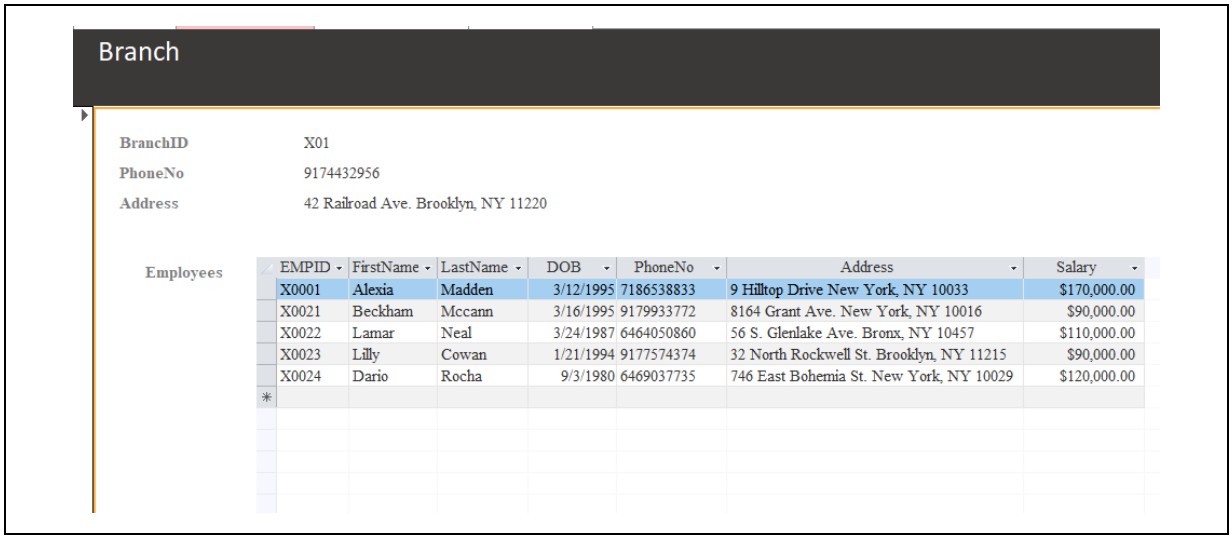
```
CREATE TABLE Sales
(SalesID char(5) PRIMARY KEY,
EMPID char(5),
SalesDate date,
FOREIGN KEY(EMPID) REFERENCES FacilityStaff(EMPID)
);

CREATE TABLE Salesitem
(ProductID char(5),
SalesID char(5),
Quantity INT,
PRIMARY KEY(ProductID, SalesID),
FOREIGN KEY(ProductID) REFERENCES Product(ProductID),
FOREIGN KEY(SalesID) REFERENCES Sales(SalesID)
);

CREATE TABLE tblUser(
Identification_Number varchar(20),
Full_Name char(100),
Password varchar(20)
);
```

V. Forms

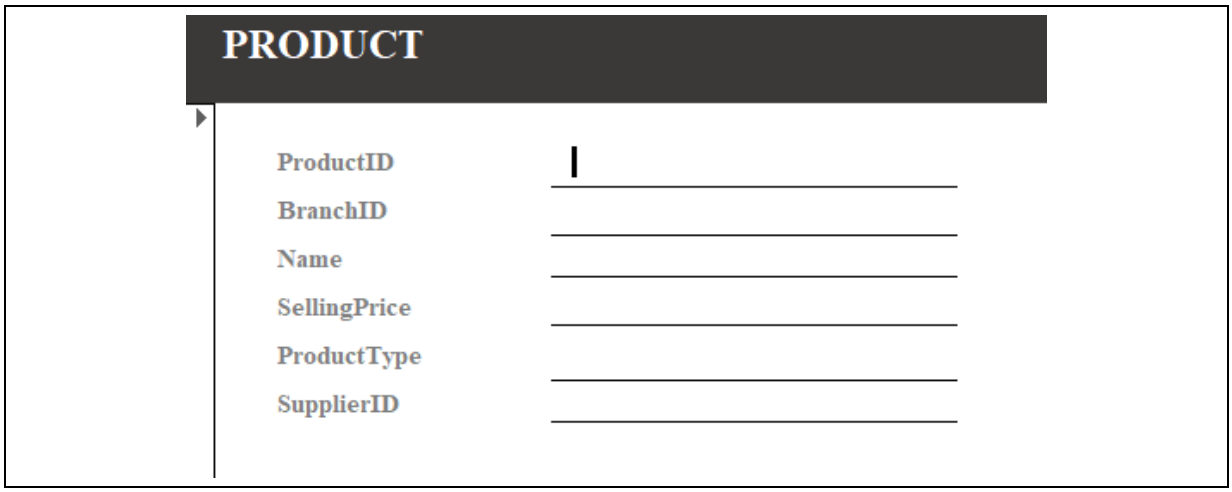
1. Employee form: This form is for Manager to enter new employee information. (More details can be seen in db file).

After entering the general information, Manager should select particular roles to enter more information. For example, Trainers form is as below after selecting Trainer.

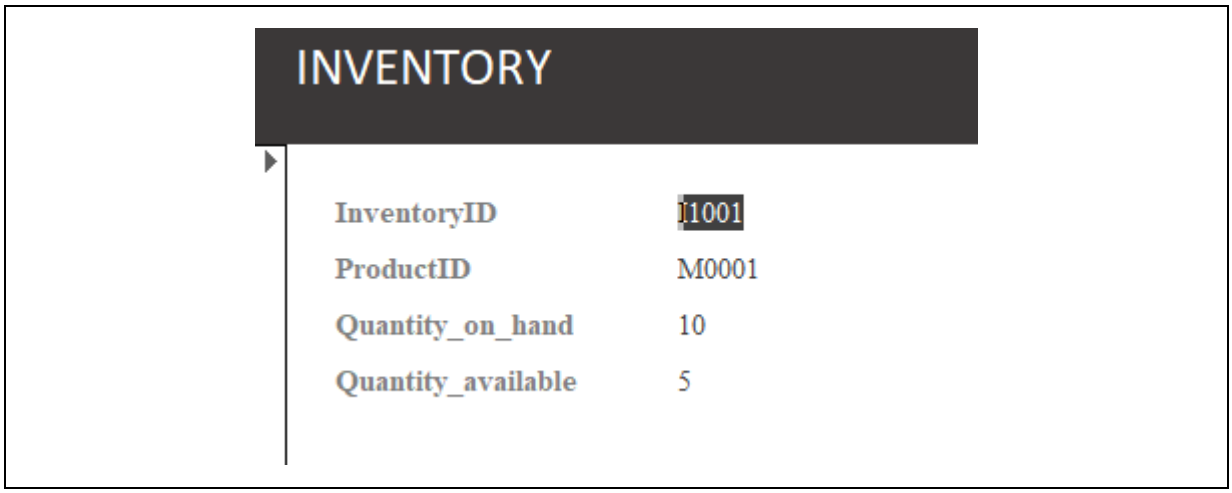
2. Branch Form: Managers can view Branch Information. Employees list appear as sub-table under branch information.



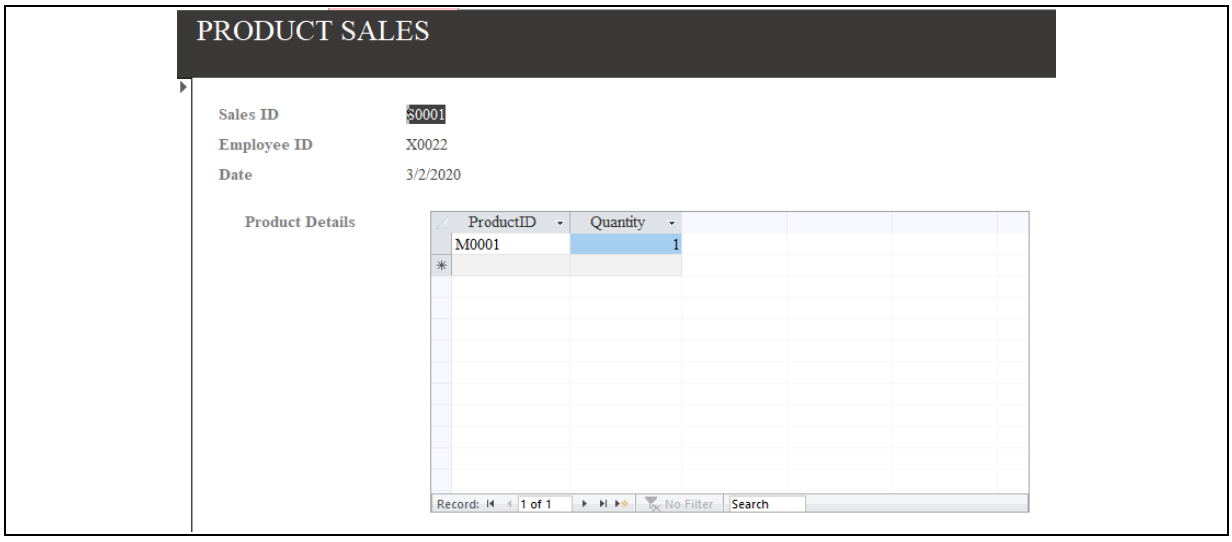
3. Product Form: Facility Staff enter each new product from supplier



4. Inventory Form: Facility Staff record inventory stock number of products

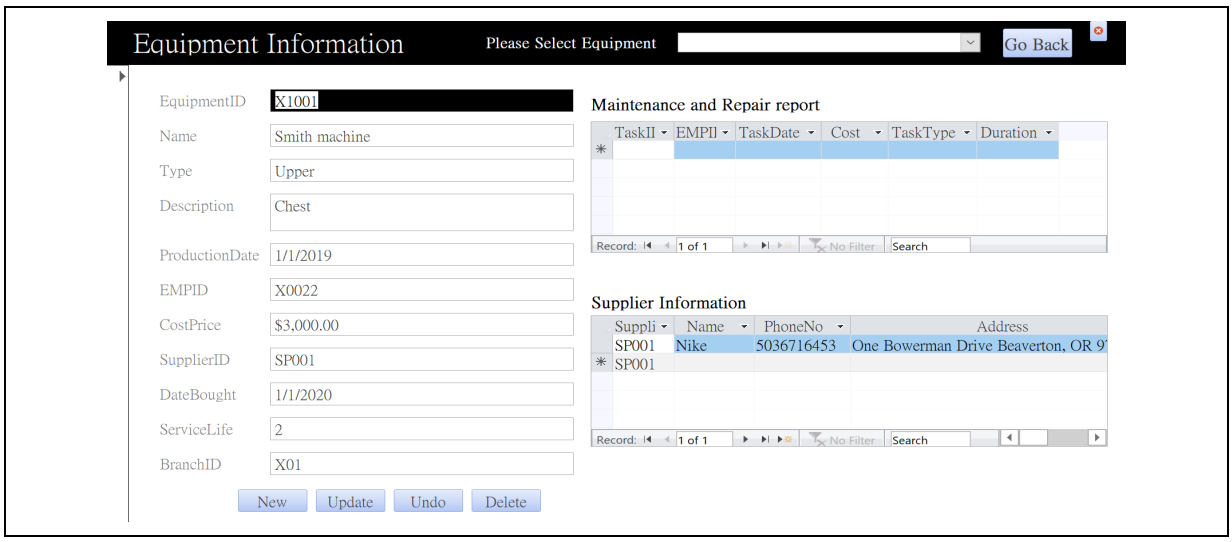


5. Product Sales Form: Once transaction occurs, Facility Staff record each sales information. Product Detail and Quantity is sub-form of this form.

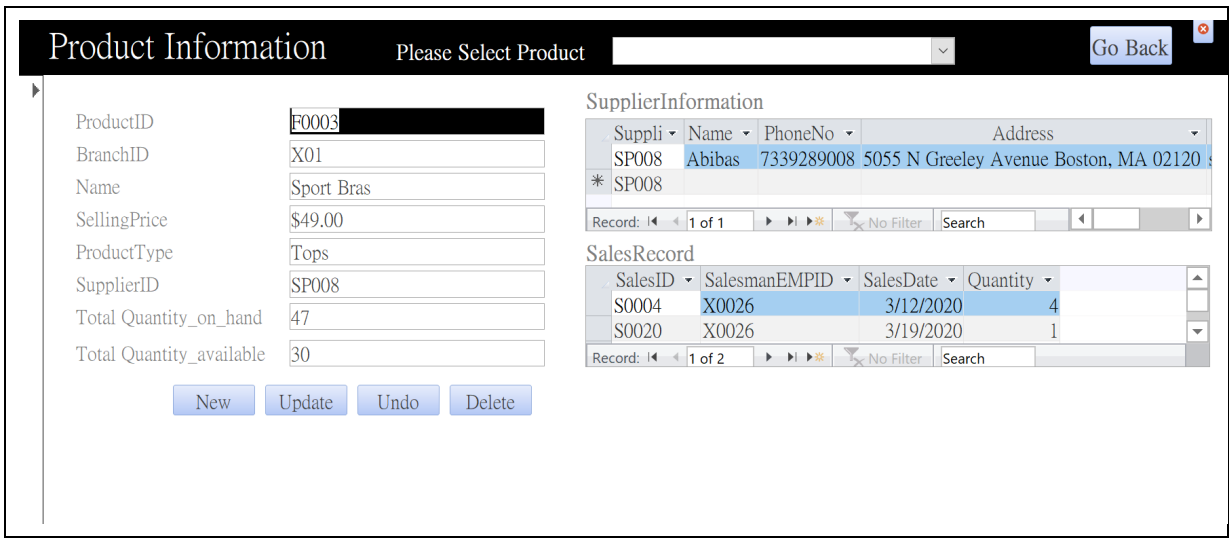


6. Facility Staff Menu form: This form serve as a menu for facility staff to do corresponding command, enter either the equipment management form or product management form.

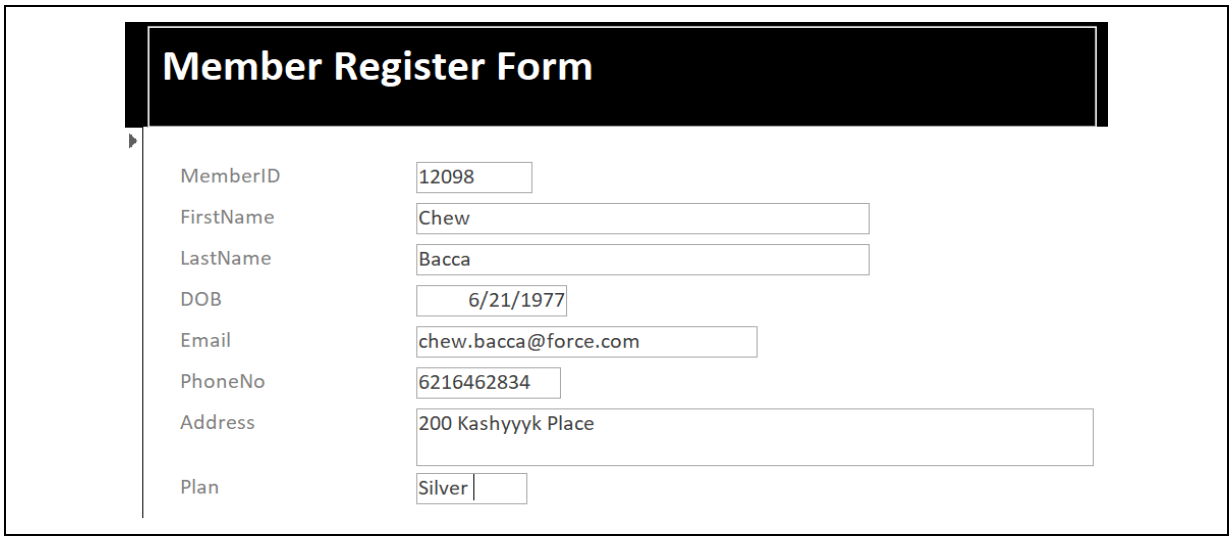
7. Equipment information: This form is for facility staff to manage the basic equipment information, maintenance and repair record and it's supplier's information.



