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**

Recording the date of sales will allow poorly sold items to be phased out proactively. The ability to define timeslots will allow 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)

**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)

**Data Dictionary**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Team: Team 1** | **Team Members: Daibo Zhang, Purna Bhattarai, Astan Doumbia, Ryan Carey** |  |  |  |  |  |
| **TABLE NAME** | **TABLE DESCRIPTION** | **# of Attributes** |  |  |  |  |
| **Client** | Contains information about clients | 11 |  |  |  |  |
| **Memberplan** | Contains information about membership plans | 3 |  |  |  |  |
| **ClientMerchandise** | Contains information about merchandise purchased by clients | 5 |  |  |  |  |
| **Merchandise** | Contains information about merchandise | 3 |  |  |  |  |
| **FitClassClient** | Contains information about clients in fitness classes | 3 |  |  |  |  |
| **FitClass** | Contains information about fitness classes | 5 |  |  |  |  |
| **Timeslot** | Contains information about timeslots | 5 |  |  |  |  |
| **FitClassLocation** | Contains information about fitness class locations | 3 |  |  |  |  |
| **Location** | Contains information about locations | 3 |  |  |  |  |
| **FitClassInstructor** | Contains information about instructor assigned to fitness classes | 3 |  |  |  |  |
| **Instructor** | Contains information about instructors | 4 |  |  |  |  |
|  |  |  |  |  |  |  |
| **Client TABLE** |  |  |  |  |  |  |
| **ATTRIBUTE NAME** | **ATTRIBUTE DESCRIPTION** | **Domain** | **Range** | **PK** | **FK** | **Null** |
| ClientID | System Generated unique number used to identify a client | number(9,0) | ALL | Yes | No | No |
| MemberplanID | Number used to identify a membership plan | number(9,0) | ALL | No | Yes | No |
| Client\_Status | Activity Status of client | varchar2(25) | ALL | No | No | No |
| Client\_First\_Name | First name of client | varchar2(50) | ALL | No | No | No |
| Client\_Last\_Name | Last name of client | varchar2(50) | ALL | No | No | No |
| Client\_Phone | Phone number of client | varchar2(15) | ALL | No | No | No |
| Client\_Address | Street Address of Client | varchar2(50) | ALL | No | No | No |
| Client\_State | State of residence of client | varchar2(2) | ALL | No | No | No |
| Client\_ZIP | Zipcode of client | varchar2(10) | ALL | No | No | No |
| Client\_Email | Email of client | varchar2(100) | ALL | No | No | No |
| Client\_RP | Reward points of client | number(4,0) | ALL | No | No | No |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
| **FitClass TABLE** |  |  |  |  |  |  |
| **ATTRIBUTE NAME** | **ATTRIBUTE DESCRIPTION** | **Domain** | **Range** | **PK** | **FK** | **Null** |
| FitClassID | System Generated unique number used to identify a Fitness class | number(9,0) | ALL | Yes | No | No |
| TimeslotID | Timeslot Identification Number | number(9,0) | ALL | No | Yes | No |
| FitClass\_Name | Name of fitness class | varchar2(50) | ALL | No | No | No |
| FitClass\_Category | Category of the Fitness Class | varchar2(50) | ALL | No | No | Yes |
| FitClass\_Desc | Description of the Class | varchar2(500) | ALL | No | No | Yes |
|  |  |  |  |  |  |  |
| **FitClassClient TABLE** |  |  |  |  |  |  |
| **ATTRIBUTE NAME** | **ATTRIBUTE DESCRIPTION** | **Domain** | **Range** | **PK** | **FK** | **Null** |
| FitClassClientID | System Generated Unique number used to identify client instructor assignments | number(9,0) | ALL | Yes | No | No |
| ClientID | ID used to identify a client | number(9,0) | ALL | No | Yes | No |
| FitClassID | ID used to identify a fitness class | number(9,0) | ALL | No | Yes | No |
|  |  |  |  |  |  |  |
| **FitClassInstructor TABLE** |  |  |  |  |  |  |
| **ATTRIBUTE NAME** | **ATTRIBUTE DESCRIPTION** | **Domain** | **Range** | **PK** | **FK** | **Null** |
| FitClassInstructorID | System Generated Unique number used to identify instructor assignments | number(9,0) | ALL | Yes | No | No |
| FitClassID | ID used to identify a Fitness class | number(9,0) | ALL | No | Yes | No |
| InstructorID | ID used to identify an Instructor | number(9,0) | ALL | No | Yes | No |
|  |  |  |  |  |  |  |
| **FitClassLocation TABLE** |  |  |  |  |  |  |
| **ATTRIBUTE NAME** | **ATTRIBUTE DESCRIPTION** | **Domain** | **Range** | **PK** | **FK** | **Null** |
| FitClassLocationID | System Generated Unique number used to identify class location assignments | number(9,0) | ALL | Yes | No | No |
| FitClassID | ID used to identify a Fitness class | number(9,0) | ALL | No | Yes | No |
| LocationID | ID used to identify a Location | number(9,0) | ALL | No | Yes | No |
|  |  |  |  |  |  |  |
| **Instructor TABLE** |  |  |  |  |  |  |
| **ATTRIBUTE NAME** | **ATTRIBUTE DESCRIPTION** | **Domain** | **Range** | **PK** | **FK** | **Null** |
| InstructorID | System Generated unique number use to identify an Instructor | number(9,0) | ALL | Yes | No | No |
| Instructor\_First\_Name | First name of instructor | varchar2(50) | ALL | No | No | No |
| Instructor\_Last\_Name | Last name of instructor | varchar2(50) | ALL | No | No | No |
| Instructor\_Phone | Phone number of instructor | varchar2(15) | ALL | No | No | No |
|  |  |  |  |  |  |  |
| **Location TABLE** |  |  |  |  |  |  |
| **ATTRIBUTE NAME** | **ATTRIBUTE DESCRIPTION** | **Domain** | **Range** | **PK** | **FK** | **Null** |
| LocationID | System Generated unique number used to identify a Location | number(9,0) | ALL | Yes | No | No |
| Location\_Name | Name of Location | varchar2(50) | ALL | No | No | No |
| Description | Description of Location | varchar(500) | ALL | No | No | Yes |
|  |  |  |  |  |  |  |
| **Memberplan TABLE** |  |  |  |  |  |  |
| **ATTRIBUTE NAME** | **ATTRIBUTE DESCRIPTION** | **Domain** | **Range** | **PK** | **FK** | **Null** |
| MemberplanID | System Generated unique number used to identify a Membership plan | number(9,0) | ALL | Yes | No | No |
| Memberplan\_Name | Name of membership plan | varchar2(50) | ALL | No | No | No |
| Memberplan\_Desc | Description of the membership plan | varchar2(500) | ALL | No | No | Yes |
|  |  |  |  |  |  |  |
| **Merchandise TABLE** |  |  |  |  |  |  |
| **ATTRIBUTE NAME** | **ATTRIBUTE DESCRIPTION** | **Domain** | **Range** | **PK** | **FK** | **Null** |
| MerchandiseID | System Generated unique number used to identify Merchandise | number(9,0) | ALL | Yes | No | No |
| Merchandise\_Name | Name of Merchandise | varchar(50) | ALL | No | No | No |
| Price | Price of the merchandise | number(9,2) | ALL | No | No | No |
|  |  |  |  |  |  |  |
| **Timeslot TABLE** |  |  |  |  |  |  |
| **ATTRIBUTE NAME** | **ATTRIBUTE DESCRIPTION** | **Domain** | **Range** | **PK** | **FK** | **Null** |
| TimeslotID | System Generated unique number used to identify a Timeslot | number(9,0) | ALL | Yes | No | No |
| Timeslot\_Name | Name of the timeslot | varchar2(50) | ALL | No | No | No |
| Timeslot\_Desc | Description of timeslot frequency or schedule | varchar2(500) | ALL | No | No | No |
| Start\_time | Start time of the timeslot (0000-2400) | number(4,0) | ALL | No | No | No |
| End\_time | End time the class ends (0000-2400) | number(4,0) | ALL | No | No | No |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
| **ClientMerchandise TABLE** |  |  |  |  |  |  |
| **ATTTRIBUTE NAME** | **ATTRIBUTE DESCRIPTION** | **Domain** | **Range** | **PK** | **FK** | **Null** |
| TransationID | System Generated unique number used to identify a transation | number(9,0) | ALL | Yes | No | No |
| ClientID | System Generated unique number used to identify a client | number(9,0) | ALL | No | Yes | No |
| MerchandiseID | System Generated unique number used to identify Merchandise | number(9,0) | ALL | No | Yes | No |
| Time\_Purchased | Time when merchandise was sold | date | ALL | No | No | No |
| Quantity | Total quanitity of merchandise sold | number(9,0) | ALL | No | No | No |

**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,

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 "FITCLASS\_CLIENT\_UNIQUE" UNIQUE ("FITCLASSID", "CLIENTID")

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 "FITCLASS\_INSTRUCTOR\_UNIQUE" UNIQUE ("FITCLASSID", "INSTRUCTORID")

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 "FITCLASS\_LOCATION\_UNIQUE" UNIQUE ("FITCLASSID", "LOCATIONID")

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

)

/

**Gym Database Application**

Team name: Team 1

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

Write one page narrative here. Briefly talk about how your application achieves the requirements specified in your project description and requirements document.

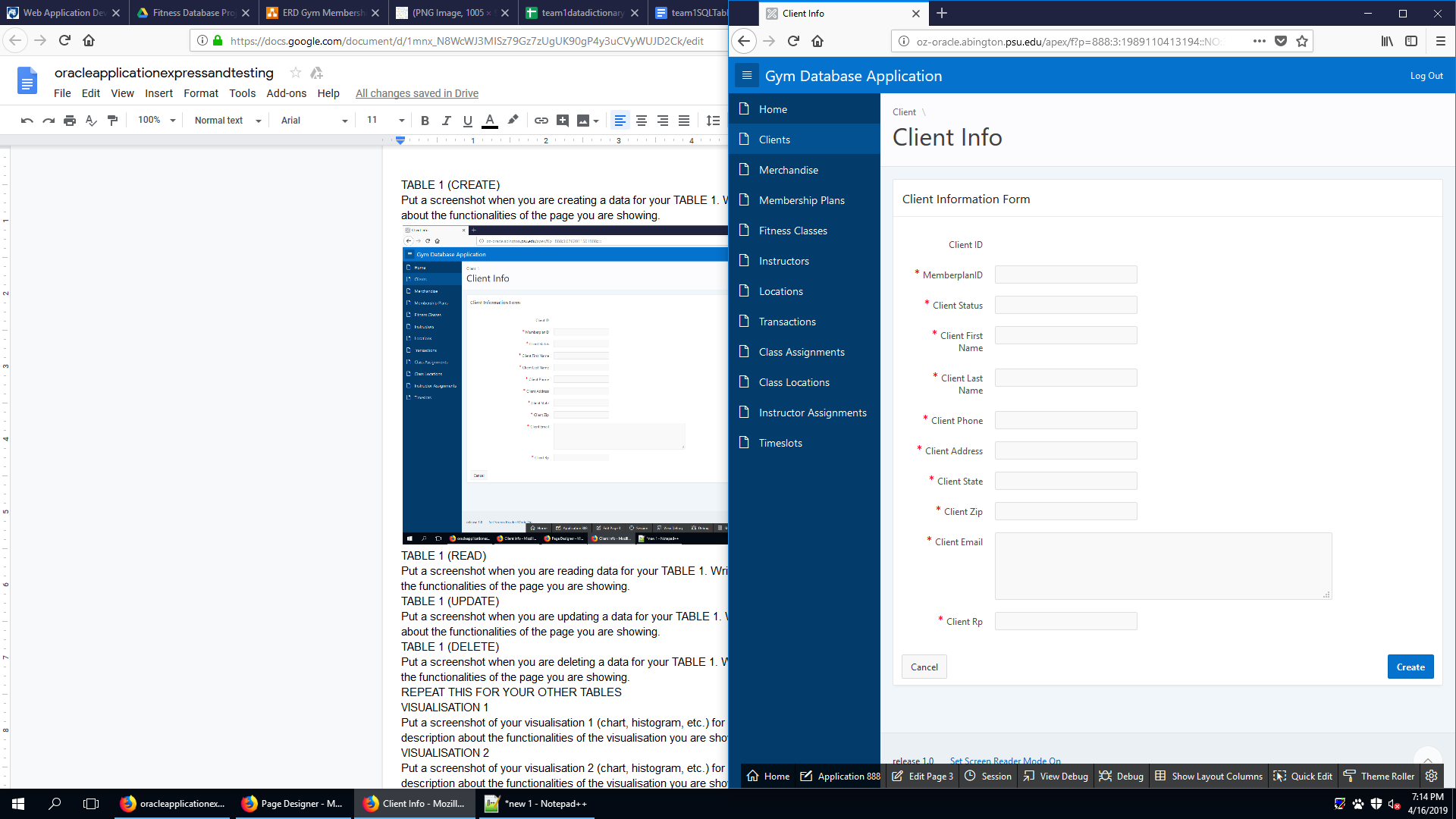
This system allows a business to keep track of memberships, fitness classes, schedules, and client information. This application allows employees to register new clients, membership purchases, and merchandise. It is capable of listing the sales for each day as well as tracking membership rewards. The application also stores lists of employees and clients. This system will be able to assign and design schedules to assign to fitness classes.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.

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

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

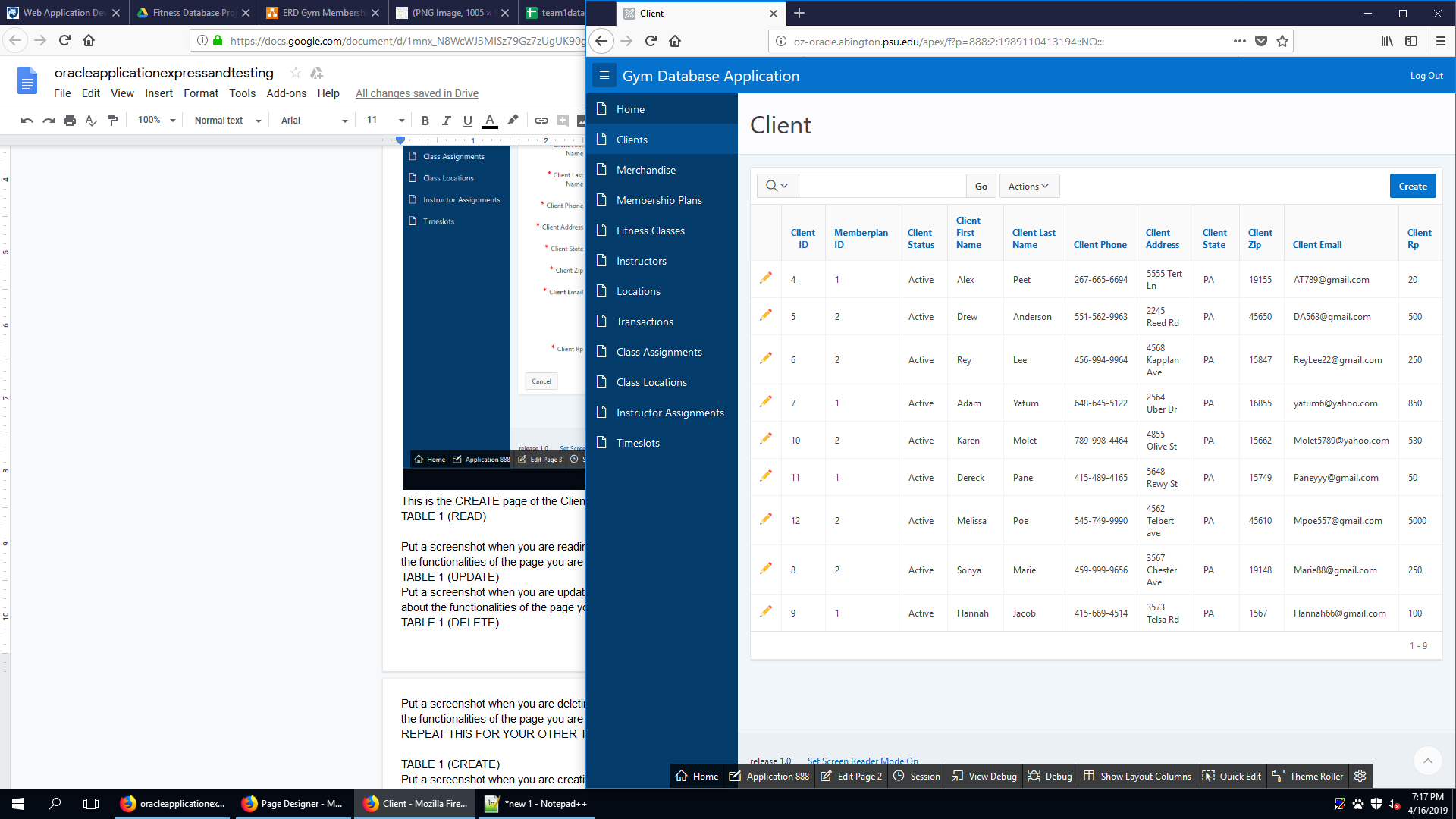
In summary our application provides an easily managed interface to all the information critical to the gym’s operation.

