IST 210 Section 003 - Team 1

Team Members: Daibo Zhang, Purna Bhattarai, Astan Doumbia, Ryan Carey

**GYM DATABASE SYSTEM (GDS)**

**Project Information**

This project is about a system that allows a business to keep track of memberships, fitness classes, schedules, and client information. This system will allow employees to register new clients, membership purchases, and merchandise. This system will list sales for each day as well as track membership rewards. This system will store the list of employees and clients. This system will be able to assign and design schedules to assign to fitness classes.

**The solution for problem solved by a database**

Scheduling conflicts are frustrating for both customers and employees. The way our requirements are structured will prevent these conflicts. Merchandise can be eliminated easily if it isn’t selling well.

**Why your project is important?**

It is important because it will keep track of class locations and timeslots to prevent scheduling conflicts. The system will also keep track of employee and customer information and memberships, inventory and sales to streamline the operations of the business. It will also allow for the ability to set and define unique schedules to assign class times.

**Users of the System**

The system will be used by employees to track sales as well as the membership status of clients. It will also be used by employees to schedule clients for classes and prevent overbooking.

**Unique aspects of your Project**

The constraints against multiple classes in one location at any given time will prevent overbooking. Recording the date of sales will allow poorly sold items to be phased out proactively. The ability to define timeslots will alow for classes to be assigned to flexible schedules that can be set to daily, weekly, bi-monthly, monthly, or even other types of schedules.

**Project Requirements**

The system should be easy to use by the end user. It should only require the entering of client purchases and only display the client bill, membership options, and scheduled classes. This database should be normalized.

**Entity Types and Attributes**

**Entities**: Client, Memberplan, Merchandise, FitClass, Timeslot, Location, Instructor

**Client**: ID, Membership Plan, Status, Name, Phone, Address, Email, Reward Points

**Memberplan**: ID, Name, Description

**Merchandise**: ID, Name, Price, Time Last Sold

**FitClass**: ID, Timeslot, Name, Category

**Timeslot**: ID, Name, Description, Start Time, End Time

**Instructor**: ID, Name, Phone

**Location**: ID, Name, Description

A client can sign up for one membership plan. A membership plan can be subscribed by multiple clients.

A client can purchase multiple items (Merchandise). An item (Merchandise) can be purchased by multiple clients.

A client can register for multiple fitness classes. A fitness class can be registered to multiple clients

An instructor can be assigned to multiple fitness classes. A fitness class can be assigned multiple instructors.

A fitness class can be held in one timeslot. A timeslot can have multiple fitness classes.

A fitness class can be held in multiple locations. A location can hold multiple fitness classes.

**Functionalities of the system:**

1. The system is going to update and store clients. Clie0pnts cannot be deleted but they can be set to inactive if they stop coming.
2. The system is going to store, update, delete membership plans. It cannot delete the membership plan if contains clients.
3. The system will store, update, and delete merchandise. Each item will have a price. If a client has purchased an item, it cannot be deleted.
4. If an item has not been bought in the past period of time, the system will update the item with date of last purchase (TimeSold)
5. The system will store, update, and delete the transactions for merchandise. Each transaction will have a time of purchase, quantity and client
6. The system will store, update, and delete client subscription of membership plans
7. The system will store, update, and delete client registration for fitness classes.
8. The system will store, update, and delete fitness class assignment for instructors
9. The system will store, update, and delete timetables(timeslot) which can be assigned to fitness classes
10. The system will store, update, and delete locations which can be assigned to fitness classes

IST 210 Section 003 - Team 1

Team Members: Daibo Zhang, Purna Bhattarai, Astan Doumbia, Ryan Carey

ER-Diagram



Translation method: Mapped  
Relational schema:

**Client** (ClientID, MemberplanID\*, Client\_Status, Client\_First\_Name, Client\_Last\_Name, Client\_Phone, Client\_Address, Client\_State, Client\_ZIP, Client\_Email, Client\_RP)