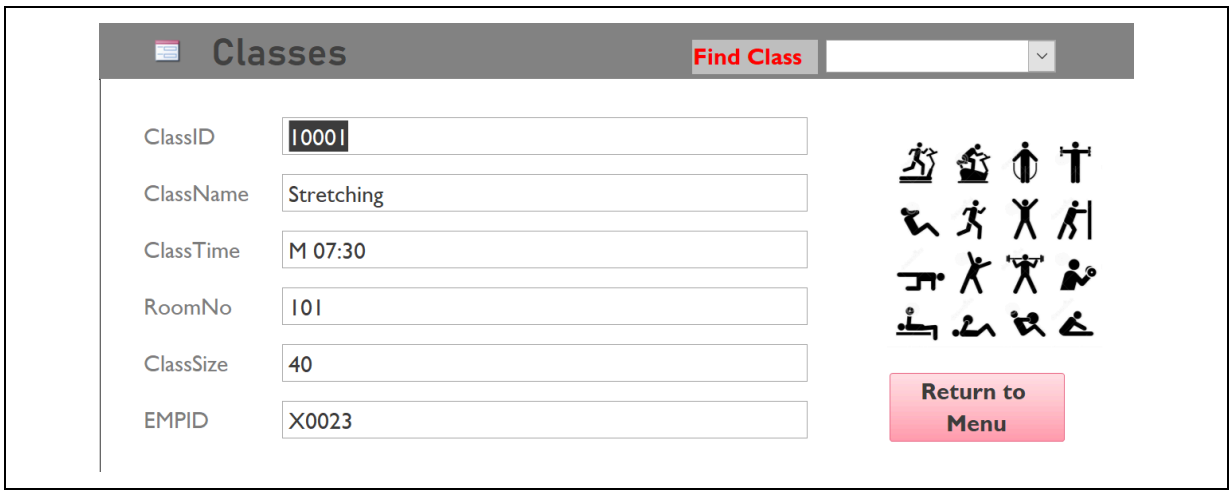
8. Product Information form: This form is for facility staff to manage the basic product information, related sales record and it's supplier's information.



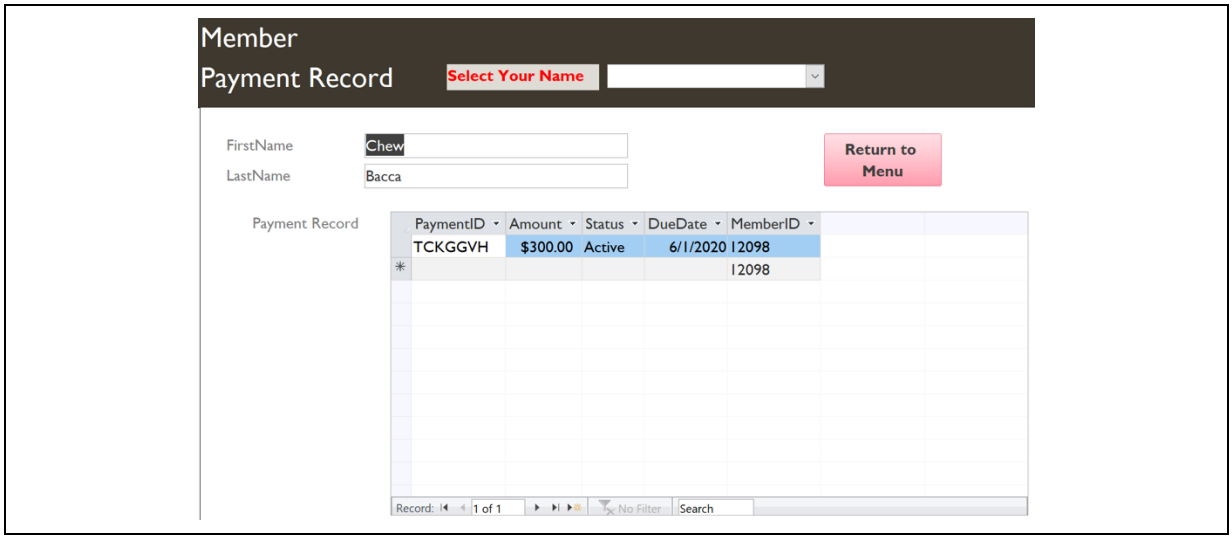
9. Member Register Form: This form is to register new member of XYZ gym



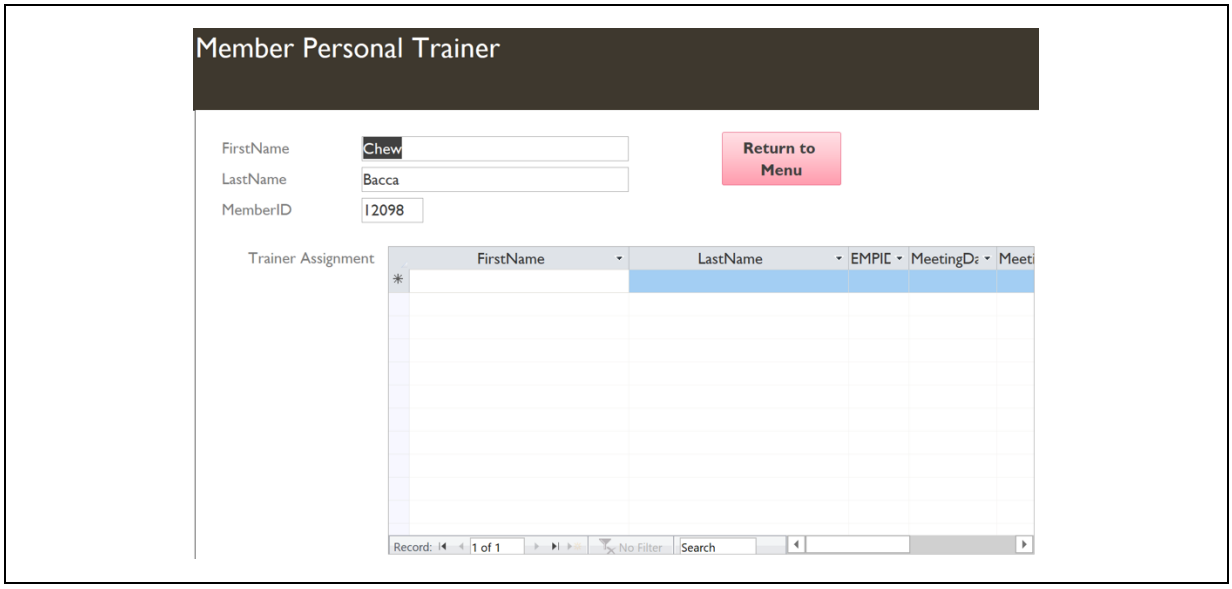
10. Classes Form: This form is for user to view/update classes info.



11. Member Payment Record: These forms is for user to view/update payment record info.



12. Member Personal Trainer Form: These forms is for user to view/update payment personal trainer info.



13. Sub forms and other forms: The sub forms are embed in one/many of the forms above. The other forms serves for the purpose of UI.

VI. Useful SQL Queries

1. XYZ Gym has 20 branches, it is important to know the payroll of all the staff each branch. Managers then could compare the salaries paid to employees with the revenue of every branch to make some adjustment in the operation strategy. Find the total salaries paid of every branch in descending order.

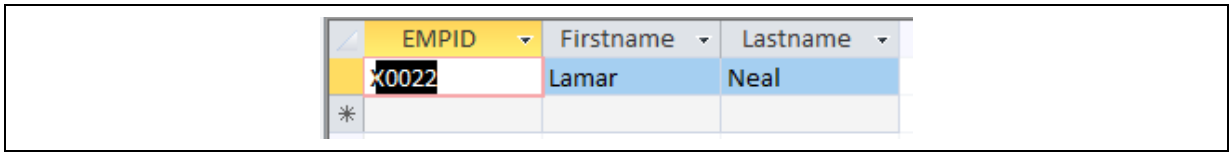
```
SELECT b.BranchID, b.Address, SUM(e.Salary) as Total_Payroll
FROM Branch b, Employees e
WHERE b.BranchID = e.BranchID
GROUP BY b.BranchID, b.Address
ORDER BY SUM(e.Salary) DESC;
```

BranchID	Address	Total_Payrol
X03	18 East Annadale Ave. Brooklyn, NY 11235	$630,000.00
X17	7853 Court St. Bronx, NY 10468	$630,000.00
X06	8518 2nd St. New York, NY 10028	$630,000.00
X07	9 Hilltop Drive New York, NY 10033	$620,000.00
X15	172 Hillcrest Street Brooklyn, NY 11234	$610,000.00
X18	18 1st Court Brooklyn, NY 11218	$590,000.00
X13	680 North Manhattan Lane New York, NY 100	$580,000.00
X01	42 Railroad Ave. Brooklyn, NY 11220	$580,000.00
X04	57 W. Parker St. Bronx, NY 10461	$570,000.00
X12	71 Homewood Dr. Brooklyn, NY 11209	$570,000.00
X05	7994 Grandrose St. New York, NY 10016	$560,000.00
X10	8605 East Nut Swamp St. New York, NY 10027	$550,000.00
X16	973 East Rocky River Street Bronx, NY 10452	$550,000.00
X11	6 Division Avenue Brooklyn, NY 11204	$520,000.00
X19	71 Peachtree Drive Brooklyn, NY 11214	$510,000.00
X08	7633 Lilac Street Brooklyn, NY 11213	$490,000.00
X20	7984 Paris Hill St. Brooklyn, NY 11214	$480,000.00
X09	107 Goldfield St. Bronx, NY 10466	$480,000.00
X14	7495 Smith Store Avenue Bronx, NY 10460	$460,000.00
X02	405 Beacon Court Bronx, NY 10461	$410,000.00

2. Each month, Branch Manager will bonus Facility Staff who has the highest KPI. KPI is evaluated based on the number of transactions within that month. Find the Facility Staff who received bonus in March.

```
SELECT EMPID, Firstname, Lastname
FROM Employees
WHERE EMPID in (
SELECT TOP 1 EMPID FROM Sales
WHERE Month(SalesDate) = 3
```

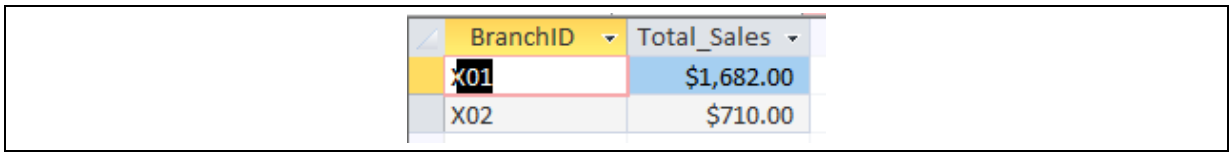
```
GROUP BY EMPID
ORDER BY COUNT(*) DESC);
```



EMPID	Firstname	Lastname
X0022	Lamar	Neal

3. To keep up with increasing consumer demand, XYZ Gym also sell merchandise which include a wide range of products (shorts, shirt, bags, accessories...). This helps to strengthen XYZ gym community and increase revenue. Manager would like to know the total merchandise sales revenue of each branch in March.

```
SELECT p.BranchID, SUM(p.SellingPrice * si.Quantity) as Total_Sales
FROM Salesitem si, Sales s, Product p
WHERE p.ProductID = si.ProductID AND si.SalesID = s.SalesID
AND Month(s.SalesDate) = 3
GROUP BY p.BranchID
ORDER BY 2 DESC;
```



BranchID	Total_Sales
X01	$1,682.00
X02	$710.00

4. Facility Staff are in charge of maintaining product inventory. To ensure that there is no out-of- stock situation, they need to order products from suppliers once the inventory number is fewer than 5 SKU. Find the list of products which need to be ordered.

```
SELECT p.ProductID, p.Name, p.SupplierID
FROM Product p, Inventory i
WHERE p.ProductID = i.ProductID
AND i.Quantity_available < 5;
```

ProductID	Name	SupplierID
F0006	Running Short	SP010
M0005	Jogger	SP010
M0007	Superman hat	SP011
M0015	A-Frame Hat	SP014
M0013	Muscle Nation Bum Bag	SP003
M0002	FKN Gym Wear Gym Bag, Grey	SP001

5. Find the top sales products in March.

```
SELECT p.ProductID, SUM(p.SellingPrice * si.Quantity) AS Total_Sales
FROM Salesitem AS si, Sales AS s, Product AS p
WHERE p.ProductID = si.ProductID AND si.SalesID = s.SalesID
AND Month(s.SalesDate) = 3
GROUP BY p.ProductID
ORDER BY 2 DESC;
```

ProductID	Total_Sales
F0021	$534.00
M0001	$350.00
F0009	$316.00
F0003	$245.00
M0012	$237.00
M0002	$190.00
M0013	$79.00
M0005	$79.00
F0020	$79.00
M0008	$59.00
M0004	$59.00
F0022	$58.00
M0015	$39.00
M0007	$39.00
F0011	$29.00

6. XYZ gym is now planning the budget of the next year. They would like to know the equipment's ID number, name, date of bought, cost price, service life, branch ID and the supplier's name which the equipment will end service life in the end of 2021.

```
SELECT e.EquipmentID, e.Name, e.DateBought, e.CostPrice, e.Servicelife,
e.BranchID, s.Name AS Supplier_Name
FROM equipment e, supplier s
WHERE e.SupplierID = s.SupplierID
AND e.Servicelife <=2;
```

Equipmer	Name	DateBoug	CostPrice	Servicelif	BranchID	Supplier_
X1001	Smith mach	1/1/2020	$3,000.00	2	X01	Nike
X2001	Smith mach	1/1/2020	$3,000.00	2	X02	S.B.D
X3001	Smith mach	1/1/2020	$3,000.00	2	X03	AWA
X4001	Smith mach	1/1/2020	$3,000.00	2	X04	Jump

7. The scale of facilities are not all the same in each branch. It seems that to charge the same rate for each branch is not quite appropriate. XYZ gym would like to make an adjustment on the membership plan fee based on the number and total cost of equipment for each branch.

```
SELECT e.BranchID, COUNT(e.EquipmentID) AS Number_of_facilities,
SUM(e.CostPrice) AS Total_eq_cost
FROM equipment e
GROUP BY e.BranchID;
```