CLIENT (CREATE)



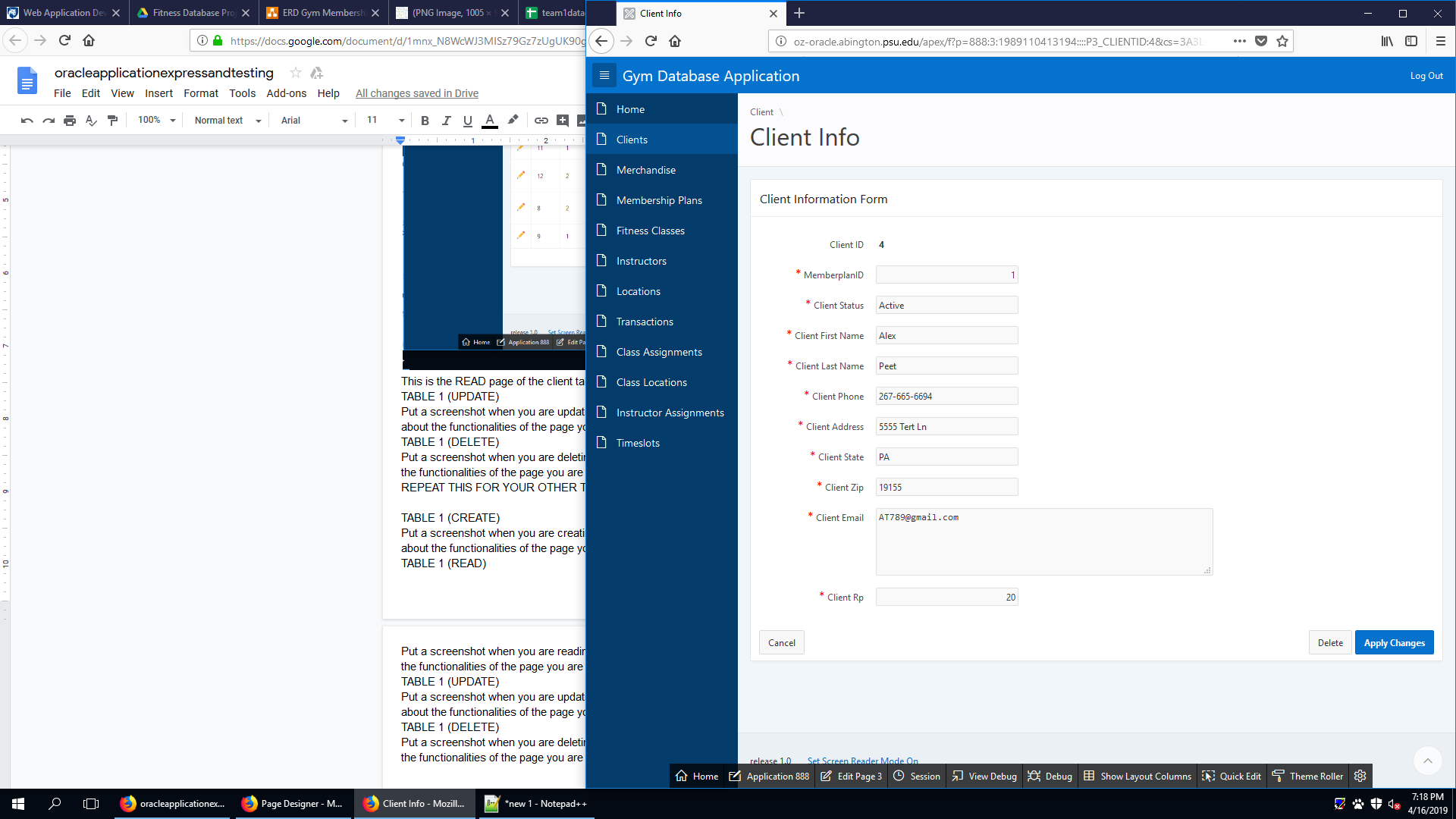
This is the CREATE page of the Clients table, it allows the entry of new clients.

CLIENT (READ)



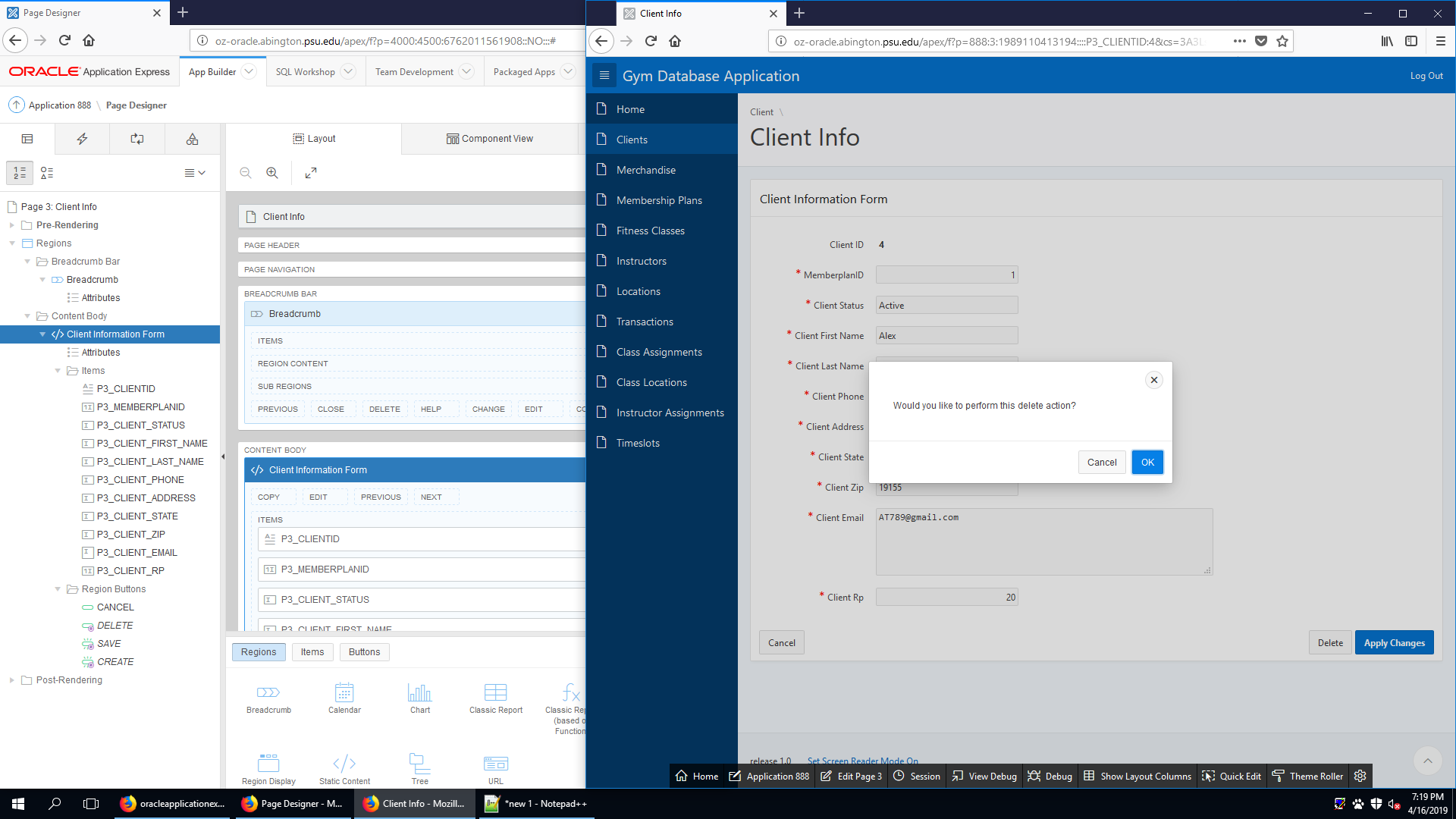
This is the READ page of the client table. It allows for the user to see all the clients

CLIENT (UPDATE)



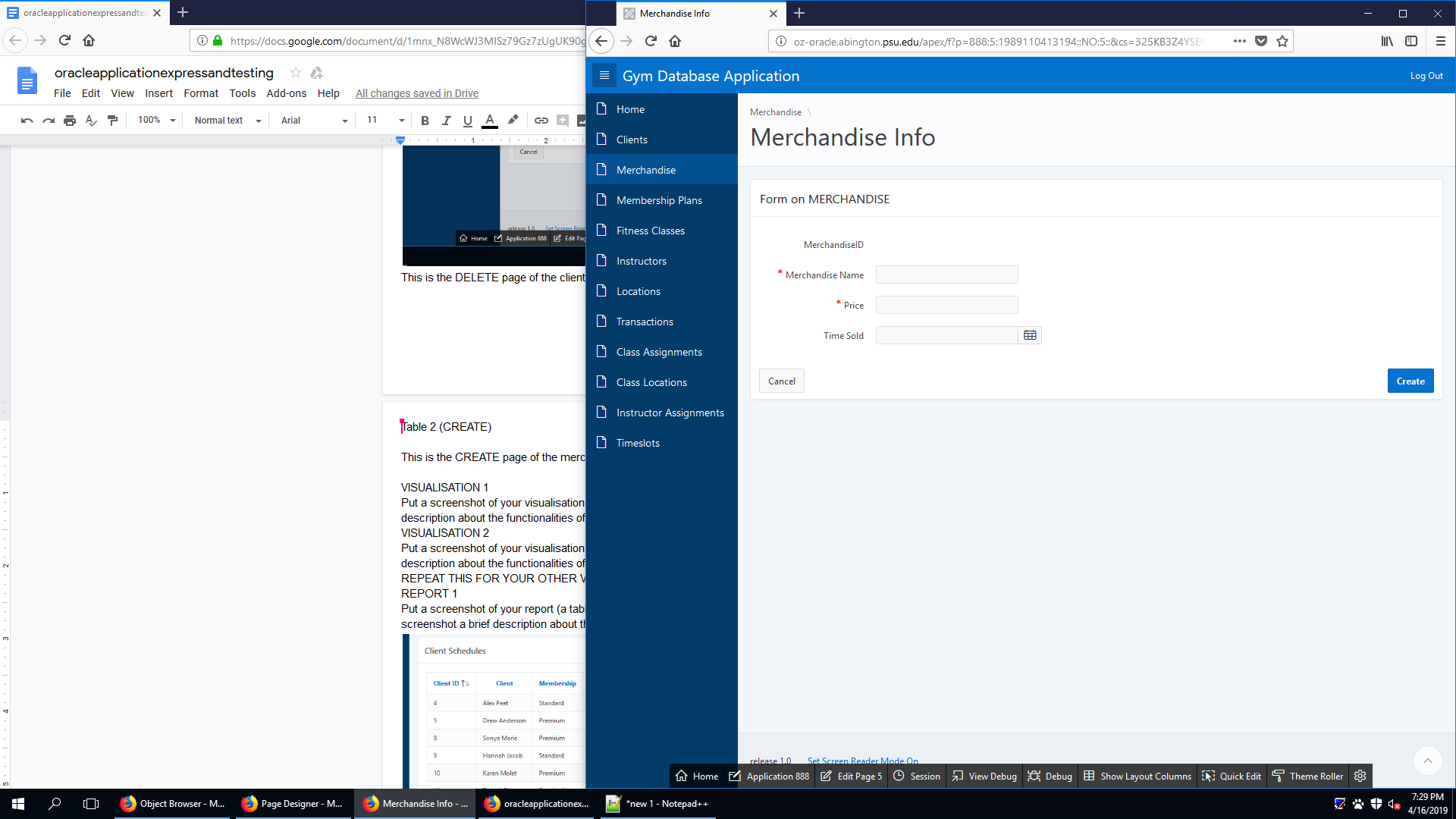
This is the UPDATE page of the client table. It allows for the user to update a currently existing customer.

CLIENT (DELETE)



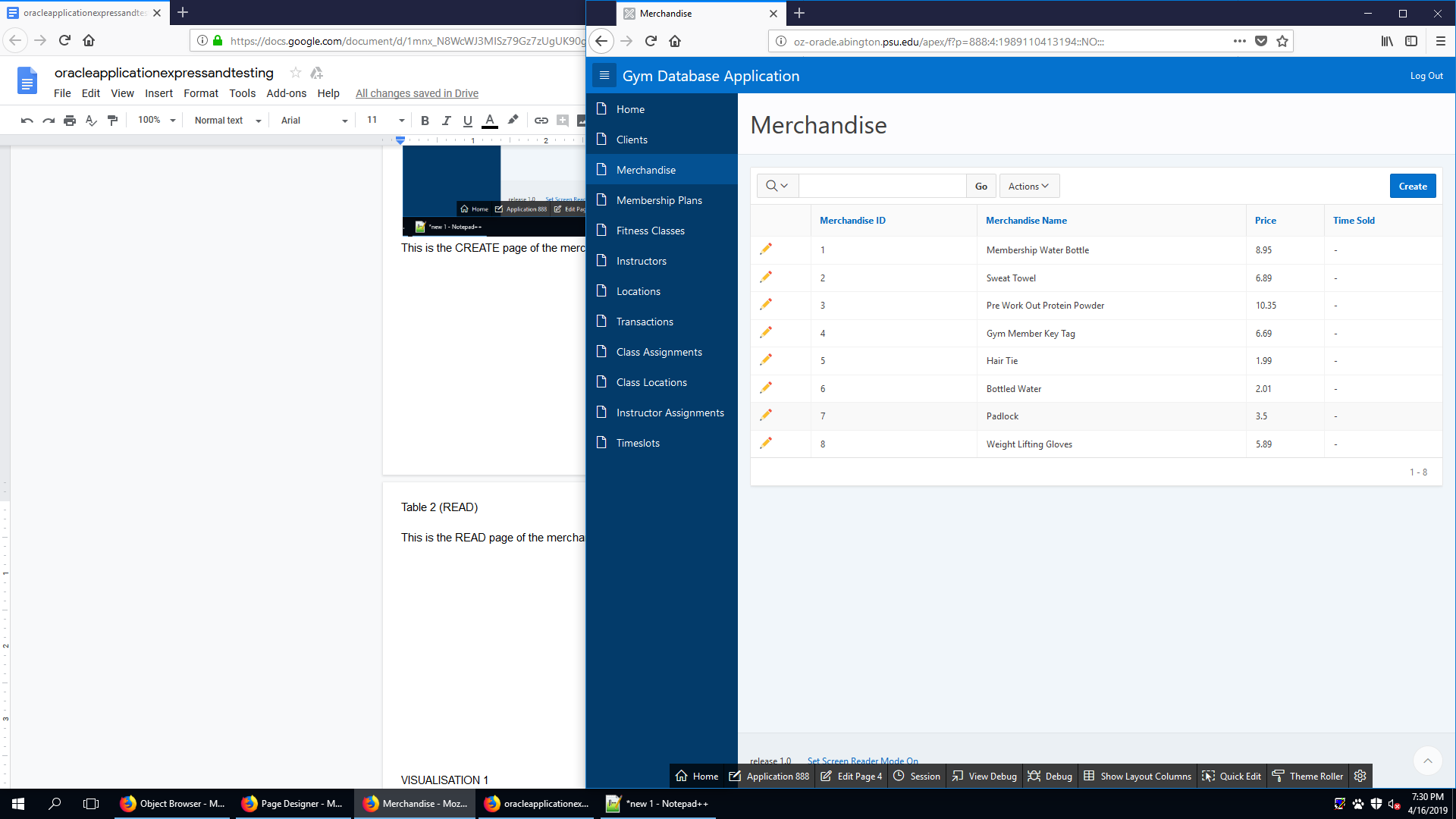
This is the DELETE page of the client table. It allows the user to delete an existing client.

MERCHANDISE (CREATE)



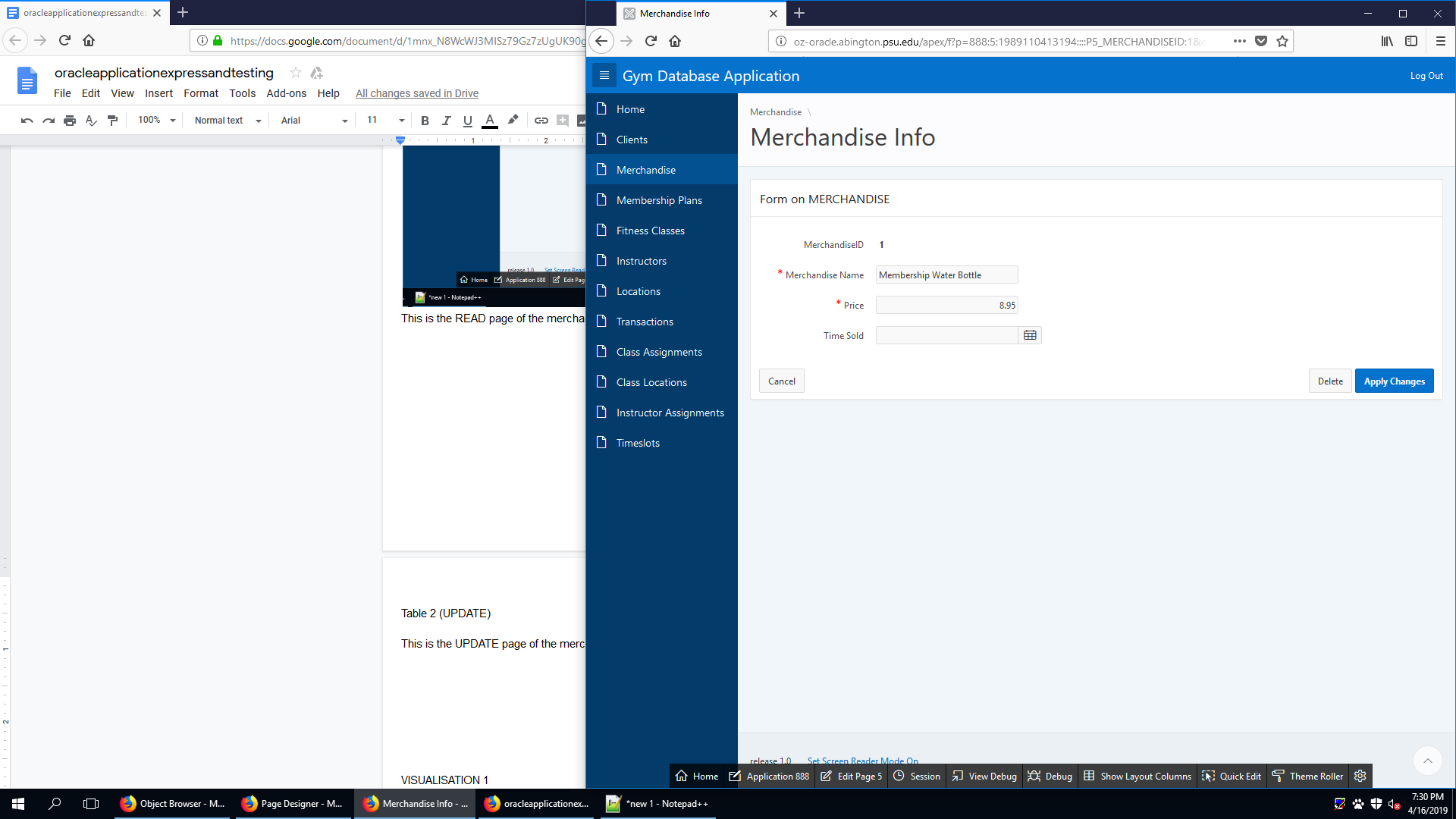
This is the CREATE page of the merchandise table. It allows the user to create new merchandise.