**Memberplan** (MemberplanID, Memberplan\_Name, Memberplan\_Desc)

**ClientMerchandise** (TransactionID,ClientID\*, MerchandiseID\*, Time\_Purchased, Quantity)

**Merchandise** (MerchandiseID, Merchandise\_Name, Price, Time\_Sold)

**FitClassClient** (FitClassClientID,ClientID\*, FitClassID\*)

**FitClass** (FitClassID, TimeslotID\*, FitClass\_Name, FitClass\_Category, FitClass\_Desc)

**Timeslot** (TimeslotID, Timeslot\_Name, Timeslot\_Desc, Start\_time, End\_time)

**FitClassLocation** (FitClassLocationID, FitClassID\*, LocationID\*)

**Location** (LocationID\*, Location\_Name, Location\_Desc)

**FitClassInstructor** (FitClassInstructorID,FitClassID\*, InstructorID\*)

**Instructor** (InstructorID, Instructor\_First\_Name, Instructor\_Last\_Name, Instructor\_Phone)

Oracle APEX.

**SQL Tables:**

**Memberplan Table**

CREATE TABLE "MEMBERPLAN"

( "MEMBERPLANID" NUMBER(9,0) GENERATED ALWAYS AS IDENTITY MINVALUE 1 MAXVALUE 9999999999999999999999999999 INCREMENT BY 1 START WITH 1 CACHE 20 NOORDER NOCYCLE NOKEEP NOSCALE NOT NULL ENABLE,

"MEMBERPLAN\_NAME" VARCHAR2(50) NOT NULL ENABLE,

"MEMBERPLAN\_DESC" VARCHAR2(500),

CONSTRAINT "MEMBERPLAN\_PK" PRIMARY KEY ("MEMBERPLANID")

USING INDEX ENABLE

)

/

**Client Table**

CREATE TABLE "CLIENT"

( "CLIENTID" NUMBER(9,0) GENERATED ALWAYS AS IDENTITY MINVALUE 1 MAXVALUE 9999999999999999999999999999 INCREMENT BY 1 START WITH 1 CACHE 20 NOORDER NOCYCLE NOKEEP NOSCALE NOT NULL ENABLE,

"MEMBERPLANID" NUMBER(9,0) NOT NULL ENABLE,

"CLIENT\_STATUS" VARCHAR2(25) NOT NULL ENABLE,

"CLIENT\_FIRST\_NAME" VARCHAR2(50) NOT NULL ENABLE,

"CLIENT\_LAST\_NAME" VARCHAR2(50) NOT NULL ENABLE,

"CLIENT\_PHONE" VARCHAR2(15) NOT NULL ENABLE,

"CLIENT\_ADDRESS" VARCHAR2(50) NOT NULL ENABLE,

"CLIENT\_STATE" VARCHAR2(2) NOT NULL ENABLE,

"CLIENT\_ZIP" VARCHAR2(10) NOT NULL ENABLE,

"CLIENT\_EMAIL" VARCHAR2(100) NOT NULL ENABLE,

"CLIENT\_RP" NUMBER(4,0) NOT NULL ENABLE,

CONSTRAINT "CLIENT\_PK" PRIMARY KEY ("CLIENTID")

USING INDEX ENABLE,

CONSTRAINT "CLIENT\_MEMBERPLAN\_FK" FOREIGN KEY ("MEMBERPLANID")

REFERENCES "MEMBERPLAN" ("MEMBERPLANID") ENABLE

)

/

**Merchandise Table**

CREATE TABLE "MERCHANDISE"

( "MERCHANDISEID" NUMBER(9,0) GENERATED ALWAYS AS IDENTITY MINVALUE 1 MAXVALUE 9999999999999999999999999999 INCREMENT BY 1 START WITH 1 CACHE 20 NOORDER NOCYCLE NOKEEP NOSCALE NOT NULL ENABLE,

"MERCHANDISE\_NAME" VARCHAR2(50) NOT NULL ENABLE,

"PRICE" NUMBER(9,2) NOT NULL ENABLE,

"TIME\_SOLD" DATE,

CONSTRAINT "MERCHANDISE\_PK" PRIMARY KEY ("MERCHANDISEID")

USING INDEX ENABLE

)

/

**CLIENTMERCHANDISE Table**

CREATE TABLE "CLIENTMERCHANDISE"