BranchID	Number_of_facilities	Total_eq_cost
X01	5	$13,000.00
X02	5	$13,000.00
X03	5	$13,000.00
X04	3	$7,000.00
X05	1	$3,000.00
X06	1	$3,000.00

8. The spinning bike is a popular facility in branch X01. Nonetheless, it has been broken twice so far in this year. The branch manager is considering to change the supplier of the spinning bike. He want to know the maintenance and repair record, and the supplier information of spinning bike for all branch had this facility.

```
SELECT e.EquipmentID, e.Name, e.BranchID, m.TaskType, m.TaskDate,
m.Duration, m.Cost, s.Name AS Supplier_Name
FROM Equipment e, Maintenance_and_repair_report m, Supplier s
WHERE e.EquipmentID = m.EquipmentID
AND e.SupplierID = s.SupplierID
AND e.Name = 'Spinning bike';
```

Equipmer	Name	BranchID	TaskType	TaskDate	Duration	Cost	Supplier_
X1004	Spinning bi	X01	Maintenanc	2/1/2020	1	$10.00	Puma
X1004	Spinning bi	X01	Repair	3/1/2020	7	$100.00	Puma
X1004	Spinning bi	X01	Repair	5/1/2020	3	$10.00	Puma
X2004	Spinning bi	X02	Maintenanc	3/1/2020	1	$10.00	Kobeer
X2004	Spinning bi	X02	Maintenanc	5/1/2020	1	$10.00	Kobeer
X3004	Spinning bi	X03	Maintenanc	2/1/2020	1	$10.00	Seaka
X5001	Spinning bi	X05	Maintenanc	2/1/2020	1	$10.00	Hokaa
X5001	Spinning bi	X05	Maintenanc	3/1/2020	1	$10.00	Hokaa

9. In order to bargain the equipment price with the supplier in the next time, XYZ gym would like to know the total number and total cost of equipment from each supplier.

```
SELECT  s.Name AS Supplier_Name, COUNT(e.SupplierID) AS Total_Num,
SUM(CostPrice) AS Amount
FROM Equipment e, Supplier s
WHERE e.SupplierID = s.SupplierID
GROUP by s.Name
ORDER by SUM(CostPrice) DESC;
```

Supplier_	Total_Nu	Amount
Jump	3	$7,000.00
Hokaa	2	$6,000.00
S.B.D	1	$3,000.00
Puma	1	$3,000.00
Nike	1	$3,000.00
U.A	1	$3,000.00
Seaka	1	$3,000.00
Bomu	1	$3,000.00
AWA	1	$3,000.00
Amup	1	$3,000.00
Kobeer	1	$3,000.00
Reebok	1	$2,000.00
NARP	1	$2,000.00
Abibas	1	$2,000.00
Ekin	1	$2,000.00
Bsman	1	$2,000.00
Adidas	1	$2,000.00

10. Membership is the most important aspect of our gym operation. Understanding how much we are spending on each customer is vital to analyze sales and costs spending. Find sum of salary resources on member who has ID of 82348.

```
SELECT sum(salary) as sums
FROM employees as e, trainers as t, trainerassignment as ta
WHERE e.empid = t.empid and t.empid = ta.empid and memberid = '82348';
```

sums
$450,000.00

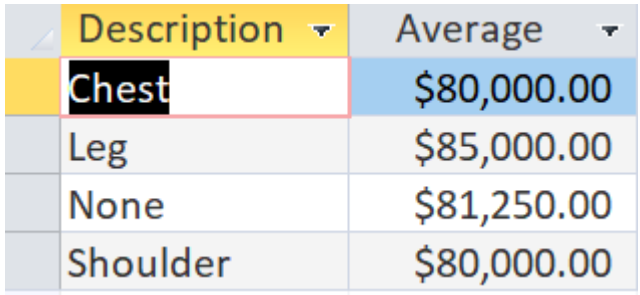
11. XYZ Gym has more than 10 membership plan types. Manager would like to know the revenue from each membership plan for marketing strategy.

```
SELECT mp.plan, Sum(mp.Rate)AS SumOfRate
FROM members AS m, membershipplan AS mp
WHERE (((m.plan)=[mp].[plan]))
GROUP BY mp.plan;
```

plan	SumOfRate
Bronze A	$10.00
Bronze B	$30.00
Bronze D	$25.00
Diamond A	$130.00
Diamond B	$135.00
Gold A	$70.00
Gold B	$150.00
Gold C	$80.00
Gold D	$85.00
Platinum A	$100.00
Platinum C	$110.00
Silver A	$80.00
Silver B	$45.00
Silver C	$150.00
Silver D	$55.00

12. Understanding which equipment classes are expensive helps us to analyze employees' salary assigned for each type of classes. Find average employee salaries by equipment type.

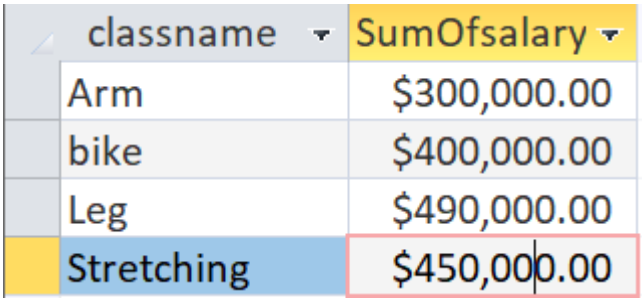
```
SELECT eq.Description, Avg(e.salary) AS Average
FROM employees AS e, classes AS c, equipmentassignment AS ea,
equipment AS eq
WHERE (((e.empid)=[c].[empid]) AND ((c.classid)=[ea].[classid]) AND
((eq.equipmentid)=[ea].[equipmentid]))
GROUP BY eq.Description;
```



Description	Average
Chest	$80,000.00
Leg	$85,000.00
None	$81,250.00
Shoulder	$80,000.00

13. Find salary of employee by the class.

```
SELECT c.classname, Sum(e.salary) AS SumOfsalary
FROM classes AS c, employees AS e
WHERE (((c.empid)=[e].[empid]))
GROUP BY c.classname;
```



classname	SumOfsalary
Arm	$300,000.00
bike	$400,000.00
Leg	$490,000.00
Stretching	$450,000.00

VII. Useful Reports

1. Employees Report: This is an Employees report that essentially shows all information of each staff alphabetically by last name, group by Branch. This report is used by Manager.

EMPLOYEE INFORMATION REPORT 5/19/2020

BranchID	EMPID	FirstName	LastName	DOB	PhoneNo	Address	Salary
X02							
	X0027	Aditya	Kim	4/20/1997	7186292284	80 Bay Meadows Lane Bronx, NY 10465	$60,000.00
X12							
	X0068	Adonis	Cohen	1/8/1990	7181208237	252 Olive Ave. Brooklyn, NY 11228	$120,000.00
X19							
	X0019	Alanna	Romero	3/31/1991	9174842506	203 Lees Creek Lane Brooklyn, NY 11203	$100,000.00
	X0094	Alanna	Sanchez	12/21/1995	6468464835	55 Princess Drive Brooklyn, NY 11233	$110,000.00
X01							
	X0001	Alexia	Madden	3/12/1995	7186538833	9 Hilltop Drive New York, NY 10033	$170,000.00
X18							
	X0090	Alexzander	Jones	9/23/1996	6465868835	145 School Dr. Brooklyn, NY 11219	$100,000.00

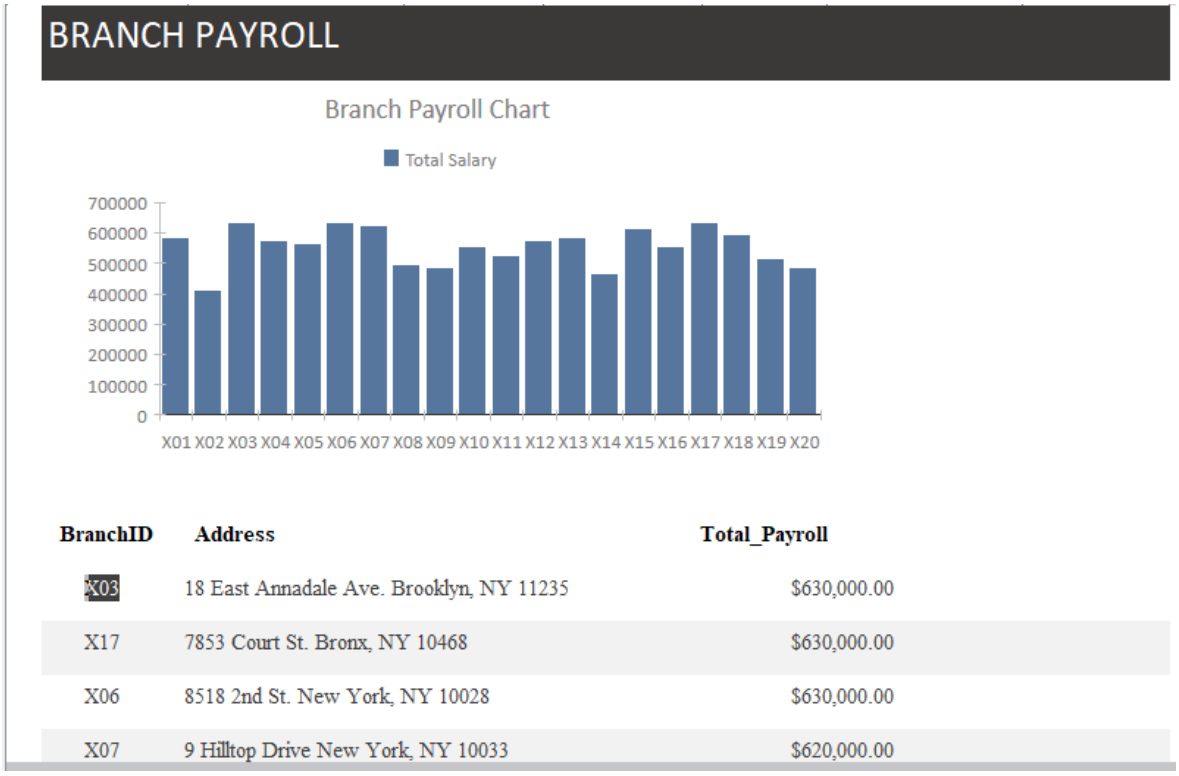
2. Facility Staff Performance Report: The KPI Report of Facility Staff help Manager know the performance of facility staff based on their numbers of transactions. The report shows top 5 staff who have highest performance in March.

Facility Staff Performance

EMPID	Firstname	Lastname	Salary	BranchID
X0022	Lamar	Neal	$110,000.00	X01
X0026	Erika	Saunders	$80,000.00	X02
X0030	Kaitlin	Galvan	$110,000.00	X03
X0034	Kristopher	Aguilar	$110,000.00	X04
X0038	Jonathon	Thomas	$130,000.00	X05
X0042	Tanya	Parrish	$80,000.00	X06

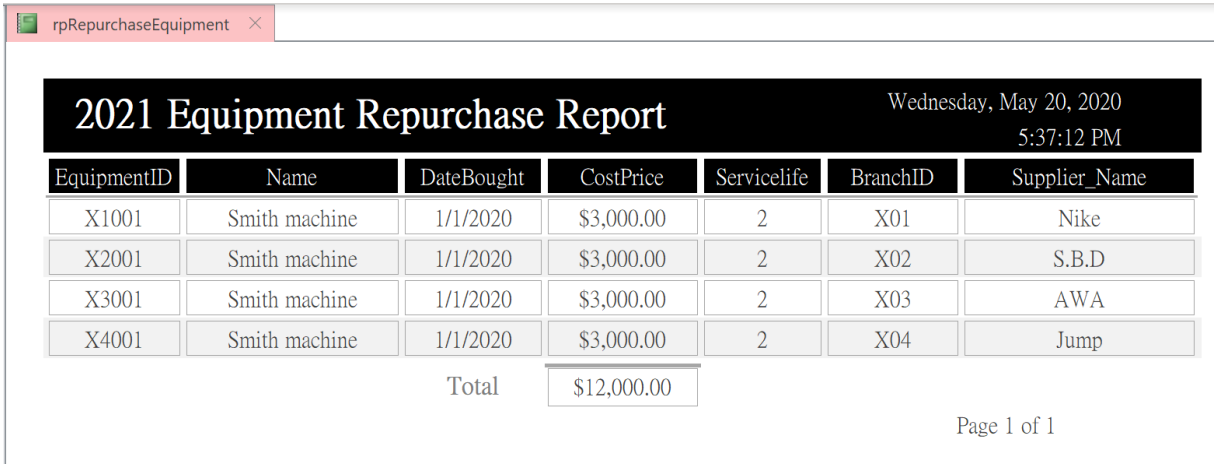
5/19/2020 Page 1 of 1

3. Branch Payroll: Manager use the report to know the total payroll of all the staff each branch to compare the salaries paid to employees of every branch.



BranchID	Address	Total_Payroll
X03	18 East Annadale Ave. Brooklyn, NY 11235	$630,000.00
X17	7853 Court St. Bronx, NY 10468	$630,000.00
X06	8518 2nd St. New York, NY 10028	$630,000.00
X07	9 Hilltop Drive New York, NY 10033	$620,000.00

4. Equipment Repurchase Report : This report is for the branch manager that they can know which equipment will end service life in the end of 2021 and they need to make the budget for repurchase purpose on the budget plan of the next year.



EquipmentID	Name	DateBought	CostPrice	Servicelife	BranchID	Supplier_Name
X1001	Smith machine	1/1/2020	$3,000.00	2	X01	Nike
X2001	Smith machine	1/1/2020	$3,000.00	2	X02	S.B.D
X3001	Smith machine	1/1/2020	$3,000.00	2	X03	AWA
X4001	Smith machine	1/1/2020	$3,000.00	2	X04	Jump
		Total	$12,000.00			

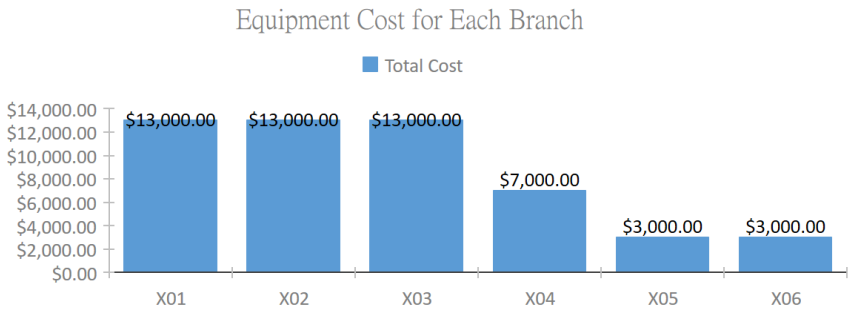
5. Branch Equipment Cost Report : This report helps the manager to make adjustment on the membership plan fee based on the total equipment cost for each branch.

rpBranchEqTotalCost

Branch Equipment Cost Report

Wednesday, May 20, 2020
5:37:33 PM

BranchID	Number of facilities	Total Cost
X01	5	$13,000.00
X02	5	$13,000.00
X03	5	$13,000.00
X04	3	$7,000.00
X05	1	$3,000.00
X06	1	$3,000.00
Total		$52,000.00



6. Branch SP Bike Service Report : This report is for the branch manager to know the service condition of spinning bike from different supplier in each branch, which helps him/her to avoid to purchase this type of equipment from the supplier with a high defective product rate again.

rpBranchSPbikeRecord

Branch SP Bike Service Report

Wednesday, May 20, 2020
5:38:07 PM

EquipmentID	Name	BranchID	TaskType	TaskDate	Duration	Cost	Supplier_Name
X1004	Spinning bike	X01	Maintenance	2/1/2020	1	$10.00	Puma
X1004	Spinning bike	X01	Repair	3/1/2020	7	$100.00	Puma
X1004	Spinning bike	X01	Repair	5/1/2020	3	$10.00	Puma
X2004	Spinning bike	X02	Maintenance	3/1/2020	1	$10.00	Kobeer
X2004	Spinning bike	X02	Maintenance	5/1/2020	1	$10.00	Kobeer
X3004	Spinning bike	X03	Maintenance	2/1/2020	1	$10.00	Seaka
X5001	Spinning bike	X05	Maintenance	2/1/2020	1	$10.00	Hokaa
X5001	Spinning bike	X05	Maintenance	3/1/2020	1	$10.00	Hokaa
					Total	$170.00	

Page 1 of 1

7. Supplier Share Report : This report helps the manager to know the supplier share of the equipment for all branch. Thus, they can bargain the equipment price with the supplier in the

next time if they have already bought lots of equipment from this supplier.