MERCHANDISE (READ)



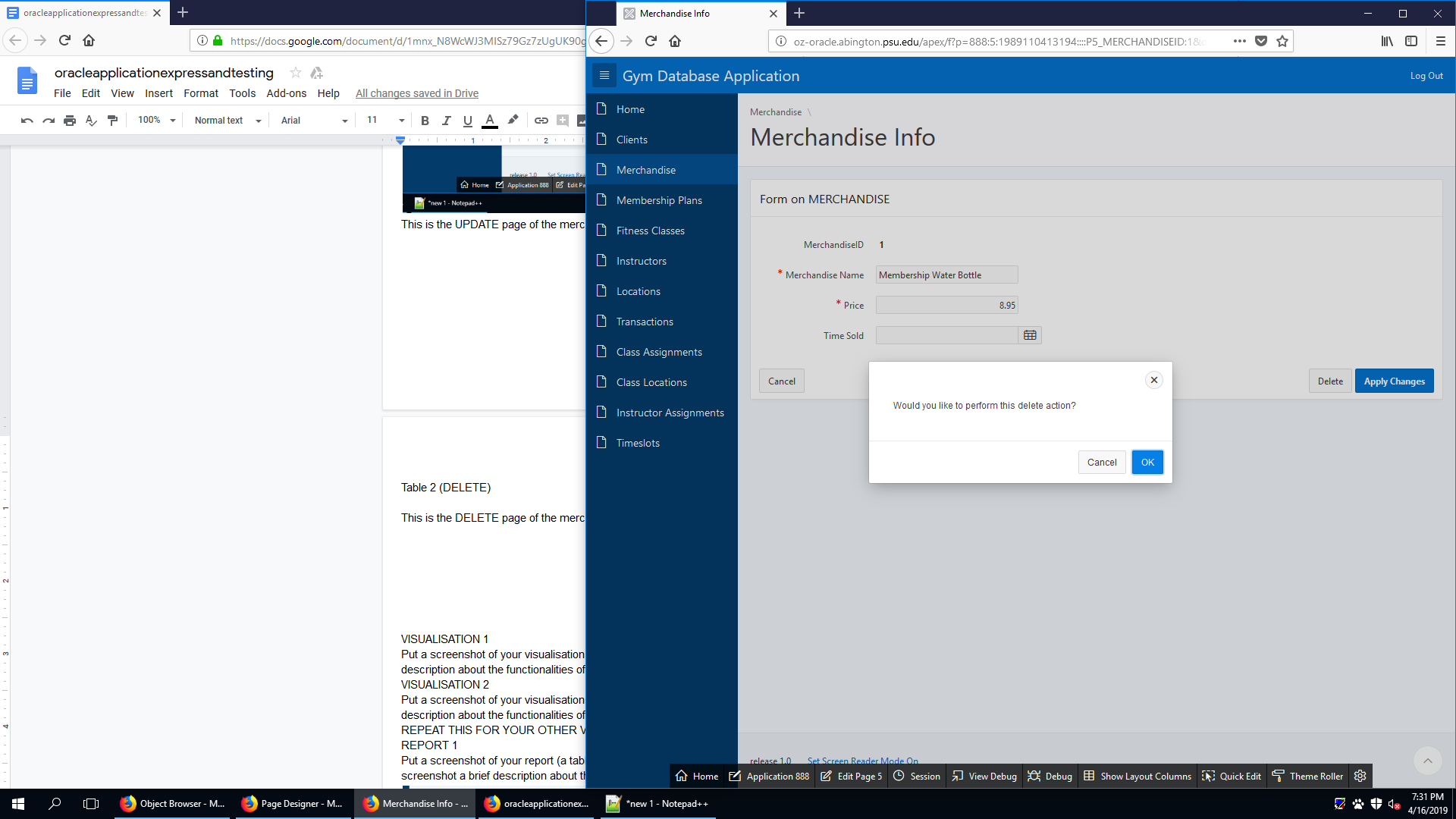
This is the READ page of the merchandise table. It allows the user to see existing merchandise.

MERCHANDISE (UPDATE)



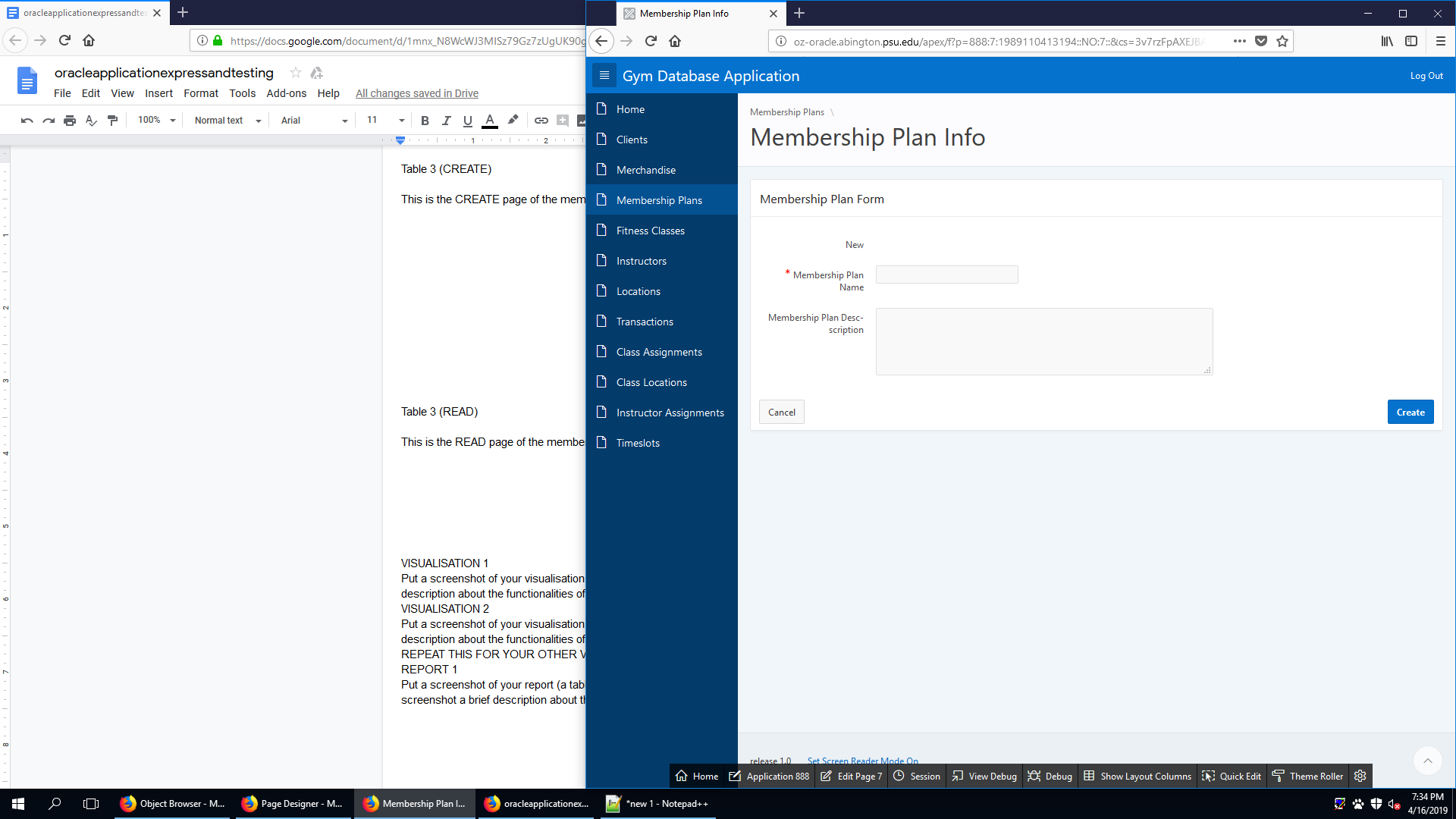
This is the UPDATE page of the merchandise table. It allows the user to update an existing merchandise.

MERCANDISE (DELETE)



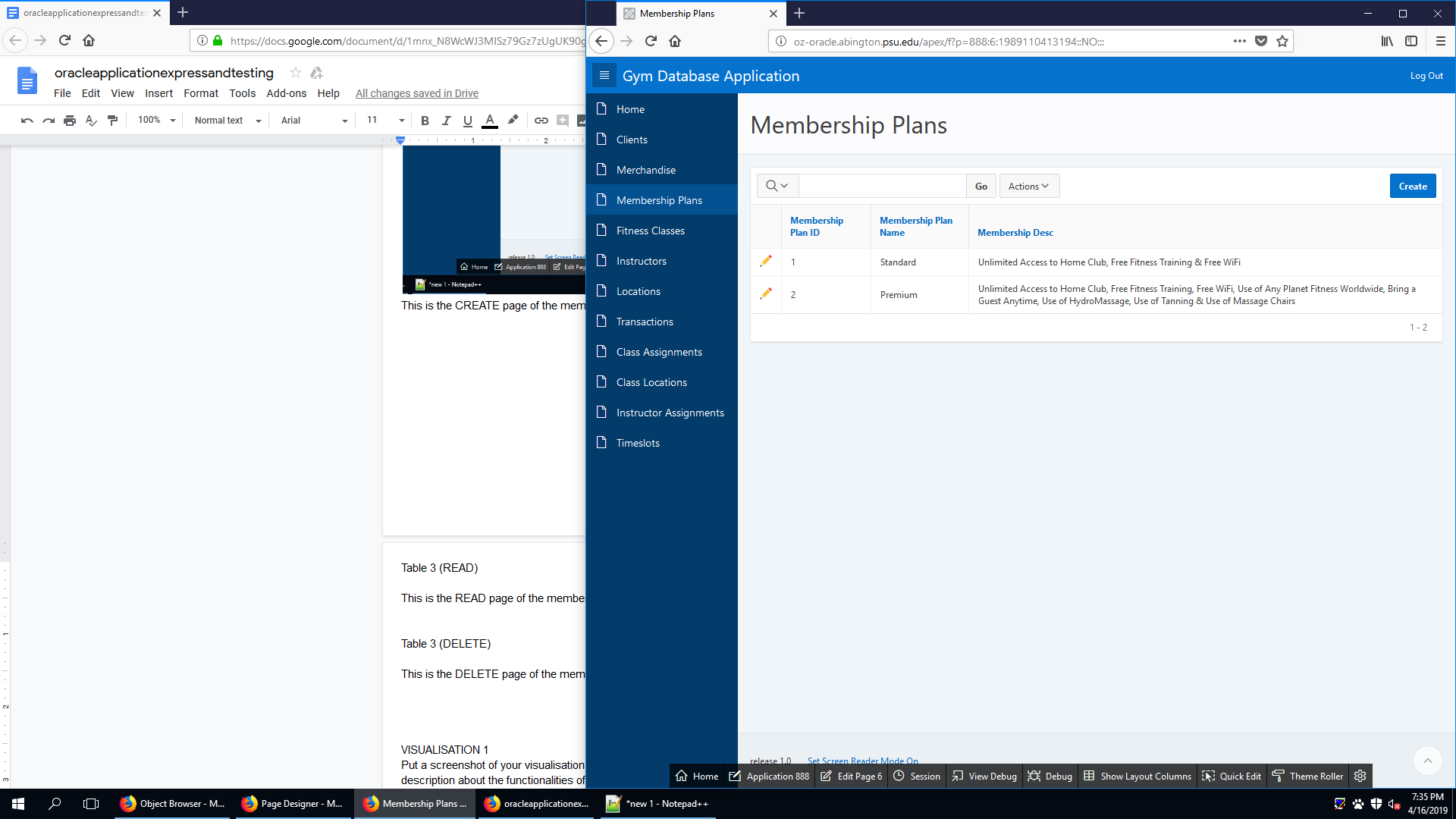
This is the DELETE page of the merchandise table. It allows the user to delete an existing merchandise.

MEMBERPLAN (CREATE)



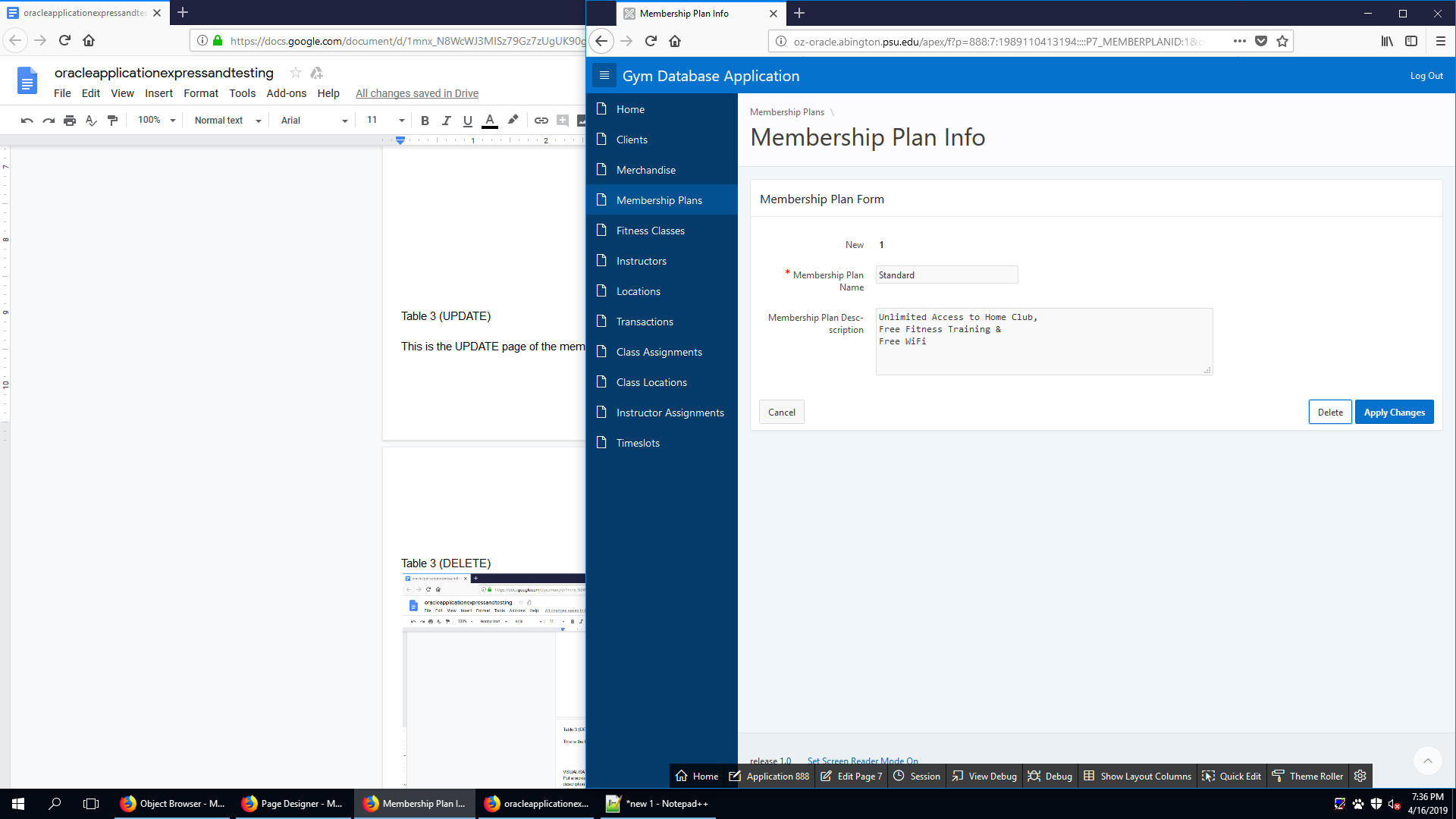
This is the CREATE page of the membership plans table. It allows the user to create new membership plans.

MEMBERPLAN (READ)



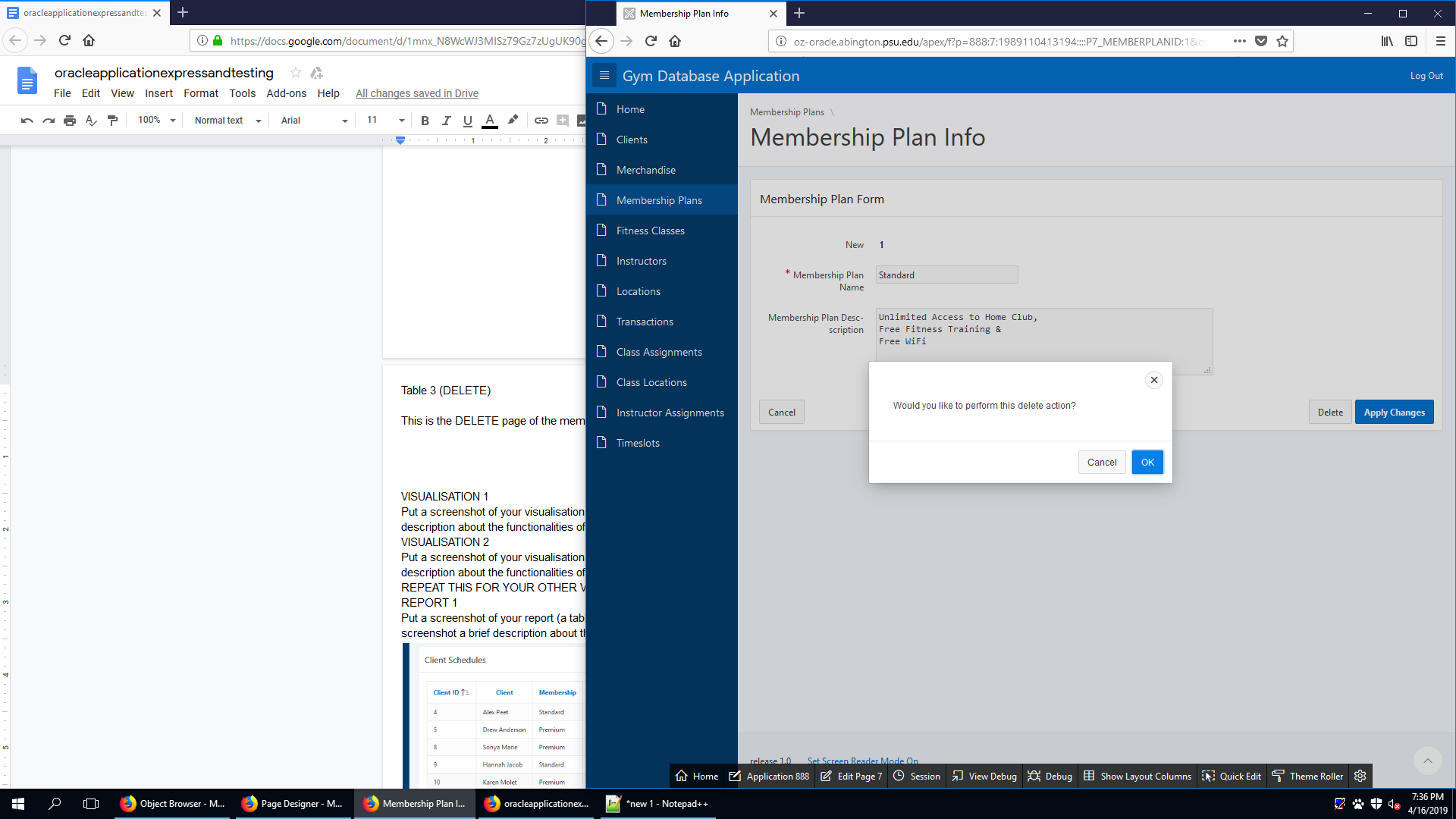
This is the READ page of the membership plans table. It allows the user to see existing membership plans.