( "CLIENTID" NUMBER(9,0) NOT NULL ENABLE,

"MERCHANDISEID" NUMBER(9,0) NOT NULL ENABLE,

"TIME\_PURCHASED" DATE NOT NULL ENABLE,

"QUANTITY" NUMBER(9,0) NOT NULL ENABLE,

"TRANSACTIONID" NUMBER(9,0) GENERATED ALWAYS AS IDENTITY MINVALUE 1 MAXVALUE 9999999999999999999999999999 INCREMENT BY 1 START WITH 1 CACHE 20 NOORDER NOCYCLE NOKEEP NOSCALE NOT NULL ENABLE,

CONSTRAINT "CLIENTMERCHANDISE\_PK" PRIMARY KEY ("TRANSACTIONID")

USING INDEX ENABLE

)

/

ALTER TABLE "CLIENTMERCHANDISE" ADD CONSTRAINT "CLIENTMERCHANDISE\_CLIENT\_FK" FOREIGN KEY ("CLIENTID")

REFERENCES "CLIENT" ("CLIENTID") ENABLE

/

ALTER TABLE "CLIENTMERCHANDISE" ADD CONSTRAINT "CLIENTMERCHANDISE\_MERCHANDISE\_FK" FOREIGN KEY ("MERCHANDISEID")

REFERENCES "MERCHANDISE" ("MERCHANDISEID") ENABLE

/

**Timeslot Table**

CREATE TABLE "TIMESLOT"

( "TIMESLOTID" NUMBER(9,0) GENERATED ALWAYS AS IDENTITY MINVALUE 1 MAXVALUE 9999999999999999999999999999 INCREMENT BY 1 START WITH 1 CACHE 20 NOORDER NOCYCLE NOKEEP NOSCALE NOT NULL ENABLE,

"TIMESLOT\_NAME" VARCHAR2(50) NOT NULL ENABLE,

"TIMESLOT\_DESC" VARCHAR2(500) NOT NULL ENABLE,

"START\_TIME" NUMBER(4,0) NOT NULL ENABLE,

"END\_TIME" NUMBER(4,0) NOT NULL ENABLE,

CONSTRAINT "TIMESLOT\_PK" PRIMARY KEY ("TIMESLOTID")

USING INDEX ENABLE

)

/

**Instructor Table**

CREATE TABLE "INSTRUCTOR"

( "INSTRUCTORID" NUMBER(9,0) GENERATED ALWAYS AS IDENTITY MINVALUE 1 MAXVALUE 9999999999999999999999999999 INCREMENT BY 1 START WITH 1 CACHE 20 NOORDER NOCYCLE NOKEEP NOSCALE NOT NULL ENABLE,

"INSTRUCTOR\_FIRST\_NAME" VARCHAR2(50) NOT NULL ENABLE,

"INSTRUCTOR\_LAST\_NAME" VARCHAR2(50) NOT NULL ENABLE,

"INSTRUCTOR\_PHONE" VARCHAR2(15) NOT NULL ENABLE,

CONSTRAINT "INSTRUCTOR\_PK" PRIMARY KEY ("INSTRUCTORID")

USING INDEX ENABLE

)

/

**Location Table**

CREATE TABLE "LOCATION"

( "LOCATIONID" NUMBER(9,0) GENERATED ALWAYS AS IDENTITY MINVALUE 1 MAXVALUE 9999999999999999999999999999 INCREMENT BY 1 START WITH 1 CACHE 20 NOORDER NOCYCLE NOKEEP NOSCALE NOT NULL ENABLE,

"LOCATION\_NAME" VARCHAR2(50) NOT NULL ENABLE,

"DESCRIPTION" VARCHAR2(500),

CONSTRAINT "LOCATION\_PK" PRIMARY KEY ("LOCATIONID")

USING INDEX ENABLE

)

/

**FitClass Table**

CREATE TABLE "FITCLASS"

( "FITCLASSID" NUMBER(9,0) GENERATED ALWAYS AS IDENTITY MINVALUE 1 MAXVALUE 9999999999999999999999999999 INCREMENT BY 1 START WITH 1 CACHE 20 NOORDER NOCYCLE NOKEEP NOSCALE NOT NULL ENABLE,