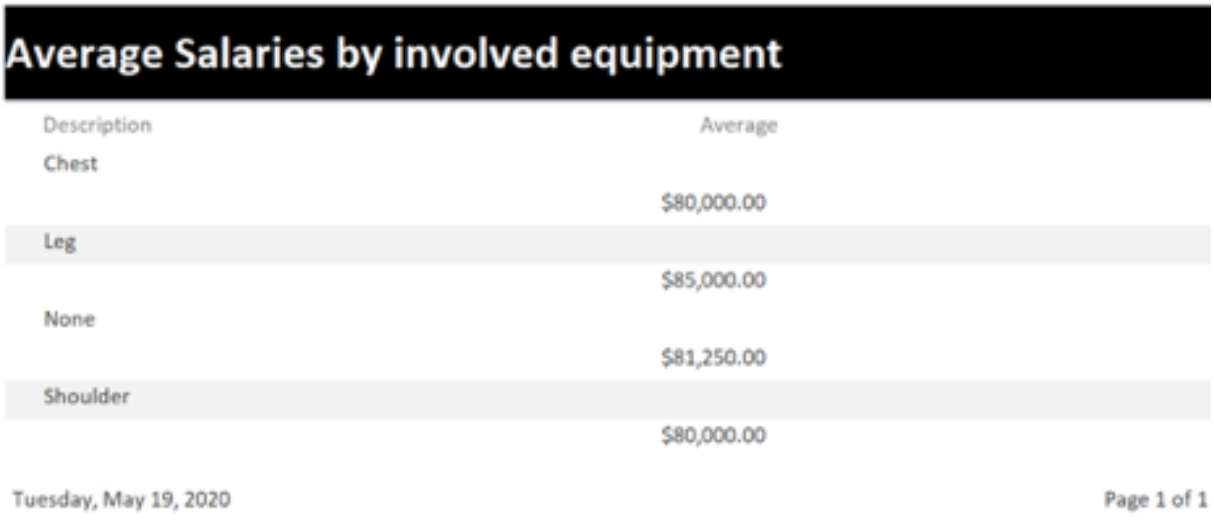
rpSupplierShare

Supplier Share Report

Wednesday, May 20, 2020
5:39:33 PM

Supplier_Name	Total_Num	Amount
Abibas	1	$2,000.00
Adidas	1	$2,000.00
Amup	1	$3,000.00
AWA	1	$3,000.00
Bomu	1	$3,000.00
Bsman	1	$2,000.00
Ekin	1	$2,000.00
Hokaa	2	$6,000.00
Jump	3	$7,000.00
Kobeer	1	$3,000.00
NARP	1	$2,000.00
Nike	1	$3,000.00
Puma	1	$3,000.00
Reebok	1	$2,000.00
S.B.D	1	$3,000.00
Seaka	1	$3,000.00

8. Average Salaries by equipment type report. The report helps to give insights on understanding which equipment classes are expensive.



Average Salaries by involved equipment

Description	Average
Chest	
	$80,000.00
Leg	
	$85,000.00
None	
	$81,250.00
Shoulder	
	$80,000.00

Tuesday, May 19, 2020 Page 1 of 1

9. Salary spending by class type report

Salary Spending by Class

classname	SalarySpending
Arm	
	$300,000.00
bike	
	$400,000.00
Leg	
	$490,000.00
Stretching	
	$450,000.00

Tuesday, May 19, 2020 Page 1 of 1

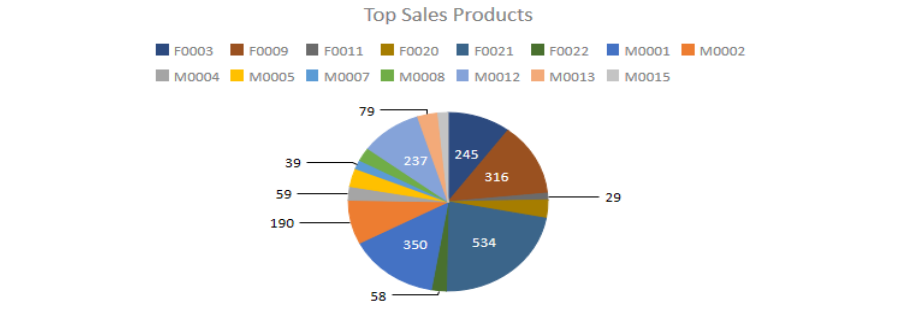
10. Revenue Report from membership plan report.

Revenue Report

plan	SumOfRate
Bronze A	$10.00
Bronze B	$30.00
Bronze D	$25.00
Diamond A	$130.00
Diamond B	$135.00
Gold A	$70.00
Gold B	$150.00
Gold C	$80.00
Gold D	$85.00
Platinum A	$100.00
Platinum C	$110.00
Silver A	$80.00

11. Merchandise report: Managers use the report to know the total merchandise sales revenue of each branch as well as top sales products.

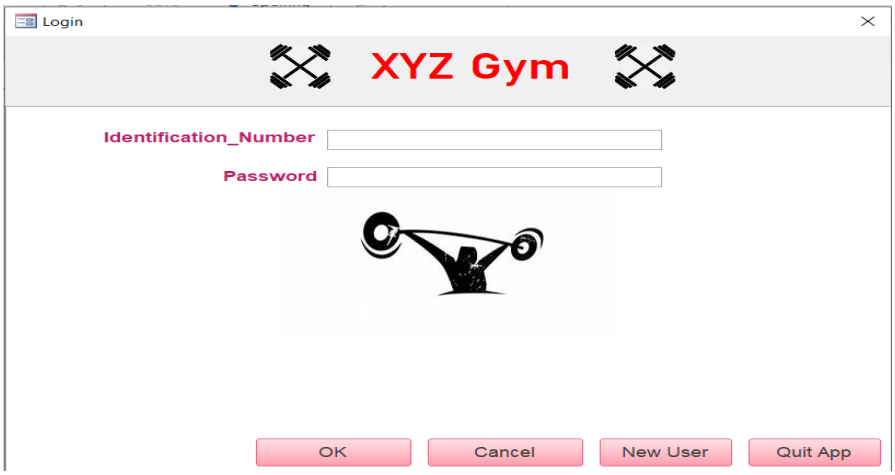
BRANCH MERCHANDISE REVENUE



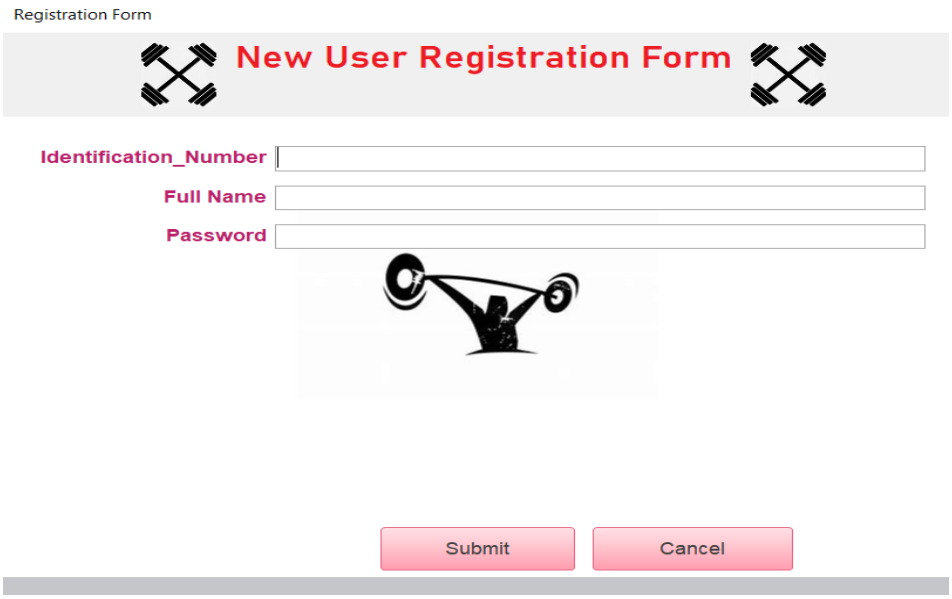
Branch ID	Total Sales
X01	
	$1,682.00
X02	
	$710.00

VIII. User Interface

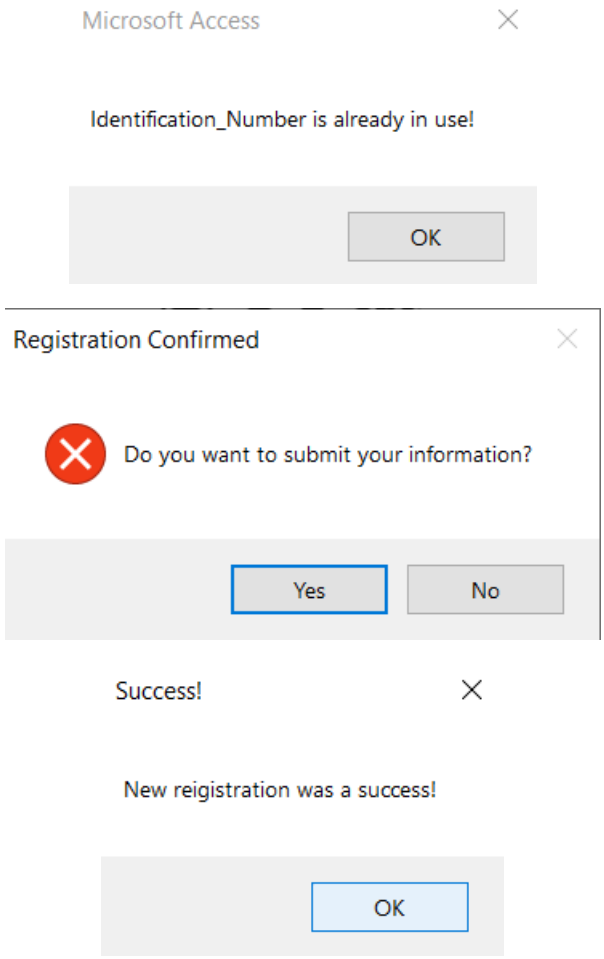
When a user opens the database, he/she is greeted with the log-in menu.



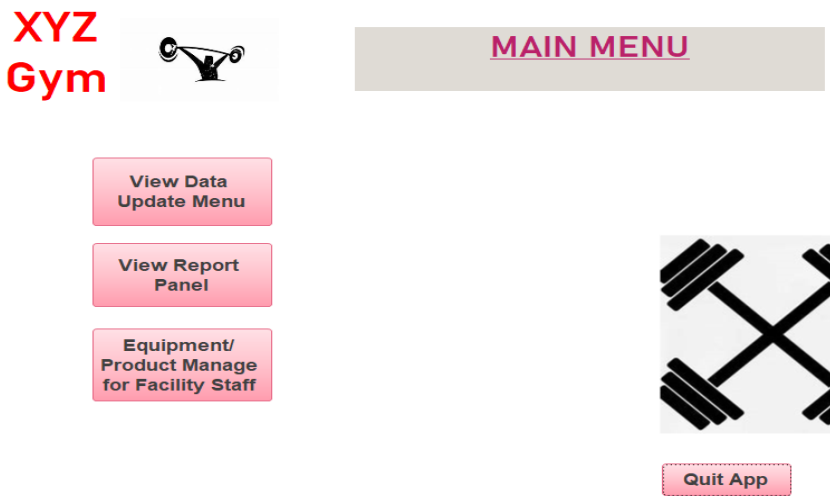
A new user can click on the "New User" button which will navigate them to a registration window.



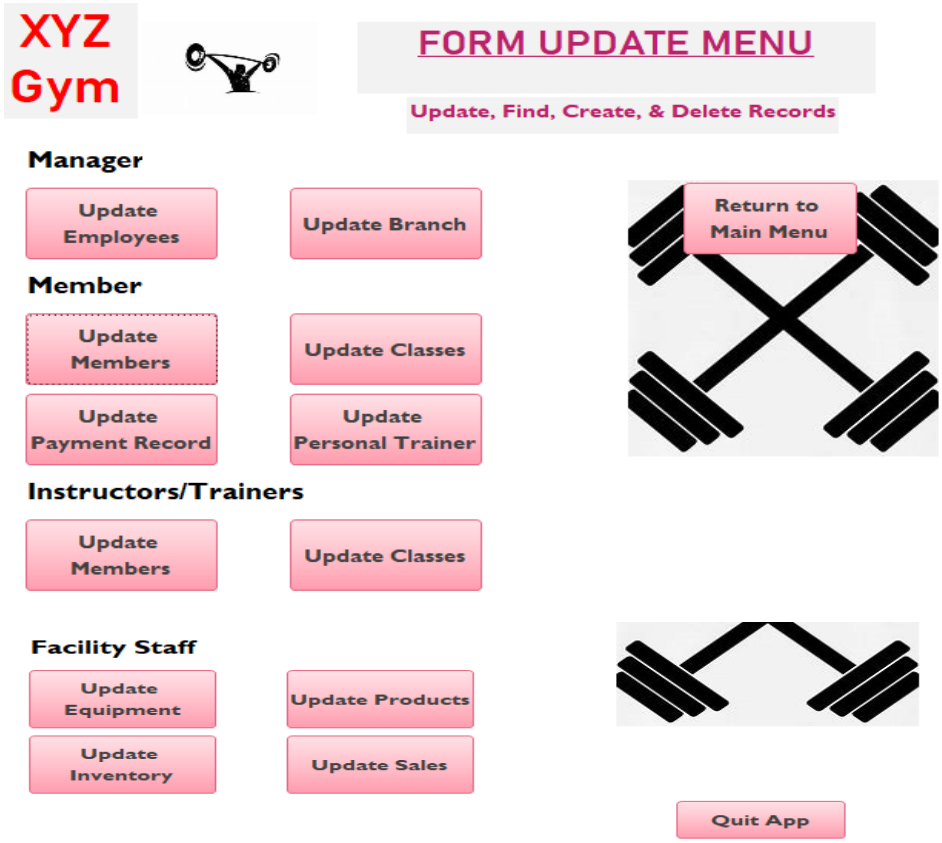
If the user tries to register with an ID Number already in use, the system will warn them and allow them to try again. Otherwise, once "Submit" is clicked, the system will ask the user if they want to confirm their registration and then notify them that their profile was successfully create.