MEMBERPLAN (UPDATE)



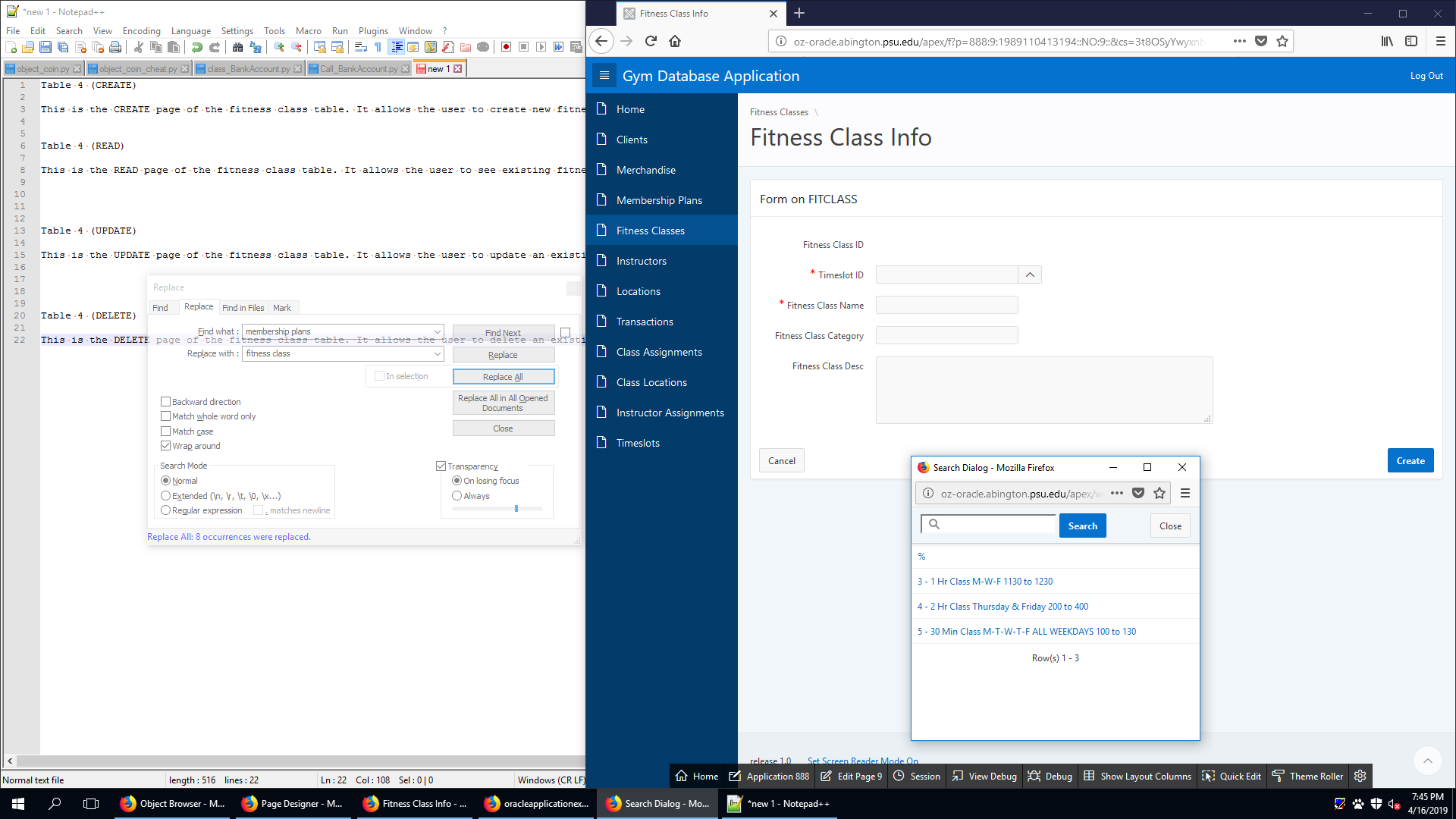
This is the UPDATE page of the membership plans table. It allows the user to update an existing membership plans.

MEMBERPLAN (DELETE)



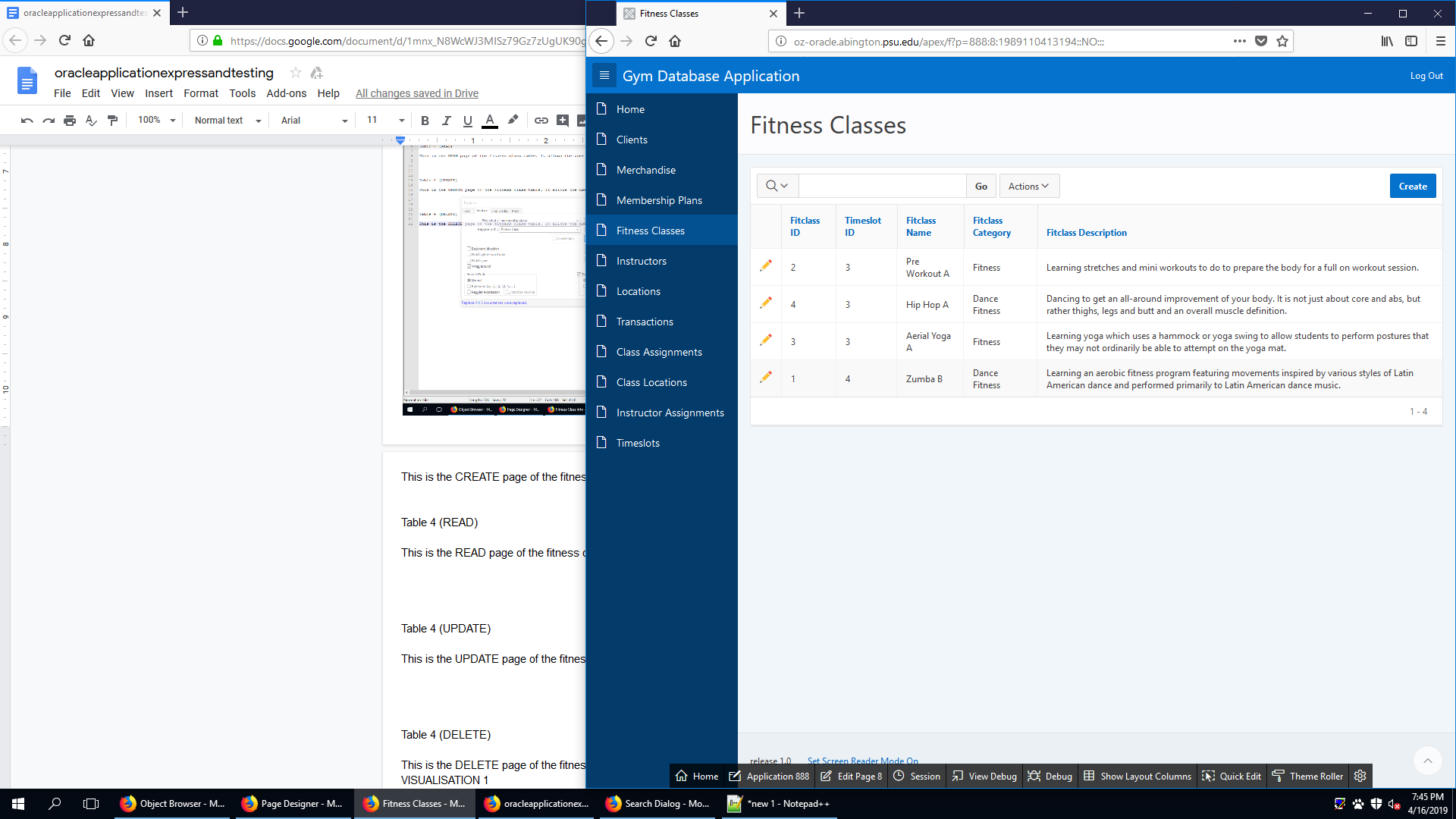
This is the DELETE page of the membership plans table. It allows the user to delete existing membership plans.

FITCLASS (CREATE)



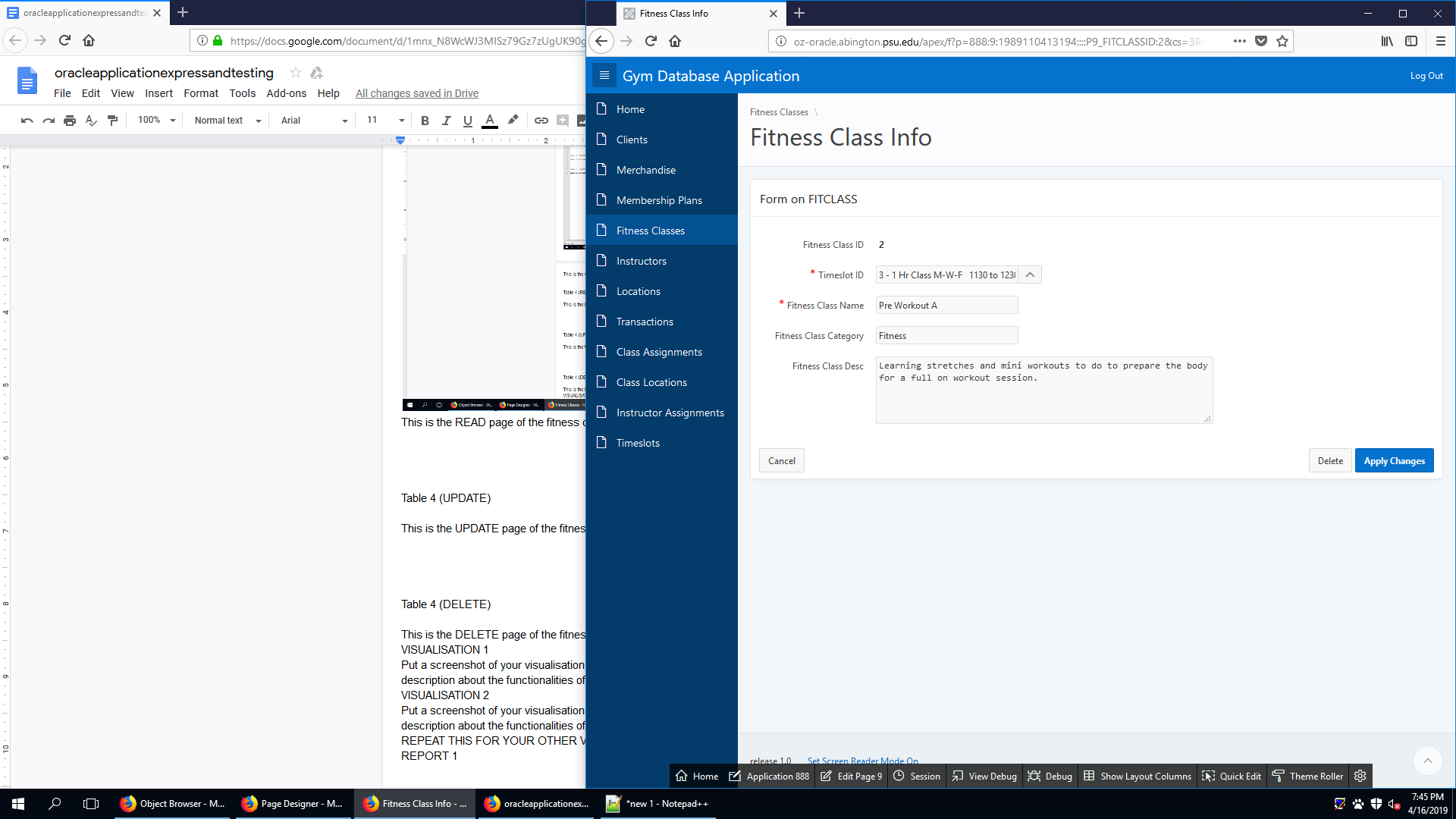
This is the CREATE page of the fitness class table. It allows the user to create a new fitness class.

FITCLASS (READ)



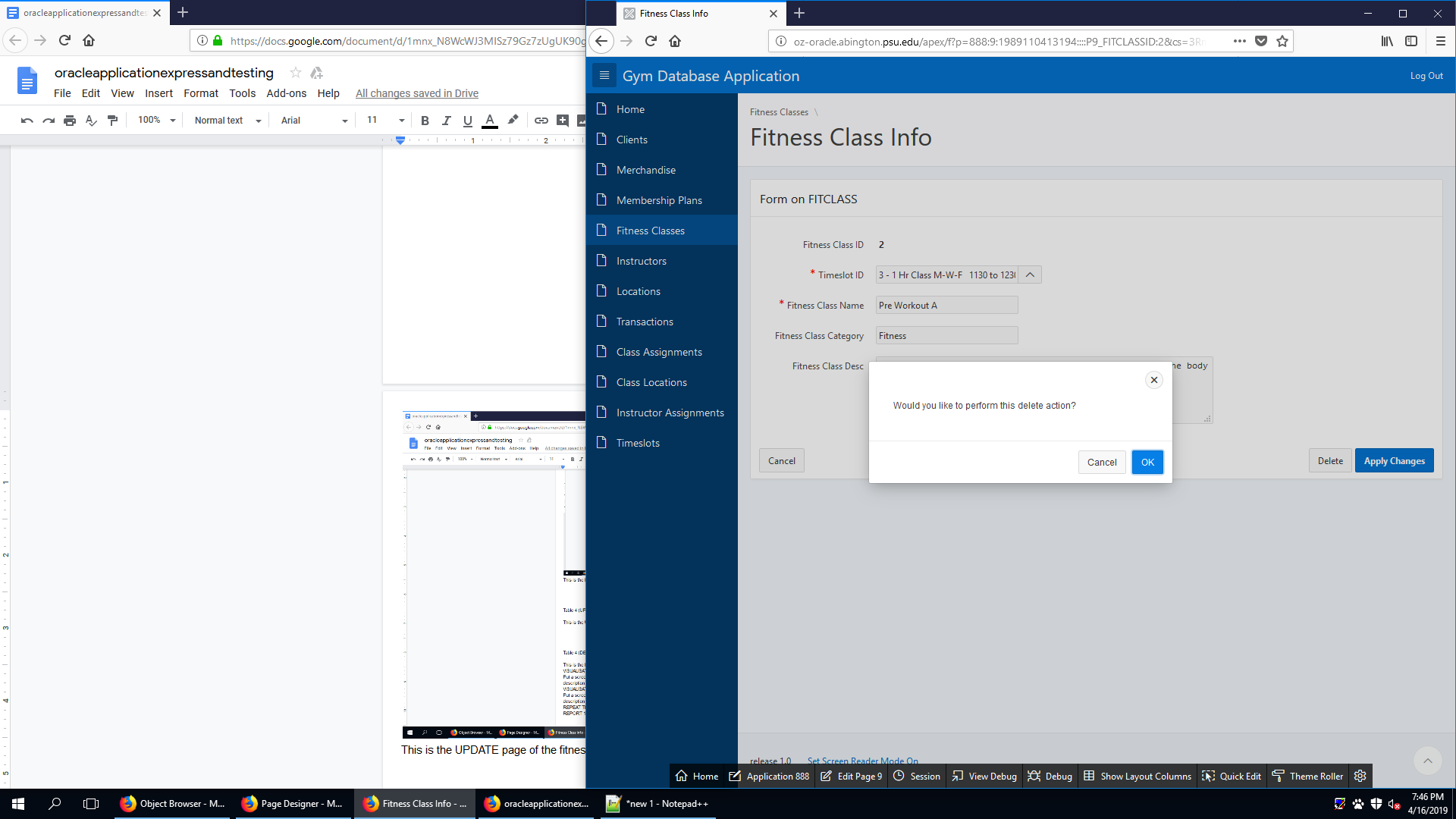
This is the READ page of the fitness class table. It allows the user to see existing fitness class.