"TIMESLOTID" NUMBER(9,0) NOT NULL ENABLE,

"FITCLASS\_NAME" VARCHAR2(50) NOT NULL ENABLE,

"FITCLASS\_CATEGORY" VARCHAR2(50),

"FITCLASS\_DESC" VARCHAR2(500),

CONSTRAINT "FITCLASS\_PK" PRIMARY KEY ("FITCLASSID")

USING INDEX ENABLE,

CONSTRAINT "FITCLASS\_TIMESLOT\_FK" FOREIGN KEY ("TIMESLOTID")

REFERENCES "TIMESLOT" ("TIMESLOTID") ENABLE

)

/

**FitClassClient Table**

CREATE TABLE "FITCLASSCLIENT"

( "FITCLASSCLIENTID" NUMBER (9,0) GENERATED ALWAYS AS IDENTITY MINVALUE 1 MAXVALUE 9999999999999999999999999999 INCREMENT BY 1 START WITH 1 CACHE 20 NOORDER NOCYCLE NOKEEP NOSCALE NOT NULL ENABLE,

"CLIENTID" NUMBER(9,0) NOT NULL ENABLE,

"FITCLASSID" NUMBER(9,0) NOT NULL ENABLE,

CONSTRAINT "FITCLASSCLIENT\_PK" PRIMARY KEY ("FITCLASSCLIENTID")

USING INDEX ENABLE,

CONSTRAINT "FITCLASSCLIENT\_CLIENT\_FK" FOREIGN KEY ("CLIENTID")

REFERENCES "CLIENT" ("CLIENTID") ENABLE,

CONSTRAINT "FITCLASSCLIENT\_FITCLASS\_FK" FOREIGN KEY ("FITCLASSID")

REFERENCES "FITCLASS" ("FITCLASSID") ENABLE

)

/

**FitClassInstructor Table**

CREATE TABLE "FITCLASSINSTRUCTOR"

( "FITCLASSINSTRUCTORID" NUMBER (9,0) GENERATED ALWAYS AS IDENTITY MINVALUE 1 MAXVALUE 9999999999999999999999999999 INCREMENT BY 1 START WITH 1 CACHE 20 NOORDER NOCYCLE NOKEEP NOSCALE NOT NULL ENABLE,

"FITCLASSID" NUMBER(9,0) NOT NULL ENABLE,

"INSTRUCTORID" NUMBER(9,0) NOT NULL ENABLE,

CONSTRAINT "FITCLASSINSTRUCTOR\_PK" PRIMARY KEY ("FITCLASSINSTRUCTORID")

USING INDEX ENABLE,

CONSTRAINT "FITCLASSINSTRUCTOR\_INSTRUCTOR\_FK" FOREIGN KEY ("INSTRUCTORID")

REFERENCES "INSTRUCTOR" ("INSTRUCTORID") ENABLE,

CONSTRAINT "FITCLASSINSTRUCTOR\_FITCLASS\_FK" FOREIGN KEY ("FITCLASSID")

REFERENCES "FITCLASS" ("FITCLASSID") ENABLE

)

/

**FitClassLocation Table**

CREATE TABLE "FITCLASSLOCATION"

( "FITCLASSLOCATIONID" NUMBER (9,0) GENERATED ALWAYS AS IDENTITY MINVALUE 1 MAXVALUE 9999999999999999999999999999 INCREMENT BY 1 START WITH 1 CACHE 20 NOORDER NOCYCLE NOKEEP NOSCALE NOT NULL ENABLE,

"FITCLASSID" NUMBER(9,0) NOT NULL ENABLE,

"LOCATIONID" NUMBER(9,0) NOT NULL ENABLE,

CONSTRAINT "FITCLASSLOCATION\_PK" PRIMARY KEY ("FITCLASSLOCATIONID")

USING INDEX ENABLE,

CONSTRAINT "FITCLASSLOCATION\_LOCATION\_FK" FOREIGN KEY ("LOCATIONID")

REFERENCES "LOCATION" ("LOCATIONID") ENABLE,

CONSTRAINT "FITCLASSLOCATION\_FITCLASS\_FK" FOREIGN KEY ("FITCLASSID")

REFERENCES "FITCLASS" ("FITCLASSID") ENABLE

)

/