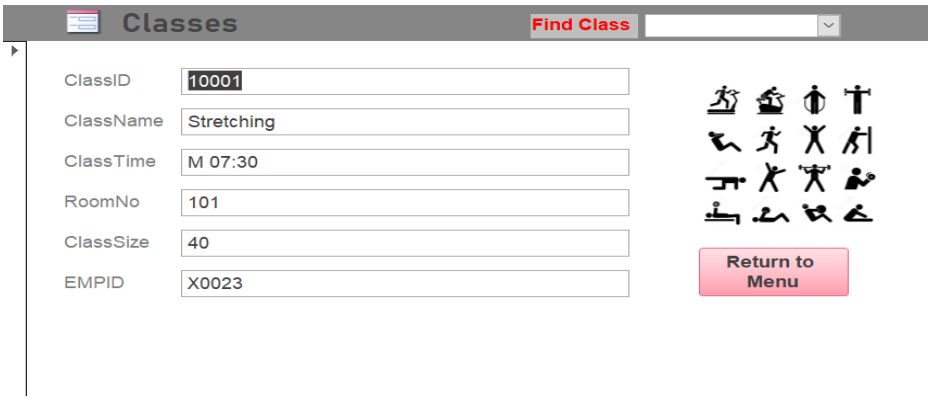
Once the user successfully logs in, they will be directed to the Main Menu



If the user clicks on "View Data Update Menu," they will navigate to the following screen:

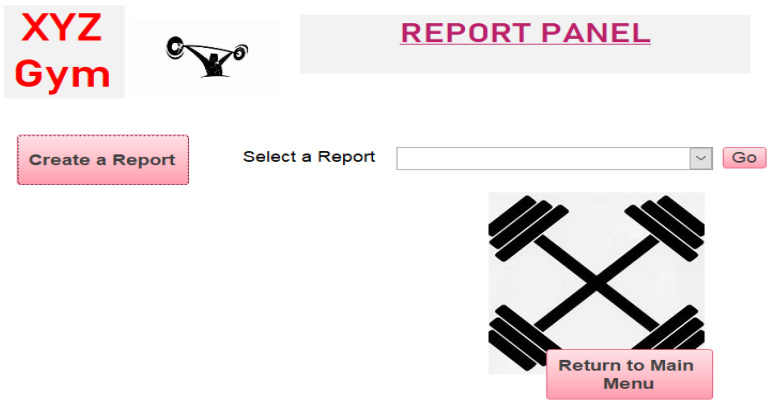


Clicking on any of the “Update” buttons will navigate the user to the named form where the user can edit. For example, the "Update Classes":

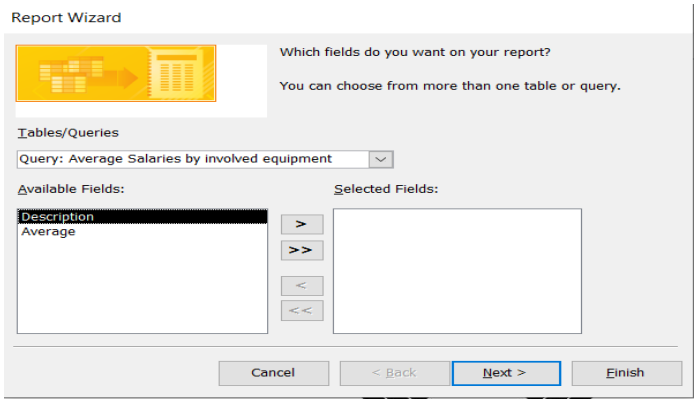


The “Quit App” button will close the database. The “Return to Menu” button will return the user to the “Form Update Menu.” The “Return to Main Menu” button will return the user to the Main Menu.

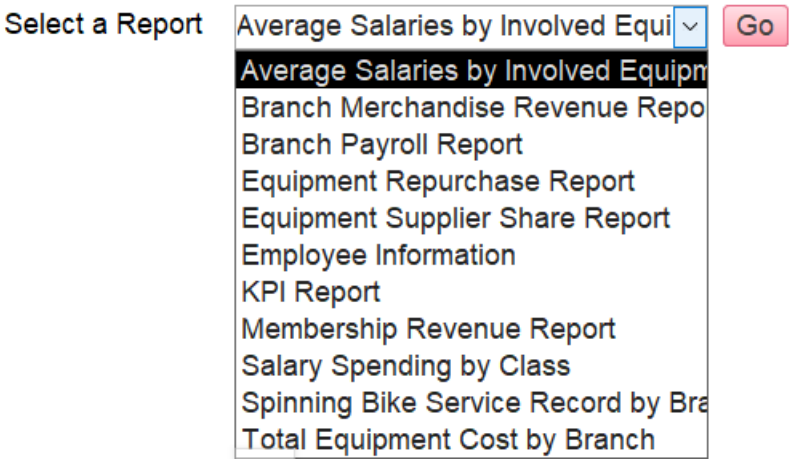
If the user instead clicks on “View Report Panel” from the Main Menu, they will navigate to the following screen:



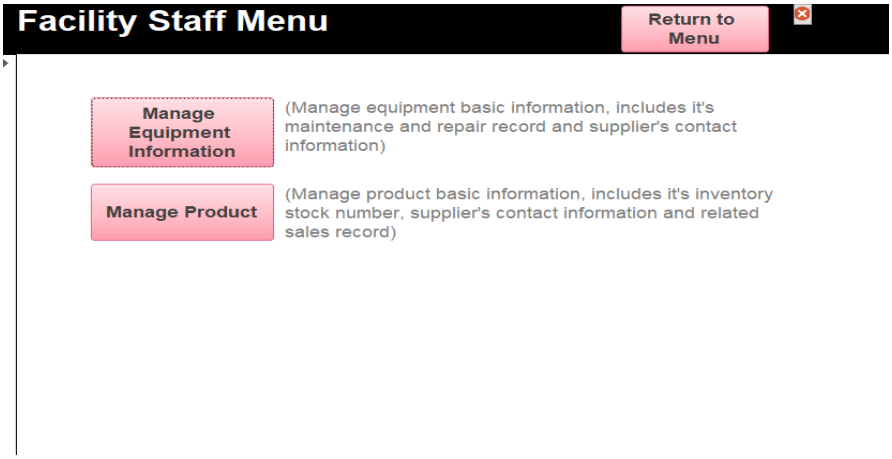
The "Create a Report" button will open the Report Wizard



"Select a Report" features a drop-down menu where the user can select any report in the database. Once a user selects a report and clicks "Go", the database will open up the report.



If the user instead clicks on "Equipment/Product Manager for Facility Staff" from the Main Menu, they will navigate to the following screen:



The "Facility Staff Menu" is intended for easy navigation to items facility staff members frequently access. Clicking on either of the two buttons will open the relevant forms and allow for updates.

IX. Conclusion

We've generated forms, 13 queries and 11 reports based on the implemented database. These forms, queries and reports will make it easier to store data into the database and to analyze critical information for the XYZ gym's new management by allowing them to utilize those data and accessing into the database easily. Also, managers, members, instructors/trainers and facility staff will be able to improve the efficiency of daily operations by having better understanding of the gym's chain.

Microsoft Access is effective in both database constructing and application layer development. One drawback our group encountered is that 1) it can only process one query at a time; 2) it's difficult for collaborating on the tool at the same time. Overall, transition to a system based database project helps us to apply what we learn from school to a real life project so we learn very much from this project.