FITCLASS (UPDATE)



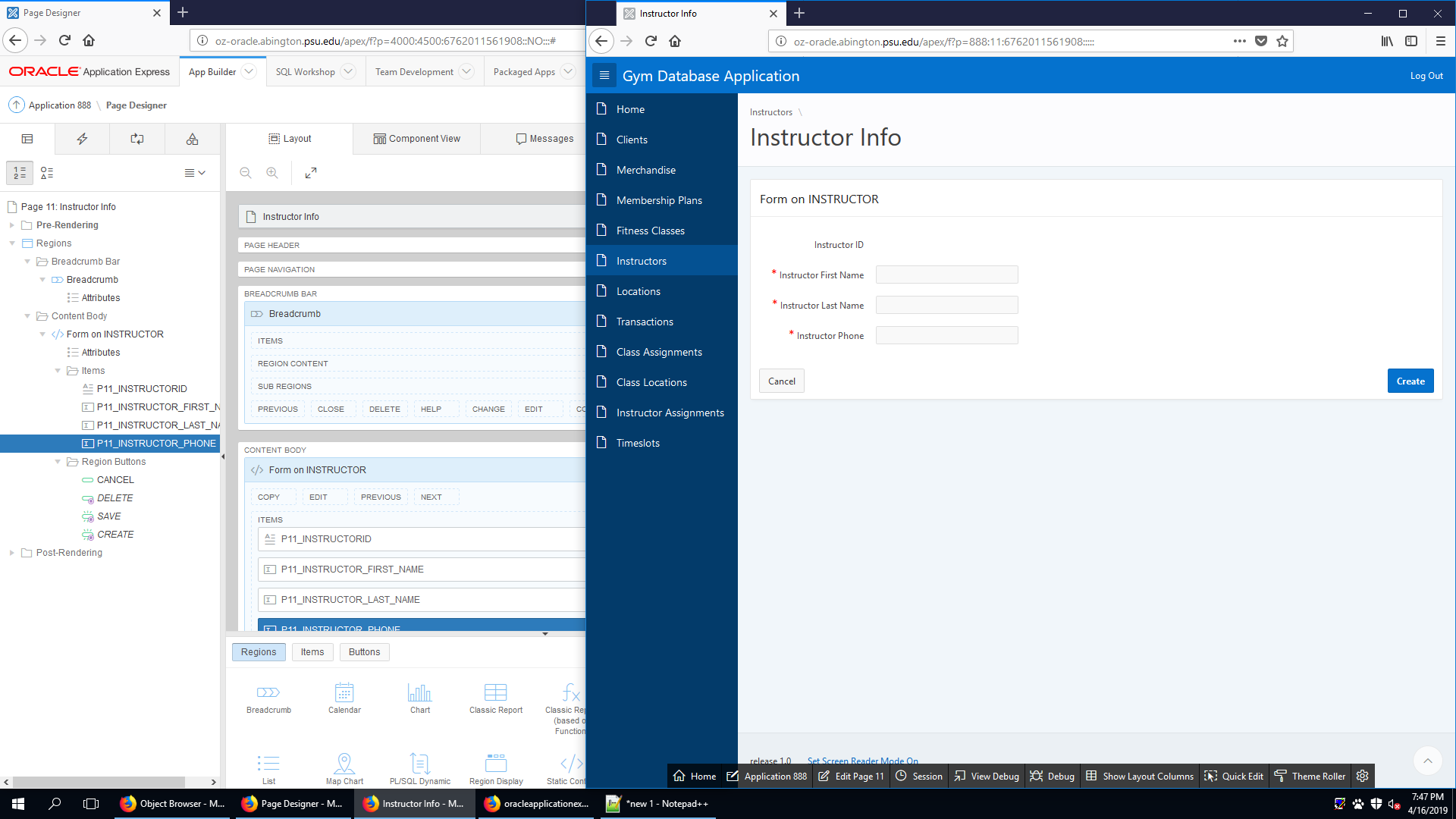
This is the UPDATE page of the fitness class table. It allows the user to update an existing fitness class.

FITCLASS (DELETE)



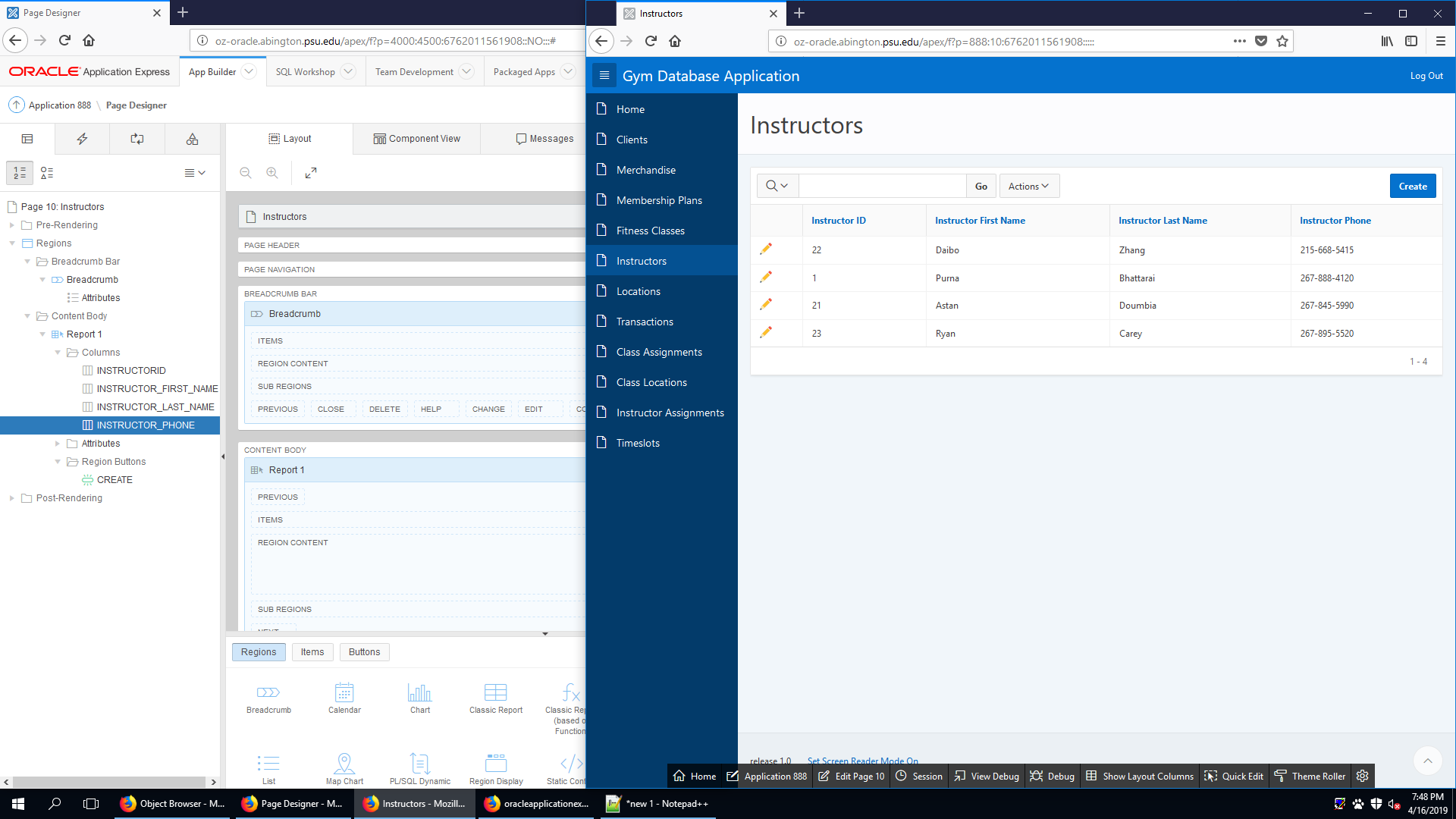
This is the DELETE page of the fitness class table. It allows the user to delete an existing fitness class.

INSTRUCTOR (CREATE)



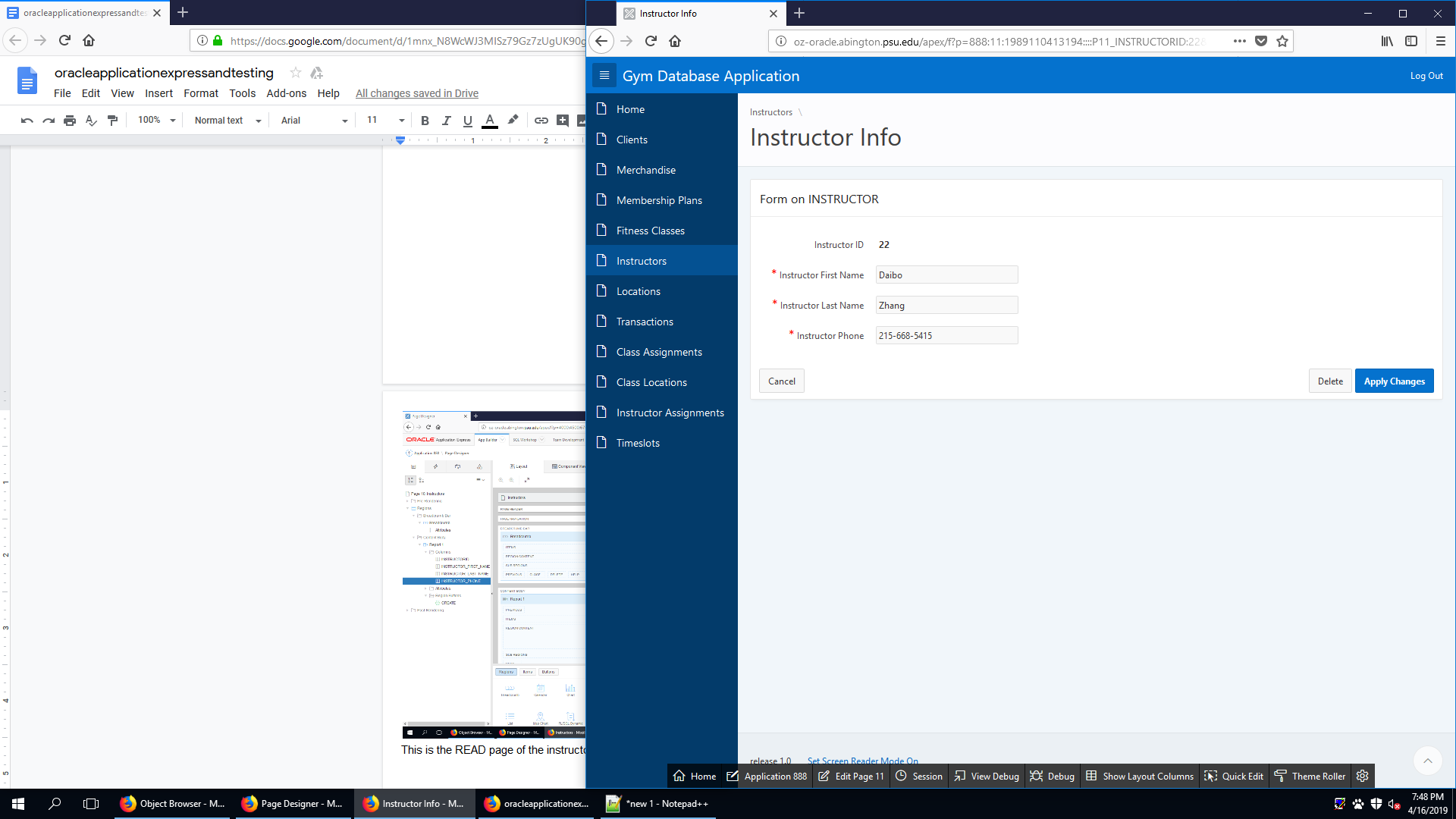
This is the CREATE page of the instructor table. It allows the user to create new instructor.

INSTRUCTOR (READ)



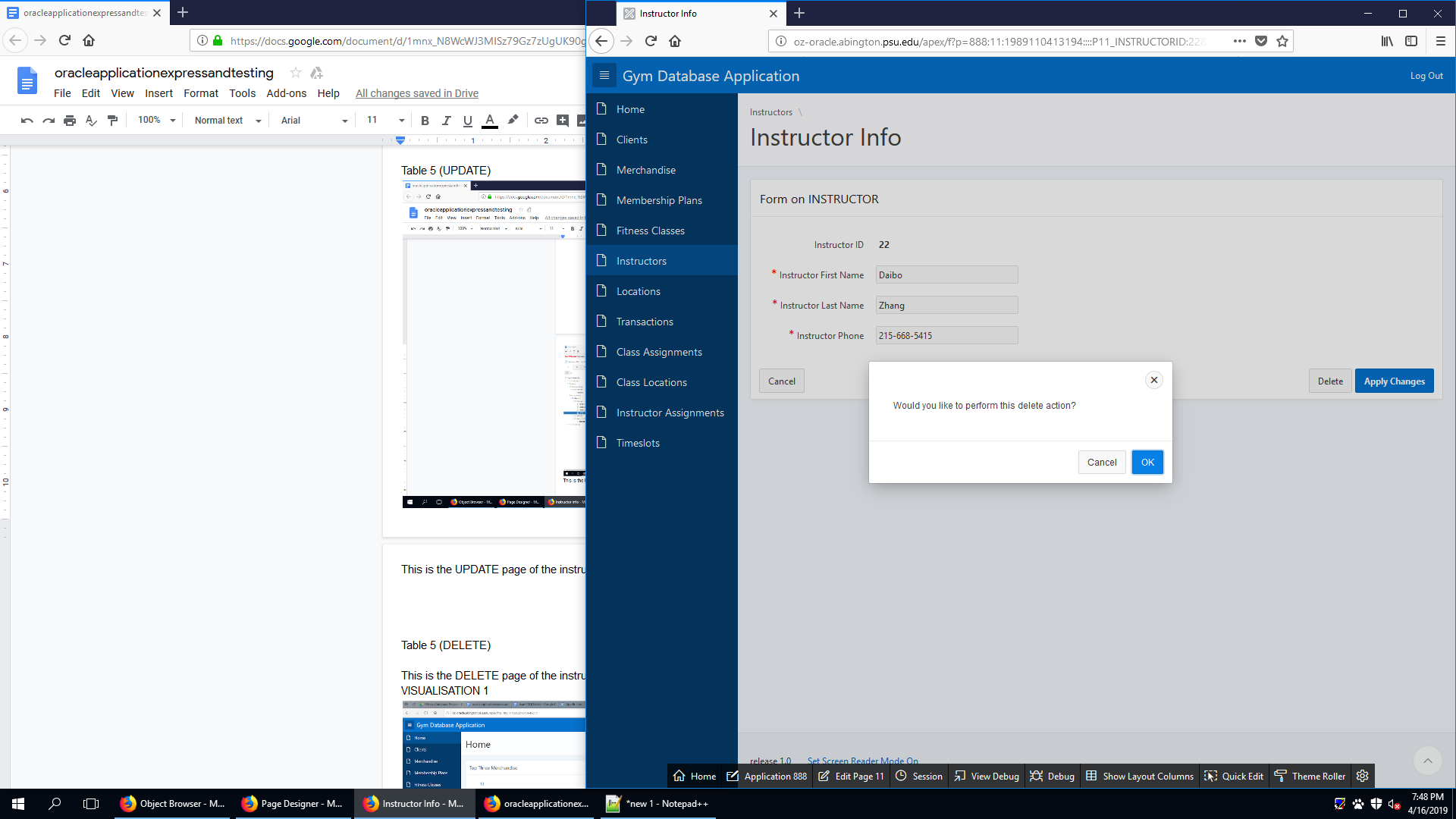
This is the READ page of the instructor table. It allows the user to see existing instructor.

INSTRUCTOR (UPDATE)



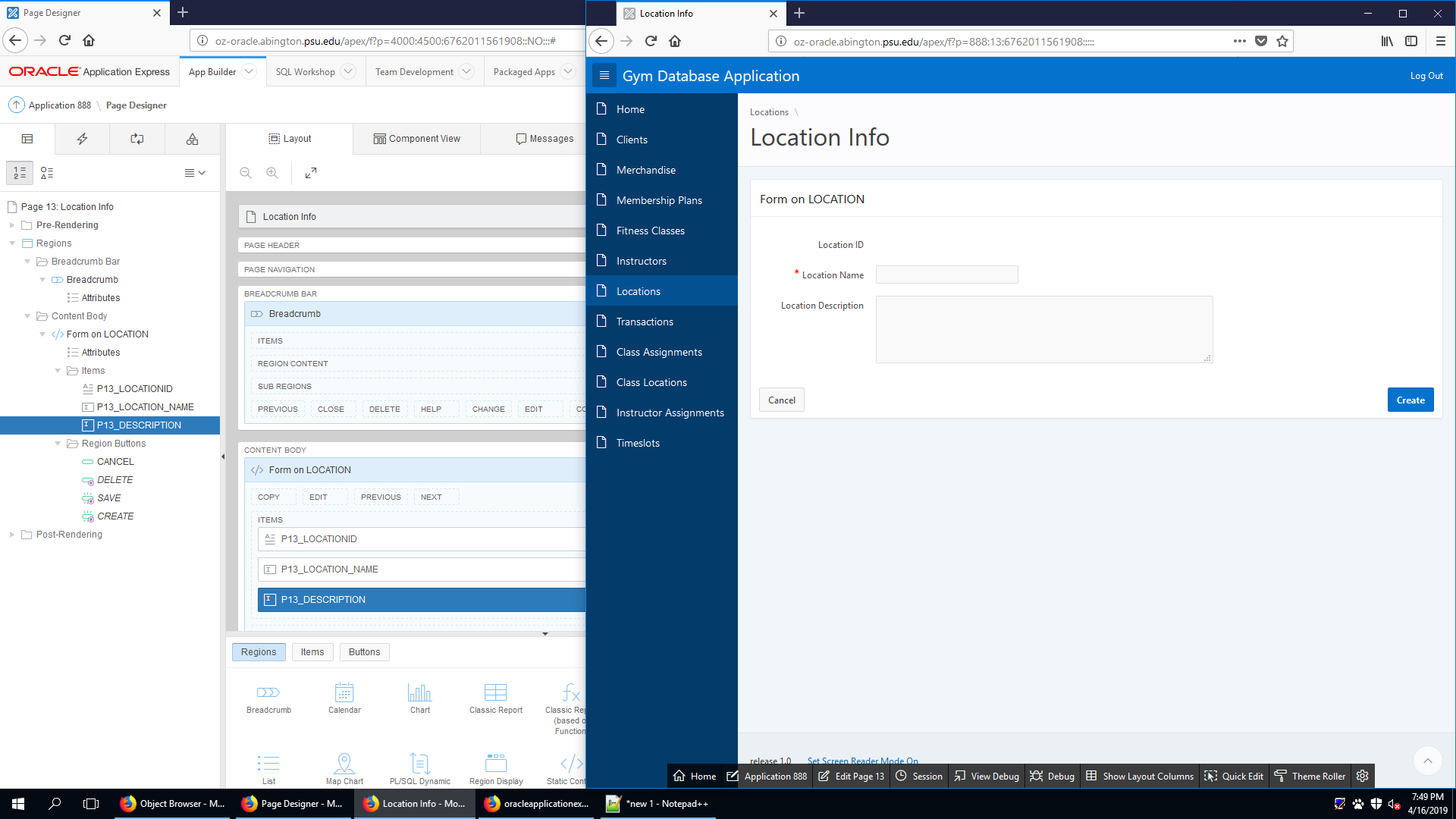
This is the UPDATE page of the instructor table. It allows the user to update an existing instructor.

INSTRUCTOR (DELETE)



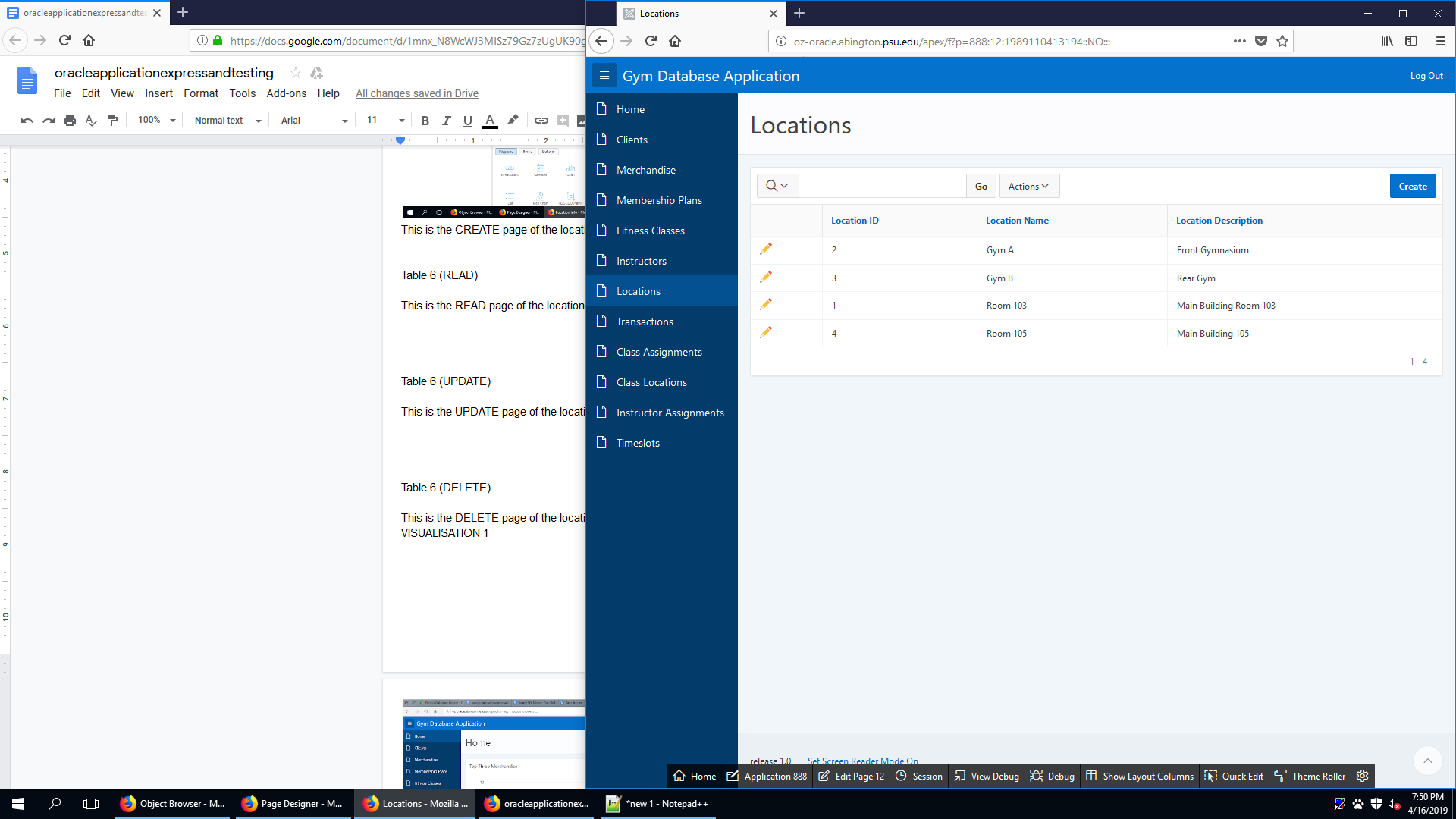
This is the DELETE page of the instructor table. It allows the user to delete an existing instructor.

LOCATION (CREATE)



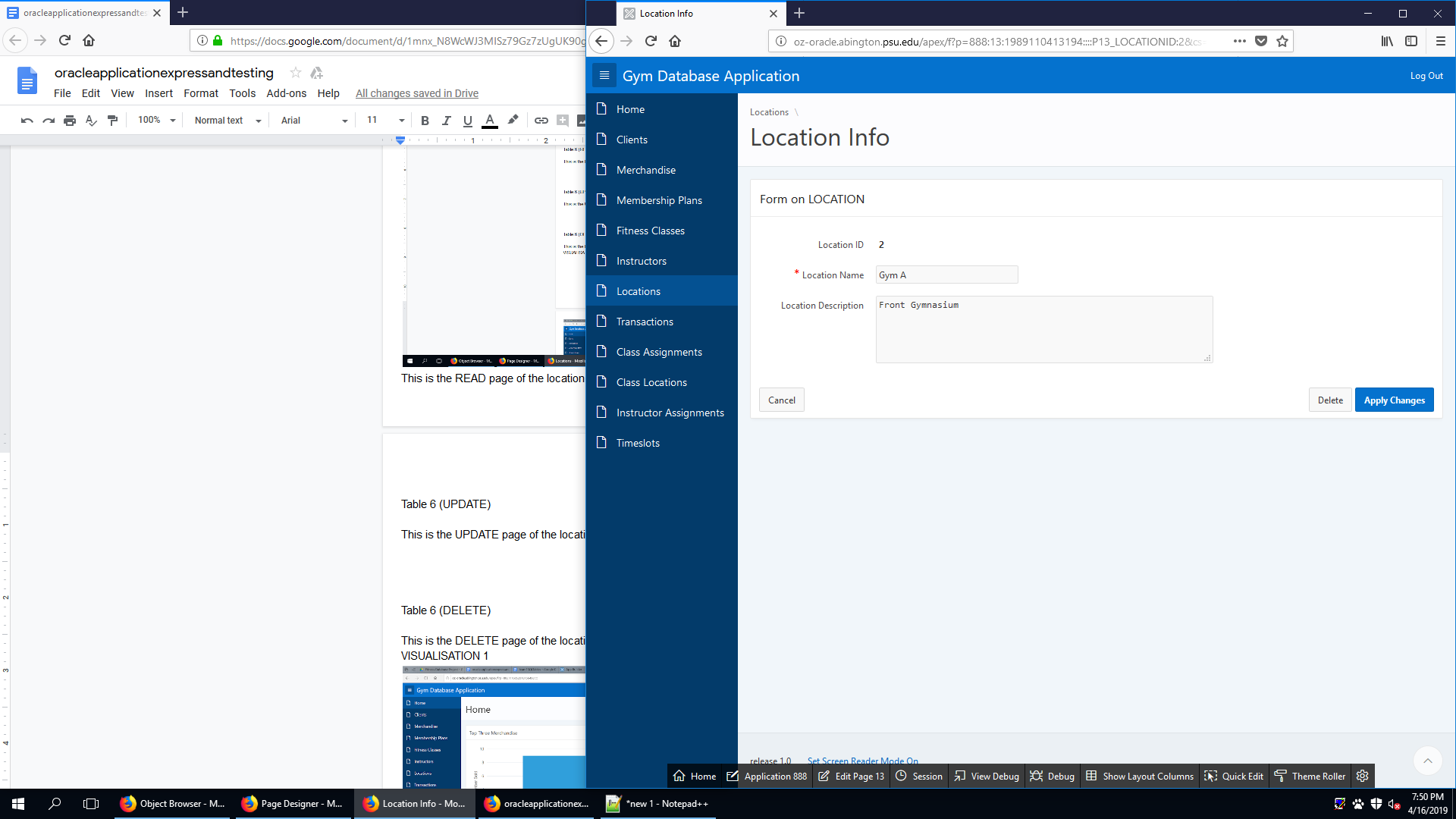
This is the CREATE page of the location table. It allows the user to create new location.

LOCATION (READ)



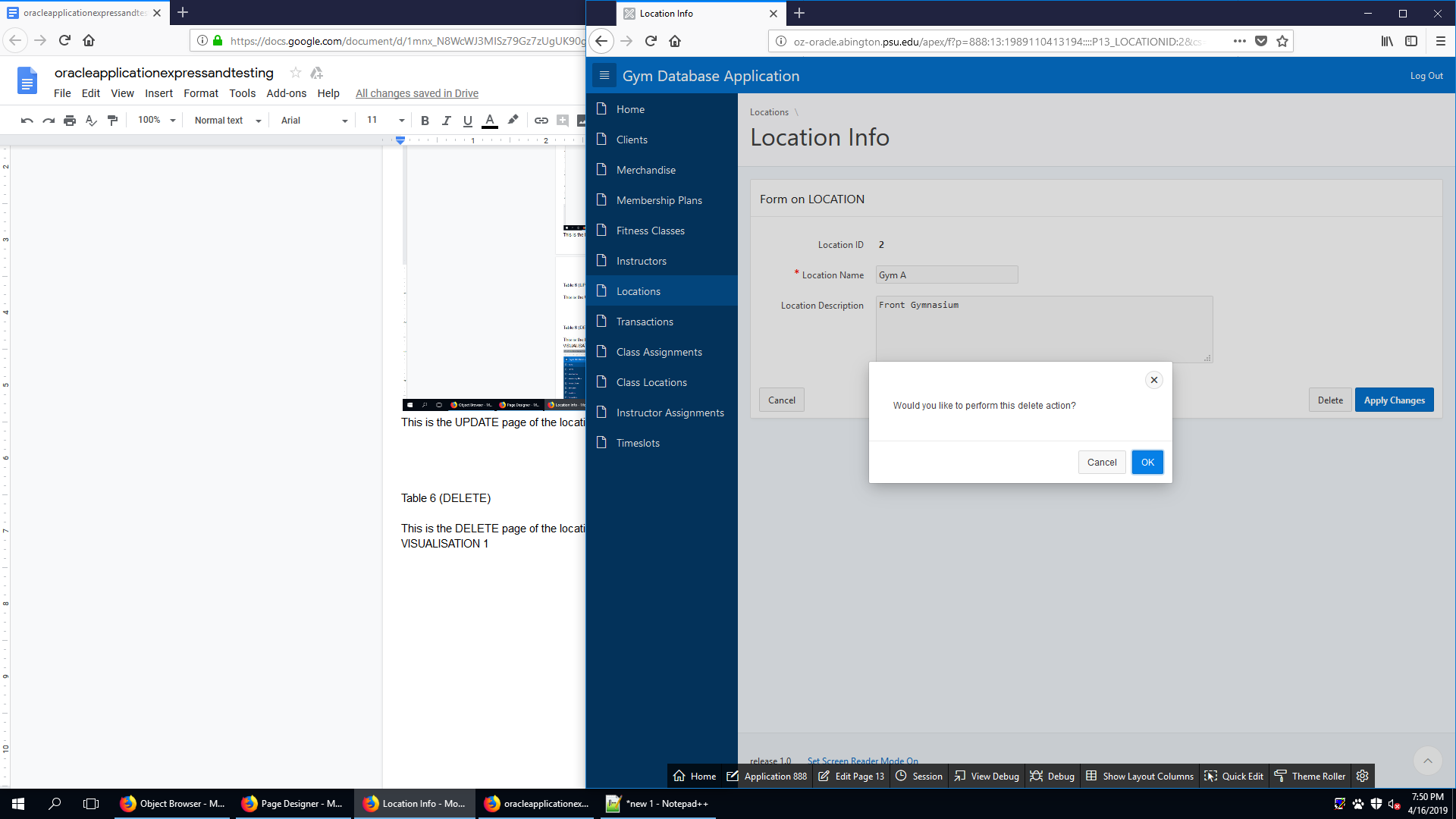
This is the READ page of the location table. It allows the user to see existing location.

LOCATION (UPDATE)



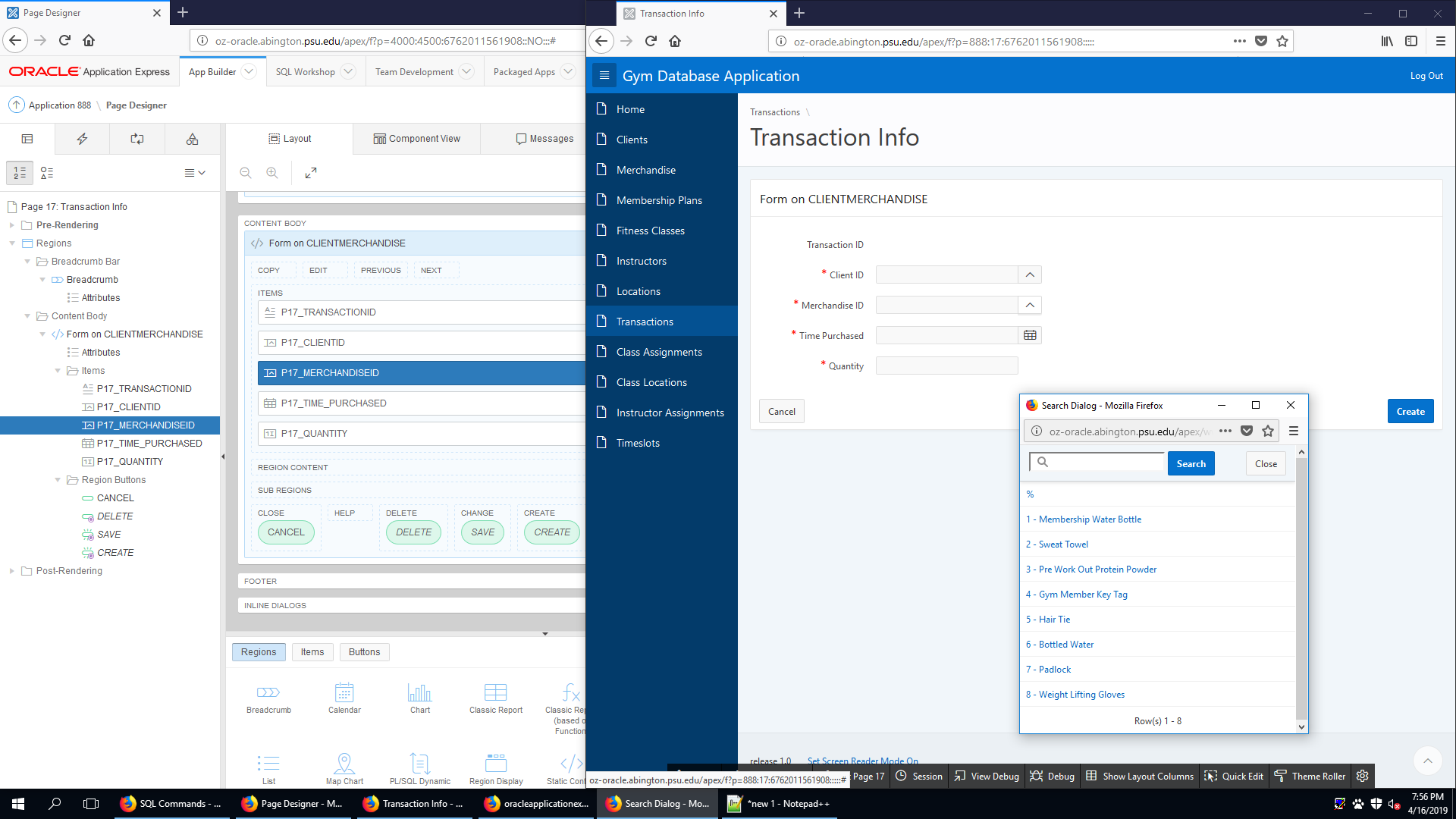
This is the UPDATE page of the location table. It allows the user to update an existing location.

LOCATION (DELETE)



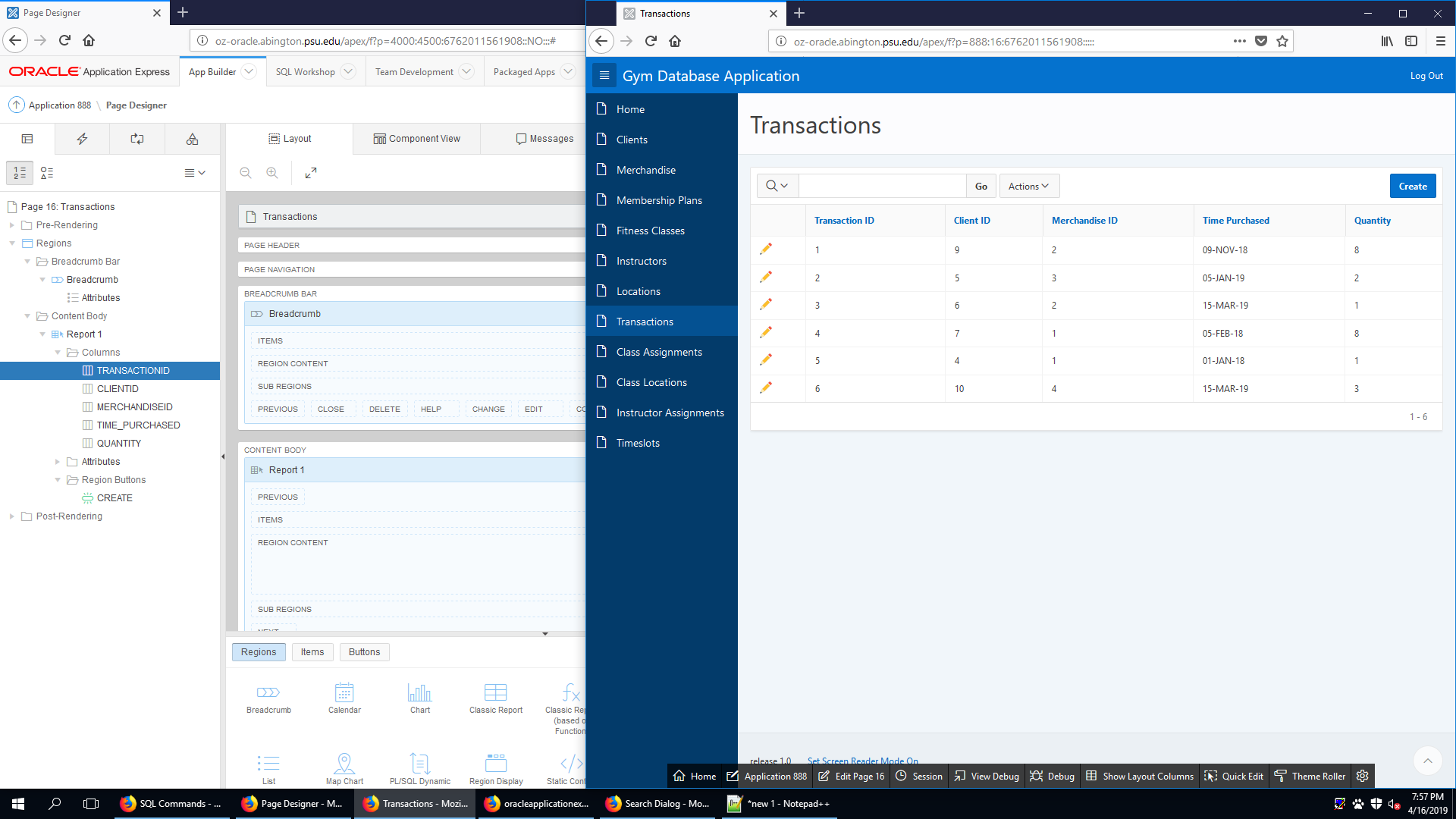
This is the DELETE page of the location table. It allows the user to delete an existing location.