X. Appendix

A. Data input

```
INSERT INTO Branch VALUES('X01','9174432956','42 Railroad Ave.
Brooklyn, NY 11220');
INSERT INTO Branch VALUES('X02','6462690129','405 Beacon Court Bronx,
NY 10461');
INSERT INTO Branch VALUES('X03','6468036963','18 East Annadale Ave.
Brooklyn, NY 11235');
INSERT INTO Branch VALUES('X04','7187176910','57 W. Parker St. Bronx,
NY 10461');
INSERT INTO Branch VALUES('X05','6462012292','7994 Grandrose St. New
York, NY 10016');
INSERT INTO Branch VALUES('X06','7184683245','8518 2nd St. New York, NY
10028');
INSERT INTO Branch VALUES('X07','9176623708','9 Hilltop Drive New York,
NY 10033');
INSERT INTO Branch VALUES('X08','9176505353','7633 Lilac Street
Brooklyn, NY 11213');
INSERT INTO Branch VALUES('X09','6463624484','107 Goldfield St. Bronx,
NY 10466');
INSERT INTO Branch VALUES('X10','7185255730','8605 East Nut Swamp St.
New York, NY 10027');
INSERT INTO Branch VALUES('X11','9179595820','6 Division Avenue
Brooklyn, NY 11204');
INSERT INTO Branch VALUES('X12','7184636624','71 Homewood Dr. Brooklyn,
NY 11209');
INSERT INTO Branch VALUES('X13','9177506689','680 North Manhattan Lane
New York, NY 10040');
INSERT INTO Branch VALUES('X14','9178725554','7495 Smith Store Avenue
Bronx, NY 10460');
INSERT INTO Branch VALUES('X15','7188833932','172 Hillcrest Street
Brooklyn, NY 11234');
INSERT INTO Branch VALUES('X16','9174605397','973 East Rocky River
Street Bronx, NY 10452');
INSERT INTO Branch VALUES('X17','6468611433','7853 Court St. Bronx, NY
10468');
INSERT INTO Branch VALUES('X18','7187843850','18 1st Court Brooklyn, NY
```

```
11218');

INSERT INTO Branch VALUES('X19','9179290790','71 Peachtree Drive
Brooklyn, NY 11214');
INSERT INTO Branch VALUES('X20','9173401272','7984 Paris Hill St.
Brooklyn, NY 11214');

INSERT INTO Employees
VALUES('X0001','Alexia','Madden','03/12/1995','7186538833','9 Hilltop
Drive New York, NY 10033','170000','X01');
INSERT INTO Employees
VALUES('X0002','Desirae','Ewing','07/20/1982','9176380989','426 Ocean
Rd. Brooklyn, NY 11230','100000','X02');
INSERT INTO Employees
VALUES('X0003','Fernanda','Dorsey','10/19/1982','9174096499','8226
Devon Court Brooklyn, NY 11235','180000','X03');
INSERT INTO Employees
VALUES('X0004','Caden','Watkins','08/14/1990','7182852748','141 West
Parker Street New York, NY 10024','120000','X04');
INSERT INTO Employees
VALUES('X0005','Armani','Bailey','05/04/1994','6468652028','3 East
Essex Drive Brooklyn, NY 11225','100000','X05');
INSERT INTO Employees
VALUES('X0006','Liberty','Guerra','06/19/1985','9171982906','35 Prairie
Rd. Bronx, NY 10473','150000','X06');
INSERT INTO Employees
VALUES('X0007','Maverick','Matthews','04/15/1988','9175416143','22
Acacia Ave. Brooklyn, NY 11218','200000','X07');
INSERT INTO Employees
VALUES('X0008','Jayson','Romero','11/22/1985','7188089254','7994
Grandrose St. New York, NY 10016','80000','X08');
INSERT INTO Employees
VALUES('X0009','Saul','Schmitt','11/20/1980','6462444322','8763 Sutor
St. New York, NY 10025','100000','X09');
INSERT INTO Employees
VALUES('X0010','Gianni','Weiss','11/12/1984','9171522176','893 Country
Lane Bronx, NY 10473','190000','X10');
INSERT INTO Employees
```

```
VALUES('X0011','Sadie','Owens','07/15/1982','6461307399','281 Holly St.
Bronx, NY 10468','160000','X11');
INSERT INTO Employees
VALUES('X0012','Roy','Sloan','08/29/1985','6463747057','9 Hilltop Drive
New York, NY 10033','140000','X12');
INSERT INTO Employees
VALUES('X0013','Lillianna','Morse','11/18/1994','7181853898','91
Hillcrest Ave. Brooklyn, NY 11210','90000','X13');
INSERT INTO Employees
VALUES('X0014','Marcus','Ford','09/11/1983','7187226586','7633 Lilac
Street Brooklyn, NY 11213','130000','X14');
INSERT INTO Employees
VALUES('X0015','Brett','Dillon','06/02/1983','7189048829','7685
Brickyard Road Brooklyn, NY 11225','160000','X15');
INSERT INTO Employees
VALUES('X0016','Rachael','Hess','12/24/1985','7183019456','62 Hudson
Drive Brooklyn, NY 11201','150000','X16');
INSERT INTO Employees
VALUES('X0017','Monica','Pierce','07/03/1990','6462723341','5 NW.
Stillwater Court Bronx, NY 10463','200000','X17');
INSERT INTO Employees
VALUES('X0018','Cornelius','Andrews','09/29/1995','6462879473','7403
Old Glendale St. Bronx, NY 10451','170000','X18');
INSERT INTO Employees
VALUES('X0019','Alanna','Romero','03/31/1991','9174842506','203 Lees
Creek Lane Brooklyn, NY 11203','100000','X19');
INSERT INTO Employees
VALUES('X0020','Eva','Parks','11/06/1990','7188712035','55 Princess
Drive Brooklyn, NY 11233','170000','X20');
INSERT INTO Employees
VALUES('X0021','Beckham','Mccann','03/16/1995','9179933772','8164 Grant
Ave. New York, NY 10016','90000','X01');
INSERT INTO Employees
VALUES('X0022','Lamar','Neal','03/24/1987','6464050860','56 S. Glenlake
Ave. Bronx, NY 10457','110000','X01');
INSERT INTO Employees
VALUES('X0023','Lilly','Cowan','01/21/1994','9177574374','32 North
Rockwell St. Brooklyn, NY 11215','90000','X01');
```

```
INSERT INTO Employees
VALUES('X0024','Dario','Rocha','09/03/1980','6469037735','746 East
Bohemia St. New York, NY 10029','120000','X01');
INSERT INTO Employees
VALUES('X0025','Dangelo','Stewart','04/29/1992','9172811082','9891
Lancaster Street Brooklyn, NY 11212','50000','X02');
INSERT INTO Employees
VALUES('X0026','Erika','Saunders','06/21/1996','9171513393','1 Harvey
Ave. Bronx, NY 10472','80000','X02');
INSERT INTO Employees
VALUES('X0027','Aditya','Kim','04/20/1997','7186292284','80 Bay Meadows
Lane Bronx, NY 10465','60000','X02');
INSERT INTO Employees
VALUES('X0028','Elliott','Ray','01/21/1995','7189427686','7627
Edgewater St. Bronx, NY 10456','120000','X02');
INSERT INTO Employees
VALUES('X0029','Dexter','Mcdonald','06/21/1990','7181502110','568
Pilgrim St. Brooklyn, NY 11229','160000','X03');
INSERT INTO Employees
VALUES('X0030','Kaitlin','Galvan','12/23/1998','6469858926','985 Hill
Field Lane New York, NY 10034','110000','X03');
INSERT INTO Employees
VALUES('X0031','Brooklynn','Kaiser','07/17/1981','9174205997','9321
Bishop Lane Brooklyn, NY 11214','80000','X03');
INSERT INTO Employees
VALUES('X0032','Tristian','Valenzuela','02/16/1996','7188080115','107
Goldfield St. Bronx, NY 10466','100000','X03');
INSERT INTO Employees
VALUES('X0033','Crystal','Henderson','06/07/1985','9179180435','55
Princess Drive Brooklyn, NY 11233','140000','X04');
INSERT INTO Employees
VALUES('X0034','Kristopher','Aguilar','11/03/1988','9171354819','973
East Rocky River Street Bronx, NY 10452','110000','X04');
INSERT INTO Employees
VALUES('X0035','Salvatore','Bennett','01/12/1985','6461630007','405
Beacon Court Bronx, NY 10461','90000','X04');
INSERT INTO Employees
VALUES('X0036','Ellen','Compton','06/07/1987','9177052204','265
```

```
Buttonwood Lane Bronx, NY 10456','110000','X04');
INSERT INTO Employees
VALUES('X0037','Quentin','Garcia','05/08/1989','9177430548','21 Bridge
Dr. Bronx, NY 10465','130000','X05');
INSERT INTO Employees
VALUES('X0038','Jonathon','Thomas','09/28/1984','7189488365','62 Hudson
Drive Brooklyn, NY 11201','130000','X05');
INSERT INTO Employees
VALUES('X0039','Frank','Stewart','02/09/1984','7184648804','612
Newcastle Avenue Brooklyn, NY 11206','100000','X05');
INSERT INTO Employees
VALUES('X0040','Malakai','Hudson','12/13/1999','6469895686','478 Arnold
Street Brooklyn, NY 11203','100000','X05');
INSERT INTO Employees
VALUES('X0041','Memphis','Ball','04/11/1987','7184400383','612
Newcastle Avenue Brooklyn, NY 11206','170000','X06');
INSERT INTO Employees
VALUES('X0042','Tanya','Parrish','06/17/1991','9175728348','48 North
Blackburn St. New York, NY 10128','80000','X06');
INSERT INTO Employees
VALUES('X0043','Isabel','Valenzuela','11/02/1997','6467523320','5 NW.
Stillwater Court Bronx, NY 10463','110000','X06');
INSERT INTO Employees
VALUES('X0044','Owen','Bryan','09/23/1984','9171244577','924 Cemetery
Rd. Brooklyn, NY 11220','120000','X06');
INSERT INTO Employees
VALUES('X0045','Lillianna','Pierce','12/19/1990','9172922151','5 NW.
Stillwater Court Bronx, NY 10463','140000','X07');
INSERT INTO Employees
VALUES('X0046','Milton','Wang','06/08/1988','7188829275','912 Vine
Street Brooklyn, NY 11223','80000','X07');
INSERT INTO Employees
VALUES('X0047','Devon','Carey','12/13/1991','6466065461','355 Rockland
St. New York, NY 10009','100000','X07');
INSERT INTO Employees
VALUES('X0048','Vaughn','Frey','05/16/1999','9176150728','1 Buckingham
St. New York, NY 10040','100000','X07');
INSERT INTO Employees
```

```
VALUES('X0049','Amiah','Blankenship','08/04/1995','7182326346','9483
Ocean St. Brooklyn, NY 11236','80000','X08');
INSERT INTO Employees
VALUES('X0050','Darren','Gibson','01/10/1984','7183783928','746 East
Bohemia St. New York, NY 10029','120000','X08');
INSERT INTO Employees
VALUES('X0051','Dakota','Figueroa','06/14/1998','7185346611','8763
Sutor St. New York, NY 10025','100000','X08');
INSERT INTO Employees
VALUES('X0052','Phoenix','Blanchard','10/10/1983','7186674573','7853
Court St. Bronx, NY 10468','110000','X08');
INSERT INTO Employees
VALUES('X0053','Francisco','Lambert','07/12/1996','9174138056','1
Harvey Ave. Bronx, NY 10472','50000','X09');
INSERT INTO Employees
VALUES('X0054','Rashad','Tate','05/20/1997','6467329662','7331 West
Trusel Drive Bronx, NY 10461','100000','X09');
INSERT INTO Employees
VALUES('X0055','Cali','Cooper','07/26/1984','6461281449','17 Marconi
Rd. Brooklyn, NY 11228','120000','X09');
INSERT INTO Employees
VALUES('X0056','Jewel','Scott','10/20/1997','6461536154','9589 SW.
Jennings Street New York, NY 10032','110000','X09');
INSERT INTO Employees
VALUES('X0057','Aspen','Munoz','05/24/1992','9171825748','680 North
Manhattan Lane New York, NY 10040','60000','X10');
INSERT INTO Employees
VALUES('X0058','Israel','Galvan','03/04/1998','6463713613','11 Green
Street Brooklyn, NY 11213','90000','X10');
INSERT INTO Employees
VALUES('X0059','Dale','Gentry','12/06/1989','6467058754','206 Devon Rd.
Bronx, NY 10453','100000','X10');
INSERT INTO Employees
VALUES('X0060','Madisyn','Ayala','07/30/1980','9179996971','426 Ocean
Rd. Brooklyn, NY 11230','110000','X10');
INSERT INTO Employees
VALUES('X0061','Lukas','Bishop','09/15/1985','9174240687','551 Chestnut
St. Brooklyn, NY 11201','60000','X11');
```

```
INSERT INTO Employees
VALUES('X0062','Leonel','Wu','07/20/1999','6462268656','221 North
Bishop Street Brooklyn, NY 11230','110000','X11');
INSERT INTO Employees
VALUES('X0063','Theresa','Horne','10/13/1988','9177798299','141 West
Parker Street New York, NY 10024','70000','X11');
INSERT INTO Employees
VALUES('X0064','Harper','Stuart','01/16/1992','6466173428','735 East
Green Lake St. New York, NY 10032','120000','X11');
INSERT INTO Employees
VALUES('X0065','Sara','Carney','09/07/1980','9172227976','7466 Holly
Dr. New York, NY 10002','80000','X12');
INSERT INTO Employees
VALUES('X0066','Brylee','Brady','07/15/1997','9174333835','22 Acacia
Ave. Brooklyn, NY 11218','110000','X12');
INSERT INTO Employees
VALUES('X0067','Giovanni','Macdonald','02/25/1988','9176231848','35
Prairie Rd. Bronx, NY 10473','120000','X12');
INSERT INTO Employees
VALUES('X0068','Adonis','Cohen','01/08/1990','7181208237','252 Olive
Ave. Brooklyn, NY 11228','120000','X12');
INSERT INTO Employees
VALUES('X0069','Douglas','Acosta','08/08/1995','7186005913','8408 West
Brookside Avenue Brooklyn, NY 11228','150000','X13');
INSERT INTO Employees
VALUES('X0070','Jade','Cohen','06/06/1982','9174350677','141 West
Parker Street New York, NY 10024','120000','X13');
INSERT INTO Employees
VALUES('X0071','Dario','Hurst','08/12/1980','9175437223','21 Deerfield
St. Bronx, NY 10463','110000','X13');
INSERT INTO Employees
VALUES('X0072','Duncan','Orr','03/01/1994','9175679307','49 West Mill
Pond Rd. New York, NY 10002','110000','X13');
INSERT INTO Employees
VALUES('X0073','Roberto','Bates','06/18/1990','9176336239','9891
Lancaster Street Brooklyn, NY 11212','50000','X14');
INSERT INTO Employees
VALUES('X0074','Krista','Hanson','01/02/1987','7183907130','87 Hill
```

```
Field Ave. Bronx, NY 10453','100000','X14');
INSERT INTO Employees
VALUES('X0075','Paula','Schaefer','12/16/1982','7182968883','8973 Cross
Dr. Brooklyn, NY 11238','90000','X14');
INSERT INTO Employees
VALUES('X0076','Marcus','Hatfield','09/16/1989','7185446493','65 Shadow
Brook Ave. Brooklyn, NY 11208','90000','X14');
INSERT INTO Employees
VALUES('X0077','Annalise','Mccann','07/22/1990','7184708418','18
Tailwater Road Bronx, NY 10453','140000','X15');
INSERT INTO Employees
VALUES('X0078','Shyla','Underwood','09/16/1981','7188937433','961
Monroe St. Brooklyn, NY 11212','110000','X15');
INSERT INTO Employees
VALUES('X0079','Octavio','Meyer','10/18/1992','7183269975','8931
Armstrong St. New York, NY 10024','80000','X15');
INSERT INTO Employees
VALUES('X0080','London','Carr','07/08/1999','9175122984','8518 2nd St.
New York, NY 10028','120000','X15');
INSERT INTO Employees
VALUES('X0081','Mia','Schmidt','08/06/1980','9171470963','637 Wood St.
New York, NY 10025','140000','X16');
INSERT INTO Employees
VALUES('X0082','Evelin','Wu','12/14/1987','9176288322','7891 Arnold
Street Brooklyn, NY 11204','100000','X16');
INSERT INTO Employees
VALUES('X0083','Layton','Landry','12/25/1982','7189215462','8763 Sutor
St. New York, NY 10025','70000','X16');
INSERT INTO Employees
VALUES('X0084','Stephany','Kim','03/21/1990','9177559085','71 Homewood
Dr. Brooklyn, NY 11209','90000','X16');
INSERT INTO Employees
VALUES('X0085','Juliana','Rich','07/19/1991','6469478107','8012 Olive
Street New York, NY 10011','170000','X17');
INSERT INTO Employees
VALUES('X0086','Keith','Stafford','10/31/1997','9174355343','9 Hilltop
Drive New York, NY 10033','100000','X17');
INSERT INTO Employees
```

```
VALUES('X0087','Joshua','Simmons','07/12/1989','9171698642','8507
Delaware Lane Bronx, NY 10458','70000','X17');
INSERT INTO Employees
VALUES('X0088','Emma','Sawyer','06/04/1996','9179140952','20 Mulberry
Ave. Brooklyn, NY 11224','90000','X17');
INSERT INTO Employees
VALUES('X0089','Naima','Campbell','05/27/1994','6469193590','56 South
Lakeshore Rd. Bronx, NY 10457','150000','X18');
INSERT INTO Employees
VALUES('X0090','Alexzander','Jones','09/23/1996','6465868835','145
School Dr. Brooklyn, NY 11219','100000','X18');
INSERT INTO Employees
VALUES('X0091','Kassidy','Jones','07/17/1986','7187878167','630
Walnutwood Court Bronx, NY 10458','70000','X18');
INSERT INTO Employees
VALUES('X0092','Phoenix','Zuniga','06/07/1981','9178231971','9865
Forest St. Brooklyn, NY 11215','100000','X18');
INSERT INTO Employees
VALUES('X0093','Madelyn','Sweeney','07/25/1996','7186891409','221 North
Bishop Street Brooklyn, NY 11230','110000','X19');
INSERT INTO Employees
VALUES('X0094','Alanna','Sanchez','12/21/1995','6468464835','55
Princess Drive Brooklyn, NY 11233','110000','X19');
INSERT INTO Employees
VALUES('X0095','Miley','Pitts','11/03/1984','7186935023','244 Penn
Street Bronx, NY 10451','80000','X19');
INSERT INTO Employees
VALUES('X0096','Lina','Page','09/07/1989','7185382406','206 Devon Rd.
Bronx, NY 10453','110000','X19');
INSERT INTO Employees
VALUES('X0097','Ashley','Baker','05/26/1998','7185620299','1 Harvey
Ave. Bronx, NY 10472','70000','X20');
INSERT INTO Employees
VALUES('X0098','Marcus','Gregory','10/24/1980','9173136553','42
Railroad Ave. Brooklyn, NY 11220','80000','X20');
INSERT INTO Employees
VALUES('X0099','Bruce','Gray','06/21/1988','6464110479','313 Wayne Dr.
Bronx, NY 10467','60000','X20');
```

```
INSERT INTO Employees
VALUES('X0100','Baron','Blankenship','05/31/1989','7187914664','7357 El
Dorado St. Bronx, NY 10467','100000','X20');

INSERT INTO FacilityStaff VALUES('X0022');
INSERT INTO FacilityStaff VALUES('X0026');
INSERT INTO FacilityStaff VALUES('X0030');
INSERT INTO FacilityStaff VALUES('X0034');
INSERT INTO FacilityStaff VALUES('X0038');
INSERT INTO FacilityStaff VALUES('X0042');
INSERT INTO FacilityStaff VALUES('X0046');
INSERT INTO FacilityStaff VALUES('X0050');
INSERT INTO FacilityStaff VALUES('X0054');
INSERT INTO FacilityStaff VALUES('X0058');
INSERT INTO FacilityStaff VALUES('X0062');
INSERT INTO FacilityStaff VALUES('X0066');
INSERT INTO FacilityStaff VALUES('X0070');
INSERT INTO FacilityStaff VALUES('X0074');
INSERT INTO FacilityStaff VALUES('X0078');
INSERT INTO FacilityStaff VALUES('X0082');
INSERT INTO FacilityStaff VALUES('X0086');
INSERT INTO FacilityStaff VALUES('X0090');
INSERT INTO FacilityStaff VALUES('X0094');
INSERT INTO FacilityStaff VALUES('X0098');

INSERT INTO Instructors VALUES('X0023');
INSERT INTO Instructors VALUES('X0027');
INSERT INTO Instructors VALUES('X0031');
INSERT INTO Instructors VALUES('X0035');
INSERT INTO Instructors VALUES('X0039');
INSERT INTO Instructors VALUES('X0043');
INSERT INTO Instructors VALUES('X0047');
INSERT INTO Instructors VALUES('X0051');
INSERT INTO Instructors VALUES('X0055');
INSERT INTO Instructors VALUES('X0059');
INSERT INTO Instructors VALUES('X0063');
INSERT INTO Instructors VALUES('X0067');
INSERT INTO Instructors VALUES('X0071');
```

```
INSERT INTO Instructors VALUES('X0075');
INSERT INTO Instructors VALUES('X0079');
INSERT INTO Instructors VALUES('X0083');
INSERT INTO Instructors VALUES('X0087');
INSERT INTO Instructors VALUES('X0091');
INSERT INTO Instructors VALUES('X0095');
INSERT INTO Instructors VALUES('X0099');

INSERT INTO Trainers VALUES('X0024','ACE');
INSERT INTO Trainers VALUES('X0028','ACSM');
INSERT INTO Trainers VALUES('X0032','ACE');
INSERT INTO Trainers VALUES('X0036','ACSM');
INSERT INTO Trainers VALUES('X0040','NASM');
INSERT INTO Trainers VALUES('X0044','ISSA');
INSERT INTO Trainers VALUES('X0048','ISSA');
INSERT INTO Trainers VALUES('X0052','NSCA');
INSERT INTO Trainers VALUES('X0056','ACSM');
INSERT INTO Trainers VALUES('X0060','ACSM');
INSERT INTO Trainers VALUES('X0064','ISSA');
INSERT INTO Trainers VALUES('X0068','ACSM');
INSERT INTO Trainers VALUES('X0072','ACE');
INSERT INTO Trainers VALUES('X0076','NSCA');
INSERT INTO Trainers VALUES('X0080','ACSM');
INSERT INTO Trainers VALUES('X0084','NSCA');
INSERT INTO Trainers VALUES('X0088','ACSM');
INSERT INTO Trainers VALUES('X0092','ISSA');
INSERT INTO Trainers VALUES('X0096','ACSM');
INSERT INTO Trainers VALUES('X0100','NSCA');

INSERT INTO isManager VALUES('X0001');
INSERT INTO isManager VALUES('X0002');
INSERT INTO isManager VALUES('X0003');
INSERT INTO isManager VALUES('X0004');
INSERT INTO isManager VALUES('X0005');
INSERT INTO isManager VALUES('X0006');
INSERT INTO isManager VALUES('X0007');
INSERT INTO isManager VALUES('X0008');
INSERT INTO isManager VALUES('X0009');
```

```
INSERT INTO isManager VALUES('X0010');
INSERT INTO isManager VALUES('X0011');
INSERT INTO isManager VALUES('X0012');
INSERT INTO isManager VALUES('X0013');
INSERT INTO isManager VALUES('X0014');
INSERT INTO isManager VALUES('X0015');
INSERT INTO isManager VALUES('X0016');
INSERT INTO isManager VALUES('X0017');
INSERT INTO isManager VALUES('X0018');
INSERT INTO isManager VALUES('X0019');
INSERT INTO isManager VALUES('X0020');



insert into ClassAssignment values('10001','82348');
insert into ClassAssignment values('10002','91242');
insert into ClassAssignment values('10003','72226');
insert into ClassAssignment values('10004','12345');
insert into ClassAssignment values('10005','96647');
insert into ClassAssignment values('10006','92551');
insert into ClassAssignment values('10007','61997');
insert into ClassAssignment values('10008','69001');
insert into ClassAssignment values('10009','91318');
insert into ClassAssignment values('10010','13920');
insert into ClassAssignment values('10011','14428');
insert into ClassAssignment values('10012','12098');
insert into ClassAssignment values('10013','96102');
insert into ClassAssignment values('10014','25547');
insert into ClassAssignment values('10015','98652');
insert into ClassAssignment values('10016','64823');
insert into ClassAssignment values('10017','46672');
insert into ClassAssignment values('10018','90466');
insert into ClassAssignment values('10019','96924');
insert into ClassAssignment values('10020','82662');

insert into Classes values('10001','Stretching',’M
07:30’,’101’,’40’,’X0023’);
insert into Classes values('10002','Leg',’W 11:30’,’104’,’40’,’X0035’);
```

```
insert into Classes values('10003','Arm',’F 12:30’,’102’,’30’,’X0027’);
insert into Classes values('10004','bike’,’Sa
13:30’,’103’,’30’,’X0031’);
insert into Classes values('10005','Stretching',’M
13:30’,’101’,’40’,’X0023’);
insert into Classes values('10006','Leg',’M 13:30’,’106’,’40’,’X0043’);
insert into Classes values('10007','Arm',’Tu
14:30’,’102’,’30’,’X0027’);
insert into Classes values('10008','bike’,’W
14:30’,’103’,’30’,’X0031');
insert into Classes values('10009','Stretching',’Su
16:30’,’101’,’40’,’X0023’);
insert into Classes values('10010','Leg',’M 14:30’,’105’,’40’,’X0039’);
insert into Classes values('10011','Arm',’M 16:30’,’102’,’30’,’X0027’);
insert into Classes values('10012','bike’,’Th
16:30’,’103’,’30’,’X0031');
insert into Classes values('10013','Stretching',’W
19:30’,’101’,’40’,’X0023’);
insert into Classes values('10014','Leg',’M 16:30’,’104’,’40’,’X0035’);
insert into Classes values('10015','Arm',’Tu
18:30’,’102’,’30’,’X0027’);
insert into Classes values('10016','bike’,’W
18:30’,’103’,’30’,’X0031');
insert into Classes values('10017','Stretching',’F
21:30’,’101’,’40’,’X0023’);
insert into Classes values('10018','Leg',’M 18:30’,’107’,’40’,’X0047’);
insert into Classes values('10019','Arm',’Tu
07:30’,’102’,’30’,’X0027’);
insert into Classes values('10020','bike’,’W
07:30’,’103’,’30’,’X0031’);

INSERT INTO MembershipPlan VALUES('Bronze A','10')
INSERT INTO MembershipPlan VALUES('Bronze B','15')
INSERT INTO MembershipPlan VALUES('Bronze C','20')
INSERT INTO MembershipPlan VALUES('Bronze D','25')
INSERT INTO MembershipPlan VALUES('Silver A','40')
INSERT INTO MembershipPlan VALUES('Silver B','45')
INSERT INTO MembershipPlan VALUES('Silver C','50')
```

```
INSERT INTO MembershipPlan VALUES('Silver D','55')
INSERT INTO MembershipPlan VALUES('Gold A','70')
INSERT INTO MembershipPlan VALUES('Gold B','75')
INSERT INTO MembershipPlan VALUES('Gold C','80')
INSERT INTO MembershipPlan VALUES('Gold D','85')
INSERT INTO MembershipPlan VALUES('Platinum A','100')
INSERT INTO MembershipPlan VALUES('Platinum B','105')
INSERT INTO MembershipPlan VALUES('Platinum C','110')
INSERT INTO MembershipPlan VALUES('Platinum D','115')
INSERT INTO MembershipPlan VALUES('Diamond A','130')
INSERT INTO MembershipPlan VALUES('Diamond B','135')
INSERT INTO MembershipPlan VALUES('Diamond C','140')
INSERT INTO MembershipPlan VALUES('Diamond D','145')
```

```
insert into Rooms values('101','X01');
insert into Rooms values('102','X02');
insert into Rooms values('103','X03');
insert into Rooms values('104','X04');
insert into Rooms values('105','X05');
insert into Rooms values('106','X06');
insert into Rooms values('107','X07');
insert into Rooms values('108','X08');
insert into Rooms values('109','X09');
insert into Rooms values('110','X10');
insert into Rooms values('111','X11');
insert into Rooms values('112','X12');
insert into Rooms values('113','X13');
insert into Rooms values('114','X14');
insert into Rooms values('115','X15');
insert into Rooms values('116','X16');
insert into Rooms values('117','X17');
insert into Rooms values('118','X18');
insert into Rooms values('119','X19');
insert into Rooms values('120','X20');

insert into Supplier values('SP001','Nike', '5036716453','One Bowerman
```

```
Drive Beaverton, OR 97005', 'Investor.Relations@nike.com');
insert into Supplier values('SP002','Adidas', '8009829337', '5055 N
Greeley Avenue Portland, OR 97217', 'support@adidas.us');
insert into Supplier values('SP003','Reebok', '8574432000', '25 Drydock
Ave, Suite 110E Boston, MA 02210', 'support@reebok.us');
insert into Supplier values('SP004','Puma', '9786981000', '10 Lyberty
Way, Westford, MA 01886', 'customerservice.us@puma.com');
insert into Supplier values('SP005','U.A', '8887276687', '1020 Hull St,
Baltimore, MD 21230', 'support@UA.us');
insert into Supplier values('SP006','S.B.D', '2074014999', '1135
Hammond Street Bangor, ME 04401', 'help@sbd-usa.com');
insert into Supplier values('SP007','Ekin', '3546176035', 'One Bowerman
Drive Bangor, ME 01104', 'support@ekin.com');
insert into Supplier values('SP008','Abibas', '7339289008', '5055 N
Greeley Avenue Boston, MA 02120', 'support@abibas.us');
insert into Supplier values('SP009','Kobeer', '2344758546', '25 Drydock
Ave Westford, MA 01688', 'customerservice@kobeer.us');
insert into Supplier values('SP010','Amup', '1896879345', '25 Drydock
Ave, Portland, OR 97721', 'support@Amup.us');
insert into Supplier values('SP011','AWA', '7866727888', '1520 Hull St,
Portland, OR 79217', 'support@au.com');
insert into Supplier values('SP012','Bsman', '9994104702', '533 Hammond
Street Baltimore, MD 22123', 'support@DBS.us');
insert into Supplier values('SP013','NARP', '5039822000', '5055 N
Greeley Avenue Westford, MA 01886', 'support@NARP.us');
insert into Supplier values('SP014','Seaka', '9787274999', 'One
Bowerman Drive Boston, MA 01102', 'support@seaka.us');
insert into Supplier values('SP015','Bomu', '8884016035', '1055 N
Greeley Avenue Bangor, ME 41104', 'support@bomu.us');
insert into Supplier values('SP016','Jump', '2074619008', '75 Drydock
Ave Baltimore, MD 11320', 'support@jump.us');
insert into Supplier values('SP017','Hokaa', '3549284758', '53 Lyberty
Way Beaverton, OR 96500', 'support@hokka.us');
insert into Supplier values('SP018','Feind', '7332346879', '320 Hull St
Portland, OR 95171', 'support@feind.us');
insert into Supplier values('SP019','Flaaash', '7958906934', '920
Folsom Street San Francisco, CA 91150', 'support@flaaash.us');
insert into Supplier values('SP020','NewBlance', '6177834000', '100
```

```
Guest St, Boston, MA 02135', 'support@newbalance.us');

insert into Equipment values('X1001','Smith
machine','Upper','Chest','1/1/2019','X0022','3000','SP001','1/1/2020','
2','X01');
insert into Equipment values('X1002','Squat
rack','Others','None','1/1/2019','X0022','2000','SP002','1/1/2020','5',
'X01');
insert into Equipment values('X1003','Military
press','Upper','Shoulder','6/1/2019','X0022','2000','SP003','1/1/2020',
'4','X01');
insert into Equipment values('X1004','Spinning
bike','Lower','None','1/1/2019','X0022','3000','SP004','1/1/2020','5','
X01');
insert into Equipment
values('X1005','Stepper','Lower','Leg','10/10/2019','X0022','3000','SP0
05','1/1/2020','3','X01');
insert into Equipment values('X2001','Smith
machine','Upper','Chest','1/1/2019','X0026','3000','SP006','1/1/2020','
2','X02');
insert into Equipment values('X2002','Squat
rack','Others','None','1/1/2019','X0026','2000','SP007','1/1/2020','5',
'X02');
insert into Equipment values('X2003','Military
press','Upper','Shoulder','6/1/2019','X0026','2000','SP008','1/1/2020',
'4','X02');
insert into Equipment values('X2004','Spinning
bike','Lower','None','1/1/2019','X0026','3000','SP009','1/1/2020','5','
X02');
insert into Equipment
values('X2005','Stepper','Lower','Leg','10/10/2019','X0026','3000','SP0
10','1/1/2020','3','X02');
insert into Equipment values('X3001','Smith
machine','Upper','Chest','1/1/2019','X0030','3000','SP011','1/1/2020','
2','X03');
insert into Equipment values('X3002','Squat
rack','Others','None','1/1/2019','X0030','2000','SP012','1/1/2020','5',
'X03');
```

```
insert into Equipment values('X3003','Military
press','Upper','Shoulder','6/1/2019','X0030','2000','SP013','1/1/2020',
'4','X03');
insert into Equipment values('X3004','Spinning
bike','Lower','None','1/1/2019','X0030','3000','SP014','1/1/2020','5','
X03');
insert into Equipment
values('X3005','Stepper','Lower','Leg','10/10/2019','X0030','3000','SP0
15','1/1/2020','3','X03');
insert into Equipment values('X4001','Smith
machine','Upper','Chest','1/1/2019','X0034','3000','SP016','1/1/2020','
2','X04');
insert into Equipment values('X4002','Squat
rack','Others','None','1/1/2019','X0034','2000','SP016','1/1/2020','5',
'X04');
insert into Equipment values('X4003','Military
press','Upper','Shoulder','6/1/2019','X0034','2000','SP016','1/1/2020',
'4','X04');
insert into Equipment values('X5001','Spinning
bike','Lower','None','1/1/2019','X0038','3000','SP017','1/1/2020','5','
X05');
insert into Equipment
values('X6001','Stepper','Lower','Leg','10/10/2019','X0042','3000','SP0
17','1/1/2020','3','X06');

insert into Maintenance_and_repair_report
values('T1001','X1004','X0022','2/1/2020','10','Maintenance','1');
insert into Maintenance_and_repair_report
values('T1002','X1005','X0022','2/1/2020','10','Maintenance','3');
insert into Maintenance_and_repair_report
values('T1003','X1004','X0022','3/1/2020','100','Repair','7');
insert into Maintenance_and_repair_report
values('T1004','X1005','X0022','4/1/2020','10','Maintenance','3');