CLIENTMERCHANDISE (CREATE)



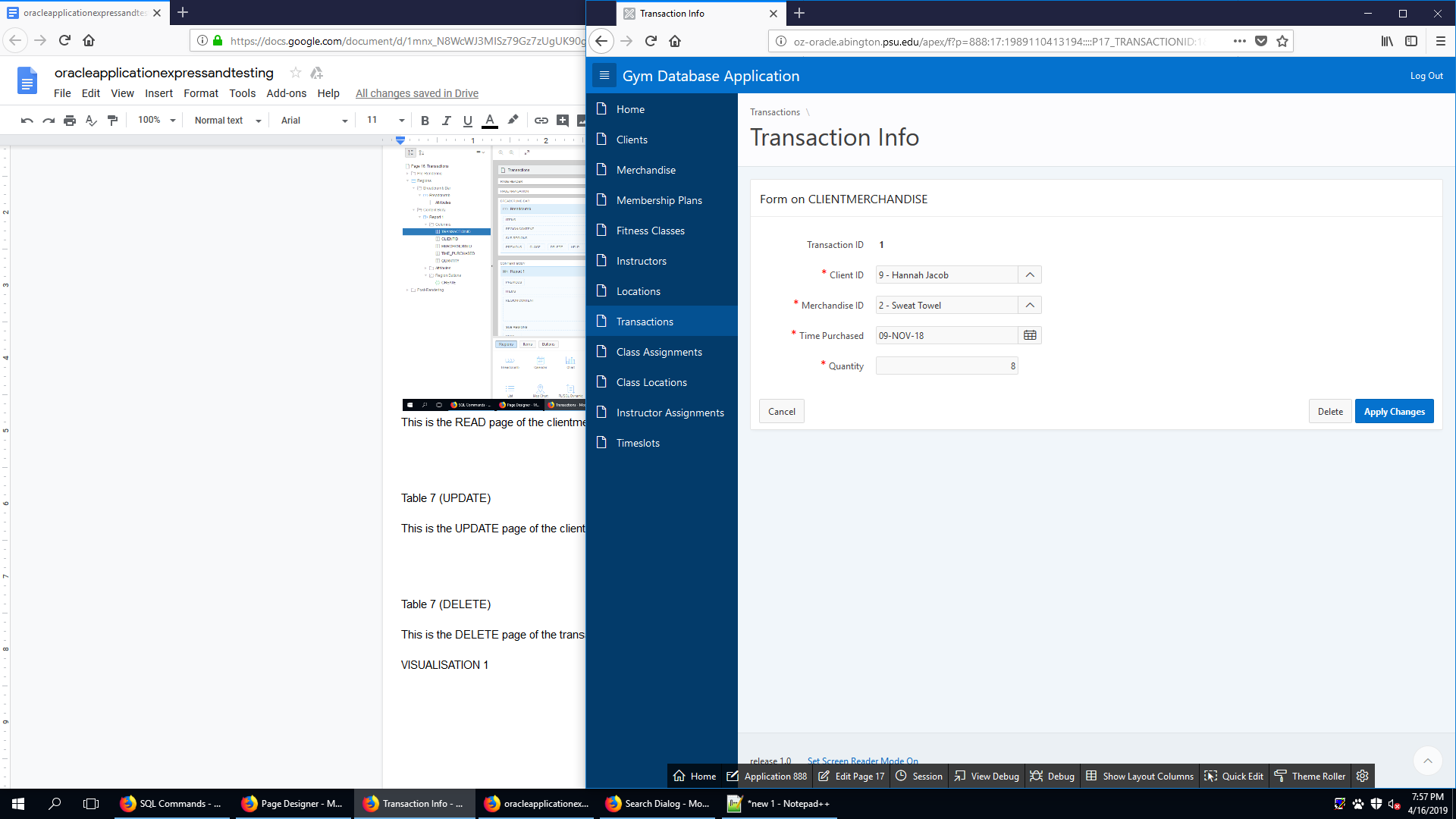
This is the CREATE page of the clientmerchandise table. It allows the user to create new transaction.

CLIENTMERCHANDISE (READ)



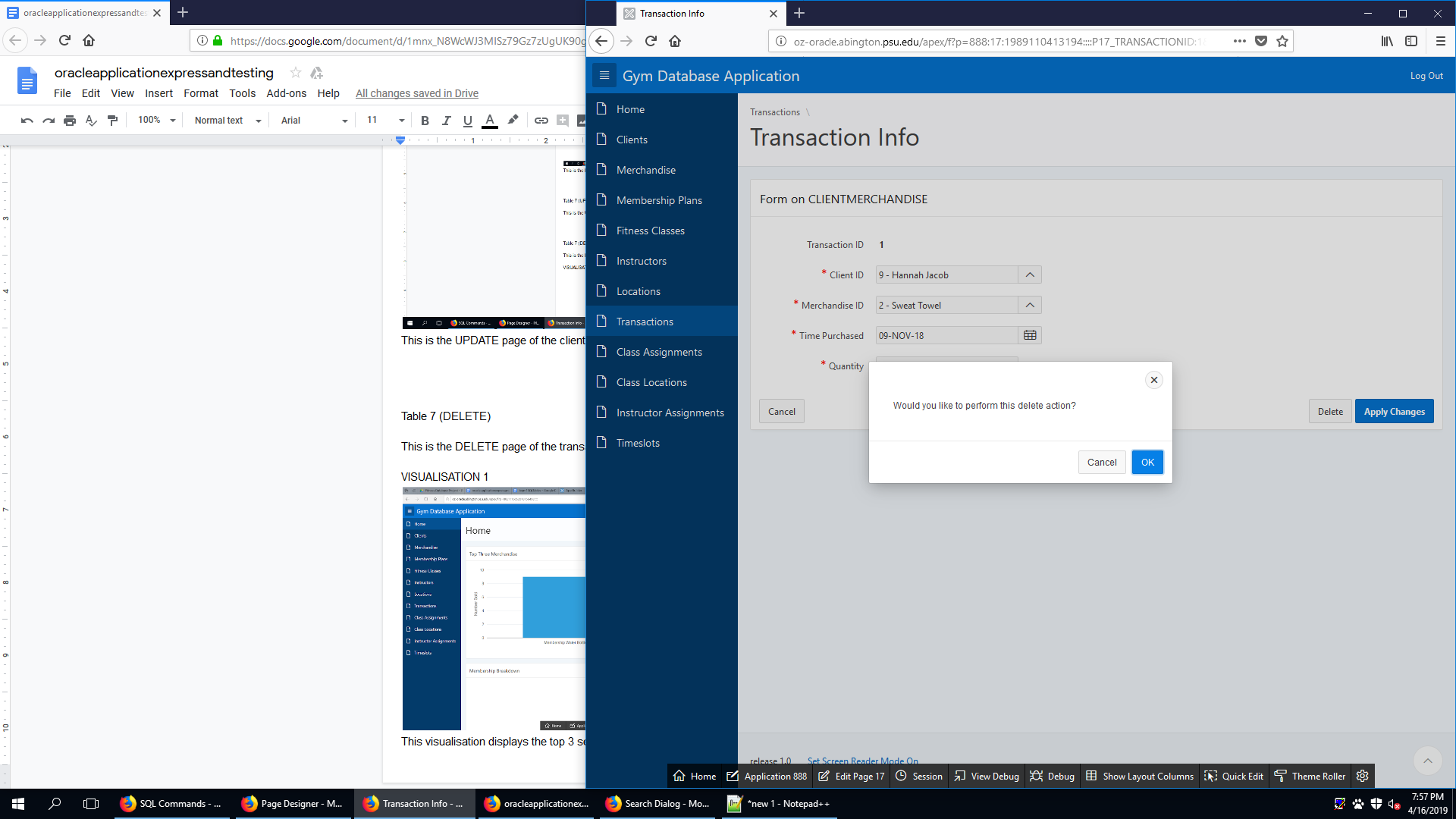
This is the READ page of the clientmerchandise table. It allows the user to see existing transaction.

CLIENTMERCHANDISE (UPDATE)



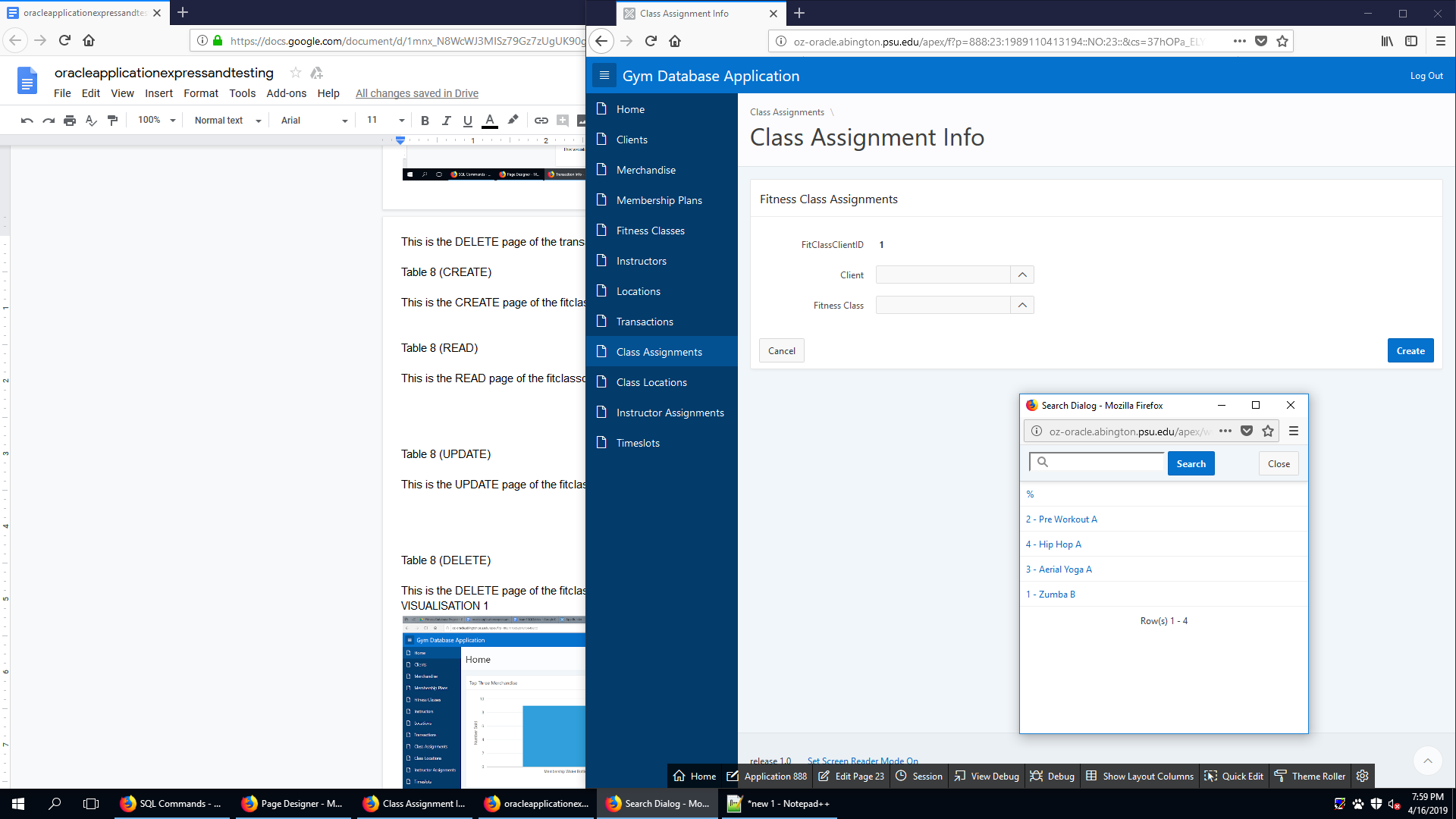
This is the UPDATE page of the clientmerchandise table. It allows the user to update an existing transaction.

CLIENTMERCHANDISE (DELETE)



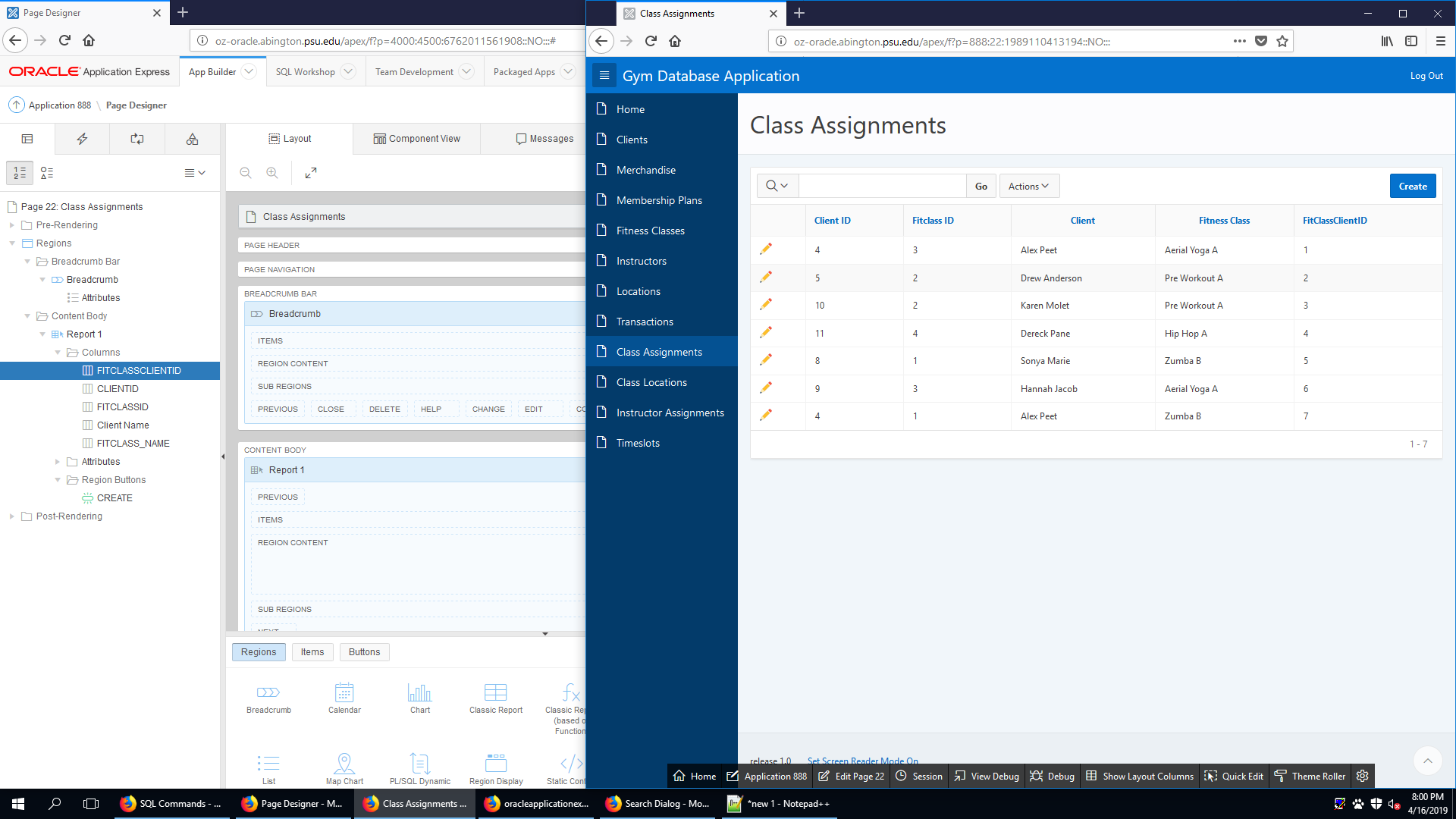
This is the DELETE page of the transaction table. It allows the user to delete an existing transaction.