insert into Maintenance_and_repair_report
values('T1005','X1004','X0022','5/1/2020','10','Repair','3');
insert into Maintenance_and_repair_report
values('T1006','X1002','X0022','5/1/2020','100','Repair','7');
insert into Maintenance_and_repair_report
```

```
values('T2001','X2004','X0026','3/1/2020','10','Maintenance','1');
insert into Maintenance_and_repair_report
values('T2002','X2005','X0026','3/1/2020','10','Maintenance','3');
insert into Maintenance_and_repair_report
values('T2003','X2004','X0026','5/1/2020','10','Maintenance','1');
insert into Maintenance_and_repair_report
values('T2004','X2001','X0026','5/1/2020','10','Maintenance','3');
insert into Maintenance_and_repair_report
values('T3001','X3004','X0030','2/1/2020','10','Maintenance','1');
insert into Maintenance_and_repair_report
values('T3002','X3005','X0030','3/1/2020','10','Maintenance','3');
insert into Maintenance_and_repair_report
values('T3003','X3005','X0030','5/1/2020','200','Repair','7');
insert into Maintenance_and_repair_report
values('T4001','X4001','X0034','3/1/2020','10','Maintenance','1');
insert into Maintenance_and_repair_report
values('T4002','X4003','X0034','3/1/2020','10','Maintenance','1');
insert into Maintenance_and_repair_report
values('T5001','X5001','X0038','2/1/2020','10','Maintenance','1');
insert into Maintenance_and_repair_report
values('T5002','X5001','X0038','3/1/2020','10','Maintenance','1');
insert into Maintenance_and_repair_report
values('T6001','X6001','X0042','2/1/2020','50','Repair','3');
insert into Maintenance_and_repair_report
values('T6002','X6001','X0042','4/1/2020','10','Maintenance','1');
insert into Maintenance_and_repair_report
values('T6003','X6001','X0042','5/1/2020','10','Maintenance','1');

Insert into Member values("82348", "Anakin", "Skywalker", "2/14/1977",
"anakin.skywalker@force.com", "7234239785", "123 Coruscant Ave", "Gold
B");
Insert into Member values("91242", "Padme", "Amidala", "10/31/1969",
"padme.amidala@force.com", "8749958732", "123 Coruscant Ave", "Silver
D");
Insert into Member values("72226", "Obiwan", "Kenobi", "7/4/1965",
"obiwan.kenobi@force.com", "8662539449", "66 Temple Road", "Silver C");
Insert into Member values("12345", "Cee", "Threepio", "4/1/1980",
"cee.threepio@force.com", "9103367233", "13 Tatooine Drive", "Gold B");
```

```
Insert into Member values("96647", "Artoo", "Deetoo", "6/1/1965",
"artoo.deetoo@force.com", "7244596278", "10 Google Street", "Bronze
A");
Insert into Member values("92551", "Luke", "Skywalker", "1/1/2000",
"luke.skywalker@force.com", "2655239998", "13 Tatooine Drive", "Diamond
B");
Insert into Member values("61997", "Leia", "Organa", "1/1/2000",
"leia.organa@force.com", "8014139467", "90 Alderaan Street", "Platinum
A");
Insert into Member values("69001", "Mace", "Windu", "4/15/1950",
"mace.windu@force.com", "9715688570", "467 Reservoir Ave", "Bronze B");
Insert into Member values("91318", "Ahsoka", "Tano", "8/15/1985",
"ahsoka.tano@force.com", "2346350899", "625 Padawan Drive ", "Gold A");
Insert into Member values("13920", "Han", "Solo", "4/1/1990",
"han.solo@force.com", "7730144245", "12 Falcon Street", "Bronze D");
Insert into Member values("14428", "Boba", "Fett", "12/25/1990",
"boba.fett@force.com", "1693230108", "1500 Kamino Street", "Silver A");
Insert into Member values("12098", "Chew", "Bacca", "6/21/1977",
"chew.bacca@force.com", "6216462834", "200 Kashyyyk Place", "Silver
A");
Insert into Member values("96102", "Yoda", "Master", "5/31/1995",
"yoda.master@force.com", "4380333424", "45 Dagobah Ave", "Gold C");
Insert into Member values("25547", "Quigon", "Jinn", "12/31/1988",
"quigon.jinn@force.com", "9397956418", "99 Temple Road", "Platinum C");
Insert into Member values("98652", "Jabba", "Hutt", "9/13/1976",
"jabba.hutt@force.com", "6233767468", "84 Tatooine Street", "Bronze
B");
Insert into Member values("64823", "Darth", "Vader", "2/14/1997",
"darth.vader@force.com", "3554321646", "19 Mustafar Blvd", "Gold D");
Insert into Member values("46672", "Darth", "Maul", "9/30/1979",
"darth.maul@force.com", "7117400507", "13 Dathomir Place", "Diamond
A");
Insert into Member values("90466", "Lando", "Calrissian", "5/2/1986",
"lando.calrissian@force.com", "2208832784", "7 Cloud Road", "Silver
B");
Insert into Member values("96924", "Din", "Djarin", "2/28/1999",
"din.djarin@force.com", "2849225703", "62 Mandalore Road", "Silver C");
Insert into Member values("82662", "Wilhuff", "Tarkin", "11/8/1980",
```

```
"wilhuff.tarkin@force.com", "7774272951", "91 Star Place", "Silver C");

INSERT INTO TrainerAssignment
VALUES('GKE44','X0024','82348','5/11/2020','13:00');
INSERT INTO TrainerAssignment
VALUES('YTI31','X0028','82348','5/18/2020','13:00');
INSERT INTO TrainerAssignment
VALUES('4FMYV','X0032','82348','5/25/2020','13:00');
INSERT INTO TrainerAssignment
VALUES('A39LU','X0036','82348','6/1/2020','13:00');
INSERT INTO TrainerAssignment
VALUES('OZ32G','X0040','14428','5/11/2020','14:30');
INSERT INTO TrainerAssignment
VALUES('ZJ1RC','X0044','14428','5/18/2020','14:30');
INSERT INTO TrainerAssignment
VALUES('OLAEN','X0048','14428','5/25/2020','14:30');
INSERT INTO TrainerAssignment
VALUES('319JP','X0052','14428','6/1/2020','14:30');
INSERT INTO TrainerAssignment
VALUES('CSLAQ','X0056','98652','5/11/2020','08:30');
INSERT INTO TrainerAssignment
VALUES('C97MP','X0060','98652','5/18/2020','08:30');
INSERT INTO TrainerAssignment
VALUES('43J69','X0064','98652','5/25/2020','08:30');
INSERT INTO TrainerAssignment
VALUES('SJ6G7','X0068','98652','6/1/2020','08:30');
INSERT INTO TrainerAssignment
VALUES('S8B41','X0072','91318','5/11/2020','10:00');
INSERT INTO TrainerAssignment
VALUES('937GM','X0076','91318','5/18/2020','10:00');
INSERT INTO TrainerAssignment
VALUES('YTA6X','X0080','91318','5/25/2020','10:00');
INSERT INTO TrainerAssignment
VALUES('QTHU0','X0084','91318','6/1/2020','10:00');
INSERT INTO TrainerAssignment
VALUES('L8CJM','X0088','72226','5/11/2020','19:00');
INSERT INTO TrainerAssignment
VALUES('PKSAB','X0092','72226','5/18/2020','19:00');
```

```
INSERT INTO TrainerAssignment
VALUES('2KAP6','X0096','72226','5/25/2020','19:00');
INSERT INTO TrainerAssignment
VALUES('0FUFF','X0100','72226','6/1/2020','19:00');

Insert into Dues values("OVMSSAF", "300", "Active", "6/1/2020",
"82348");
Insert into Dues values("XXOEDVR", "200", "Inactive", "6/1/2020",
"91242");
Insert into Dues values("UVQGYEK", "200", "Active", "6/1/2020",
"72226");
Insert into Dues values("OZGQZXK", "600", "Active - Past Due",
"6/1/2020", "12345");
Insert into Dues values("MAJBJEA", "100", "Inactive", "6/1/2020",
"96647");
Insert into Dues values("LDAMHUM", "300", "Active", "6/1/2020",
"92551");
Insert into Dues values("PXTRSBF", "300", "Active", "6/1/2020",
"61997");
Insert into Dues values("EQJXVRQ", "100", "Active", "6/1/2020",
"69001");
Insert into Dues values("CZXHSCW", "300", "Active", "6/1/2020",
"91318");
Insert into Dues values("KMBXWJP", "100", "Inactive", "6/1/2020",
"13920");
Insert into Dues values("ZALSXJE", "200", "Active", "6/1/2020",
"14428");
Insert into Dues values("TCKGGVH", "300", "Active", "6/1/2020",
"12098");
Insert into Dues values("NRIVUPV", "300", "Active", "6/1/2020",
"96102");
Insert into Dues values("MZNDTML", "200", "Inactive", "6/1/2020",
"25547");
Insert into Dues values("YHIOIVZ", "100", "Active", "6/1/2020",
"98652");
Insert into Dues values("ZPTEEKS", "300", "Inactive", "6/1/2020",
"64823");
Insert into Dues values("WQZWCBV", "300", "Active", "6/1/2020",
```

```
"46672");
Insert into Dues values("ZTWPZMC", "200", "Active", "6/1/2020",
"90466");
Insert into Dues values("NTMVLEX", "400", "Active - Past Due",
"6/1/2020", "96924");
Insert into Dues values("HNVASSD", "200", "Active", "6/1/2020",
"82662");

insert into Product values('M0001','Muscle Fit
Shirts','50','Tops','SP002',’X01’);
insert into Product values('M0002', ‘FKN Gym Wear Gym Bag –
Grey’,'95','Accessories','SP001',’X01’);
insert into Product values('F0003',’Sport
Bras’,'49','Tops','SP008',’X01');
insert into Product values('M0004',’Training
Tights’,'59','Bottom','SP006',’X01');
insert into Product
values('M0005',’Jogger’,'79','Bottom','SP010',’X01');
insert into Product values('F0006',’Running
Short’,'49','Bottom','SP010',’X01');
insert into Product values('M0007',’Superman
hat’,'39','Accessories','SP011',’X01');
insert into Product values('M0008',’7/8
Leggings’,'59','Bottom','SP005',’X01');
insert into Product
values('F0009',’Jogger’,'79','Bottom','SP010',’X01');
insert into Product values('F0011',’BTS
Towels’,'29','Accessories','SP002',’X01');
insert into Product
values('M0012',’Jogger’,'79','Bottom','SP010',’X01');
insert into Product values('M0013',’Muscle Nation Bum
Bag,'79','Accessories','SP003',’X01');
insert into Product values('F0014',’Pink
Shaker’,'49','Accessories','SP008',’X02');
insert into Product values('M0015',’A-Frame
Hat’,'39','Accessories','SP014',’X02');
insert into Product values('F0016',’White
socks’,'19','Accessories','SP013',’X02');
```

```
insert into Product
values('M0017',’Swimwear’,'99','Bottom','SP003',’X02');
insert into Product values('F0018',’Booty
Short’,'49','Bottom','SP015',’X02');
insert into Product values('M0019',’Bike
Short’,'39','Bottom','SP019',’X02');
insert into Product values('F0020',’Zip up
Hoodie’,'79','Top','SP018',’X02');
insert into Product values('F0021',’Crop
Hoodie’,'89','Top','SP008',’X02');
insert into Product values('F0022',’Black
socks’,'29','Accessories','SP020',’X02');

insert into Inventory values(‘I1001’, 'M0001','10','5');
insert into Inventory values(‘I3002’, 'M0002','17','15');
insert into Inventory values(‘I2003’, 'F0003','27','15');
insert into Inventory values(‘I1004’, 'M0004','11','5');
insert into Inventory values(‘I2005’, 'M0005','10','1');
insert into Inventory values(‘I1006’, 'F0006','2','0');
insert into Inventory values(‘I2007’, 'M0007','5','0');
insert into Inventory values(‘I3008’, 'M0008','17','15');
insert into Inventory values(‘I1009’, 'F0009','10','5');
insert into Inventory values(‘I3010’, 'M0002','7','2');
insert into Inventory values(‘I2011’, 'F0011','10','5');
insert into Inventory values(‘I1012’, 'F0003','20','15');
insert into Inventory values(‘I2013’, 'F0014','30','10');
insert into Inventory values(‘I3014’, 'M0013','12','10');
insert into Inventory values(‘I2015’, 'M0015','10','1');
insert into Inventory values(‘I1016’, 'F0014','17','15');
insert into Inventory values(‘I2017’, 'M0013','10','1');
insert into Inventory values(‘I2018’, 'M0015','8','5');
insert into Inventory values(‘I3019’, 'M0015','22','8');
insert into Inventory values(‘I1020’, 'M0015','14','6');

insert into Sales  values(‘S0001’, 'X0022','3/2/2020');
insert into Sales  values(‘S0002’, 'X0022','1/2/2020');
insert into Sales  values(‘S0003’, 'X0022','3/23/2020');
insert into Sales  values(‘S0004’, 'X0026','3/12/2020');
```

```
insert into Sales  values(‘S0005’, 'X0022','3/3/2020');
insert into Sales  values(‘S0006’, 'X0030','3/5/2020');
insert into Sales  values(‘S0007’, 'X0026','3/7/2020');
insert into Sales  values(‘S0008’, 'X0042','3/8/2020');
insert into Sales  values(‘S0009’, 'X0034','3/1/2020');
insert into Sales  values(‘S0010’, 'X0034','3/3/2020');
insert into Sales  values(‘S0011’, 'X0034','3/10/2020');
insert into Sales  values(‘S0012’, 'X0038','3/3/2020');
insert into Sales  values(‘S0013’, 'X0030','3/5/2020');
insert into Sales  values(‘S0014’, 'X0026','3/12/2020');
insert into Sales  values(‘S0015’, 'X0022','3/2/2020');
insert into Sales  values(‘S0016’, 'X0042','3/9/2020');
insert into Sales  values(‘S0017’, 'X0022','3/7/2020');
insert into Sales  values(‘S0018’, 'X0038','3/12/2020');
insert into Sales  values(‘S0019’, 'X0030','3/15/2020');
insert into Sales  values(‘S0020’, 'X0026','3/19/2020');

insert into Salesitem  values(‘M0001’, 'S0001','1');
insert into Salesitem  values(‘M0001’, 'S0002','2');
insert into Salesitem  values(‘M0002’, 'S0003','1');
insert into Salesitem  values(‘F0003’, 'S0004','4');
insert into Salesitem  values(‘M0004’, 'S0005','1');
insert into Salesitem  values(‘M0005’, 'S0006','1');
insert into Salesitem  values(‘M0007’, 'S0006','1');
insert into Salesitem  values(‘M0008’, 'S0007','1');
insert into Salesitem  values(‘F0022’, 'S0008','2');
insert into Salesitem  values(‘M0001’, 'S0009','3');
insert into Salesitem  values(‘F0011’, 'S0010','1');
insert into Salesitem  values(‘M0001’, 'S0011','1');
insert into Salesitem  values(‘M0001’, 'S0012','2');
insert into Salesitem  values(‘M0012’, 'S0013','3');
insert into Salesitem  values(‘M0013’, 'S0014','1');
insert into Salesitem  values(‘M0015’, 'S0015','1');
insert into Salesitem  values(‘M0002’, 'S0016','1');
insert into Salesitem  values(‘F0020’, 'S0017','1');
insert into Salesitem  values(‘F0021’, 'S0018','6');
insert into Salesitem  values(‘F0009’, 'S0019','4');
insert into Salesitem  values(‘F0003’, 'S0020','1');
```

```
INSERT INTO EquipmentAssignment VALUES('X1001','10001')
INSERT INTO EquipmentAssignment VALUES('X1002','10001')
INSERT INTO EquipmentAssignment VALUES('X1003','10001')
INSERT INTO EquipmentAssignment VALUES('X1004','10001')
INSERT INTO EquipmentAssignment VALUES('X1005','10001')
INSERT INTO EquipmentAssignment VALUES('X2001','10003')
INSERT INTO EquipmentAssignment VALUES('X2002','10003')
INSERT INTO EquipmentAssignment VALUES('X2003','10003')
INSERT INTO EquipmentAssignment VALUES('X2004','10003')
INSERT INTO EquipmentAssignment VALUES('X2005','10003')
INSERT INTO EquipmentAssignment VALUES('X3001','10004')
INSERT INTO EquipmentAssignment VALUES('X3002','10004')
INSERT INTO EquipmentAssignment VALUES('X3003','10004')
INSERT INTO EquipmentAssignment VALUES('X3004','10004')
INSERT INTO EquipmentAssignment VALUES('X3005','10004')
INSERT INTO EquipmentAssignment VALUES('X4001','10002')
INSERT INTO EquipmentAssignment VALUES('X4002','10002')
INSERT INTO EquipmentAssignment VALUES('X4003','10002')
INSERT INTO EquipmentAssignment VALUES('X5001','10010')
INSERT INTO EquipmentAssignment VALUES('X6001','10006')
```

B. Function to run Report Wizard when "Create Report" button is clicked:

```
Public Function RunReportWizard()
On Error GoTo PROC_ERR
DoCmd.RunCommand (acCmdNewObjectReport)
PROC_EXIT:
   Exit Function
PROC_ERR:
  If Err.Number <> 2501 Then
     MsgBox "Error: (" & Err.Number & ") " & Err.Description, vbCritical
  End If
  Resume PROC_EXIT
End Function
```