FITCLASSCLIENT (CREATE)



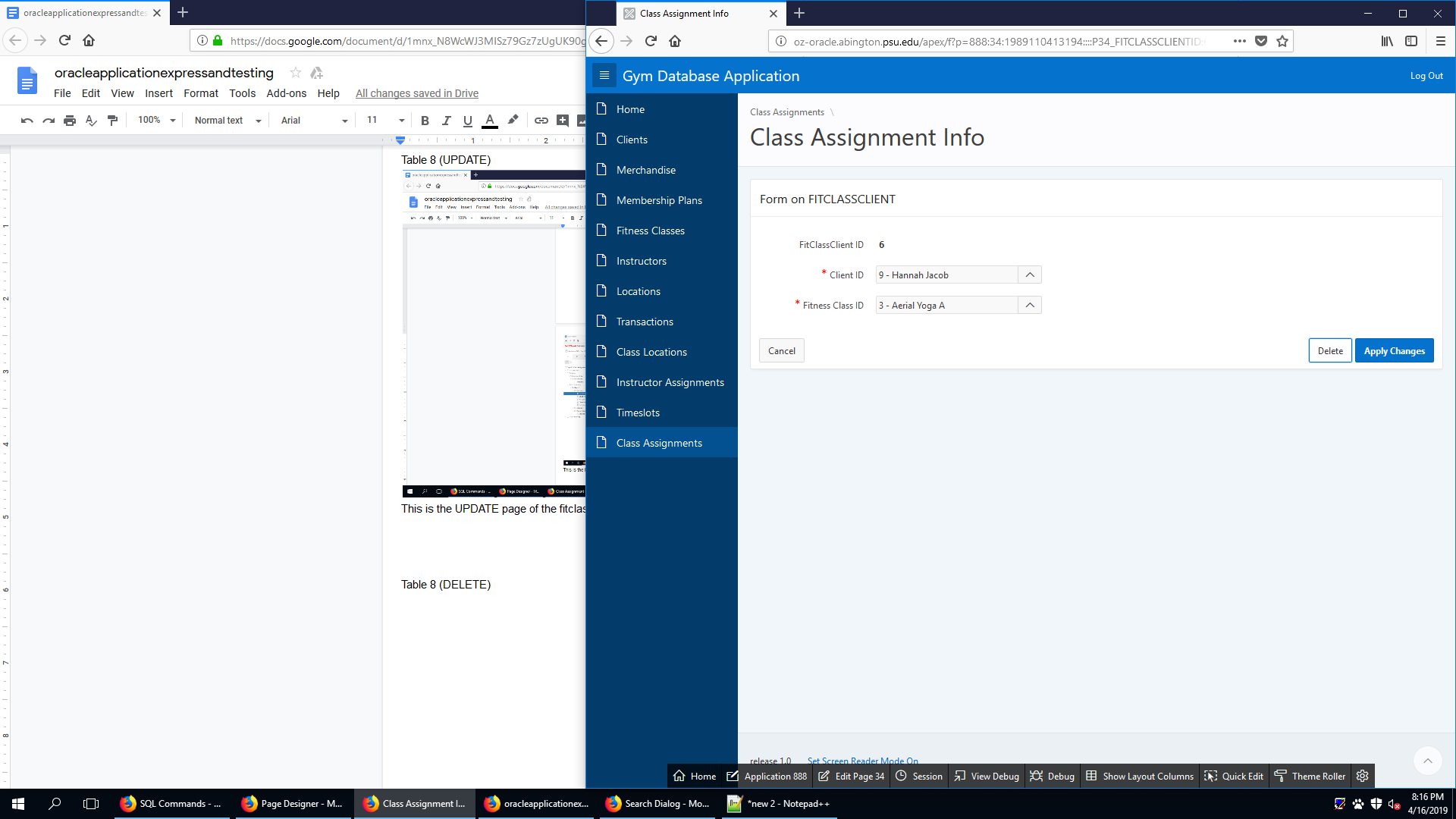
This is the CREATE page of the fitclassclient table. It allows the user to create a new class assignment.

FITCLASSCLIENT (READ)



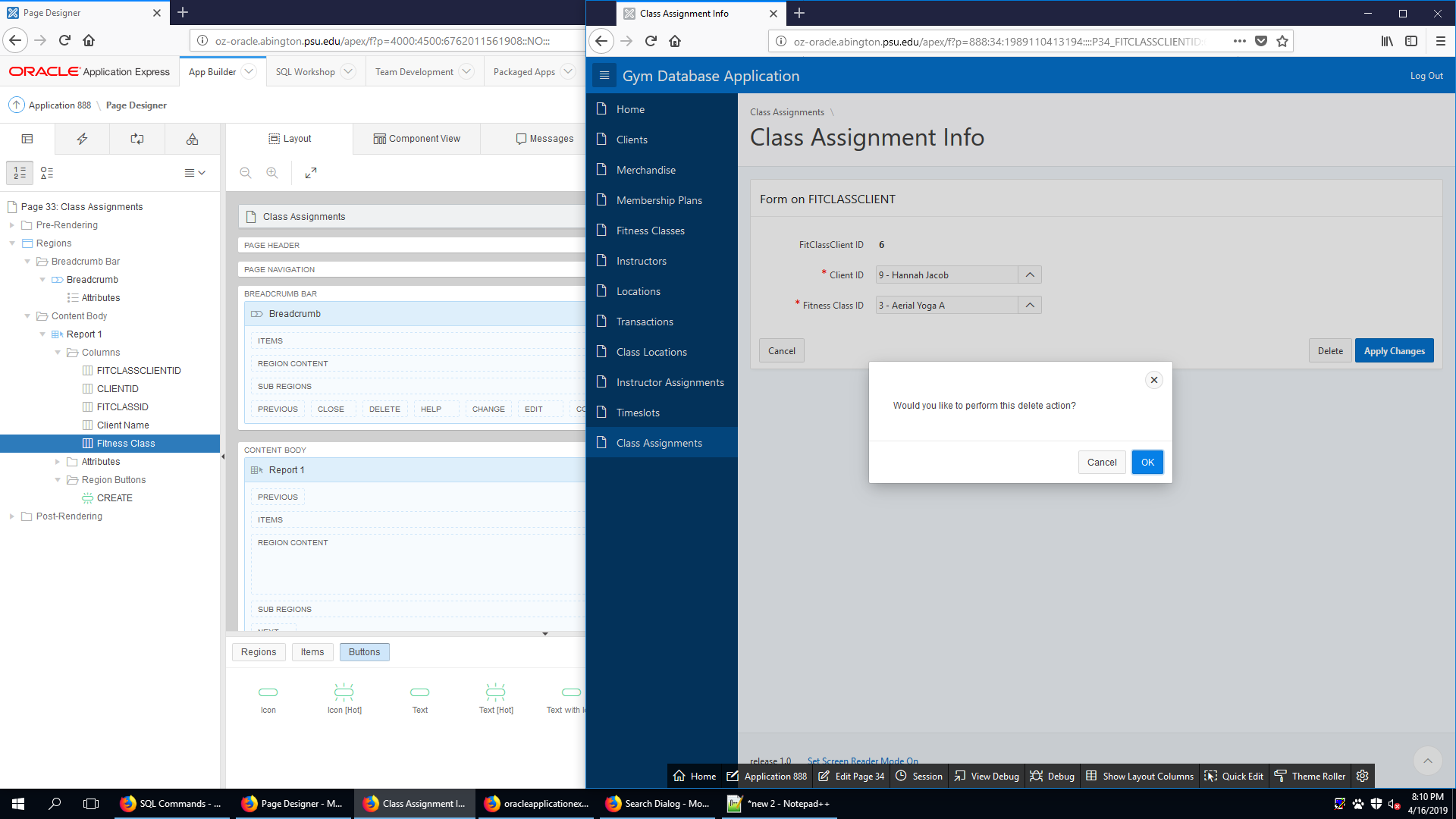
This is the READ page of the fitclassclient table. It allows the user to see an existing class assignment.

FITCLASSCLIENT (UPDATE)



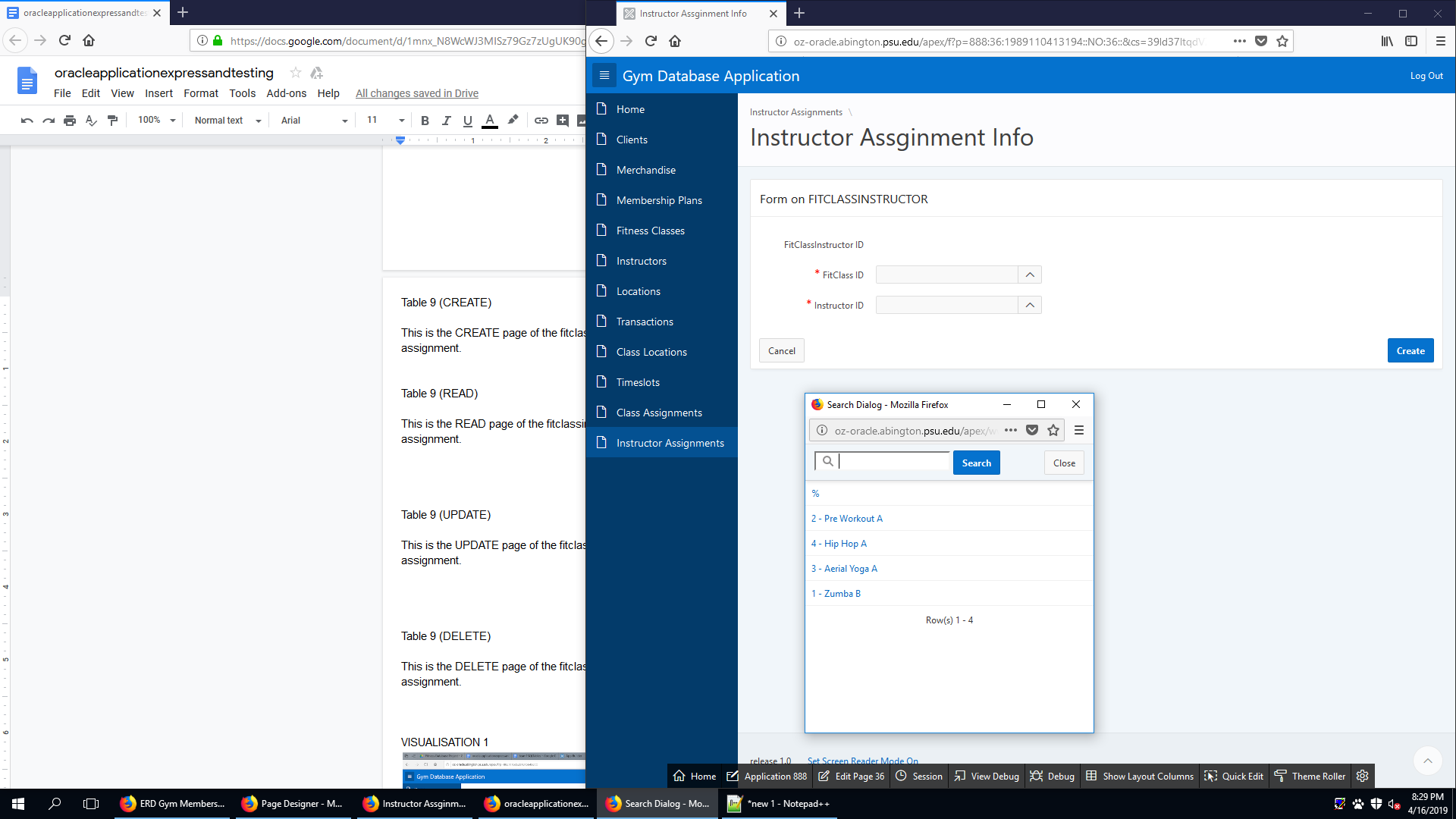
This is the UPDATE page of the fitclassclient table. It allows the user to update an existing class assignment.

FITCLASSCLIENT (DELETE)



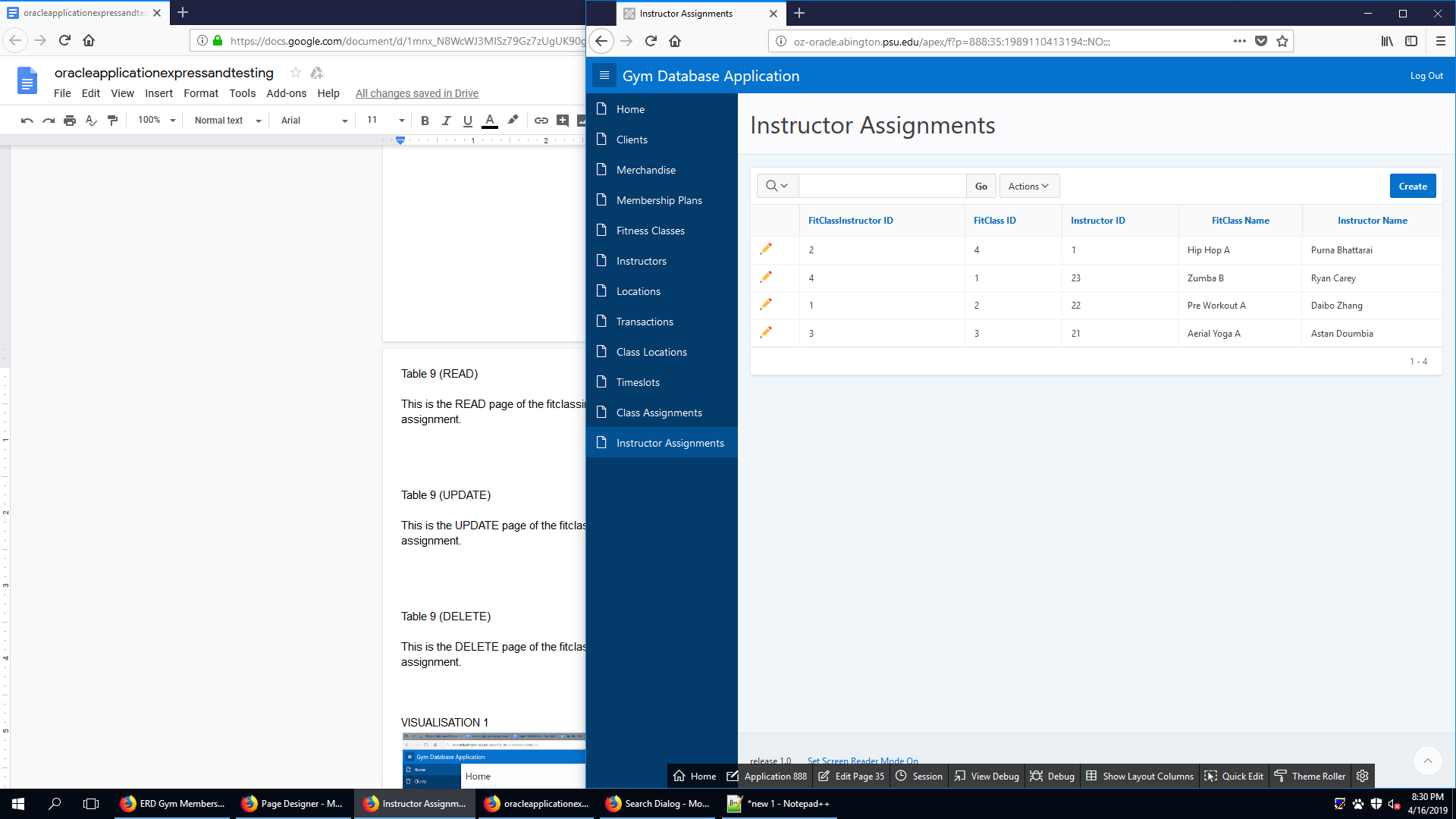
This is the DELETE page of the fitclassclient table. It allows the user to delete an existing class assignment.

FITCLASSINSTRUCTOR (CREATE)



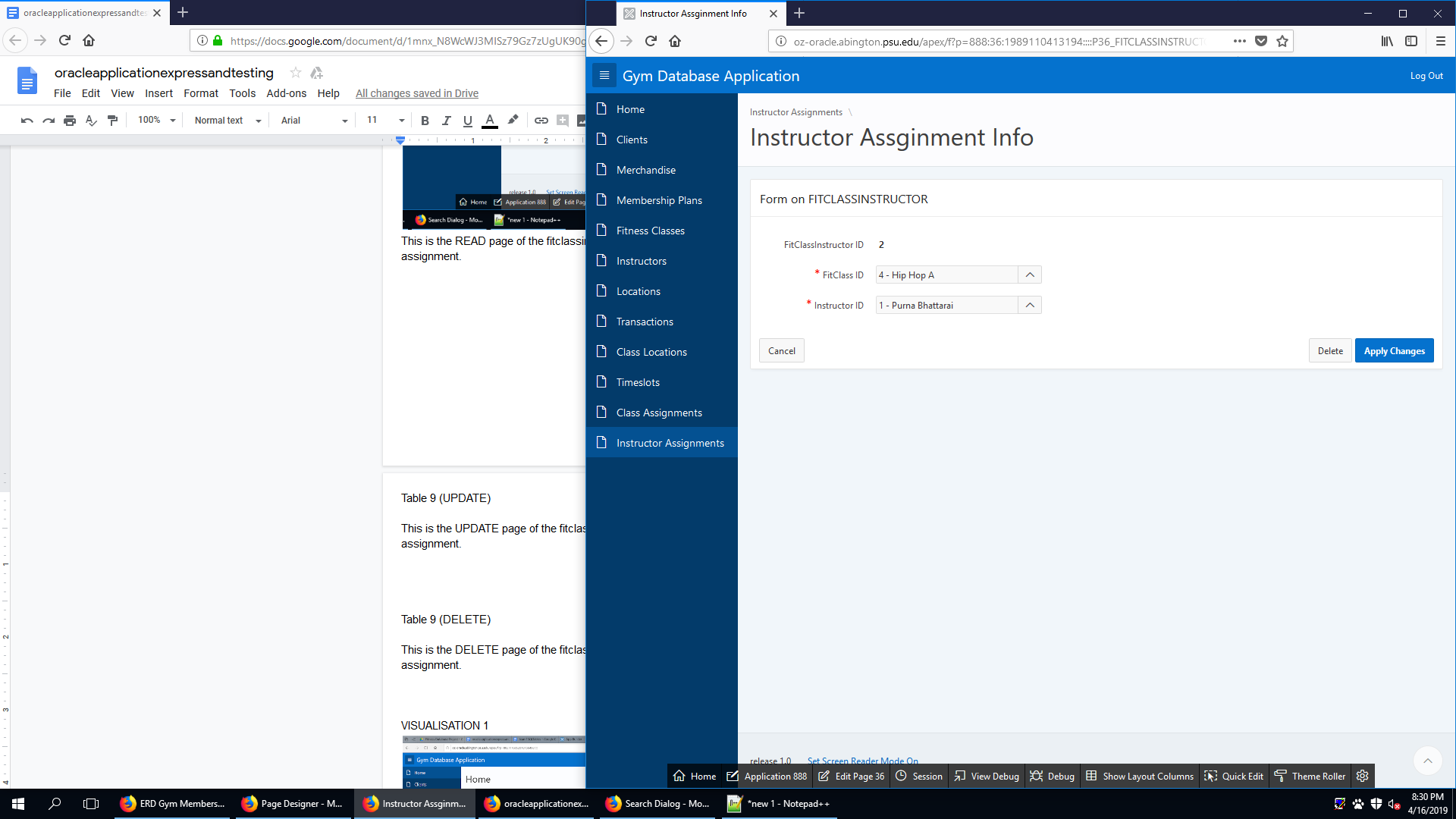
This is the CREATE page of the fitclassinstructor assignment table. It allows the user to create new instructor assignment.

FITCLASSINSTRUCTOR (READ)



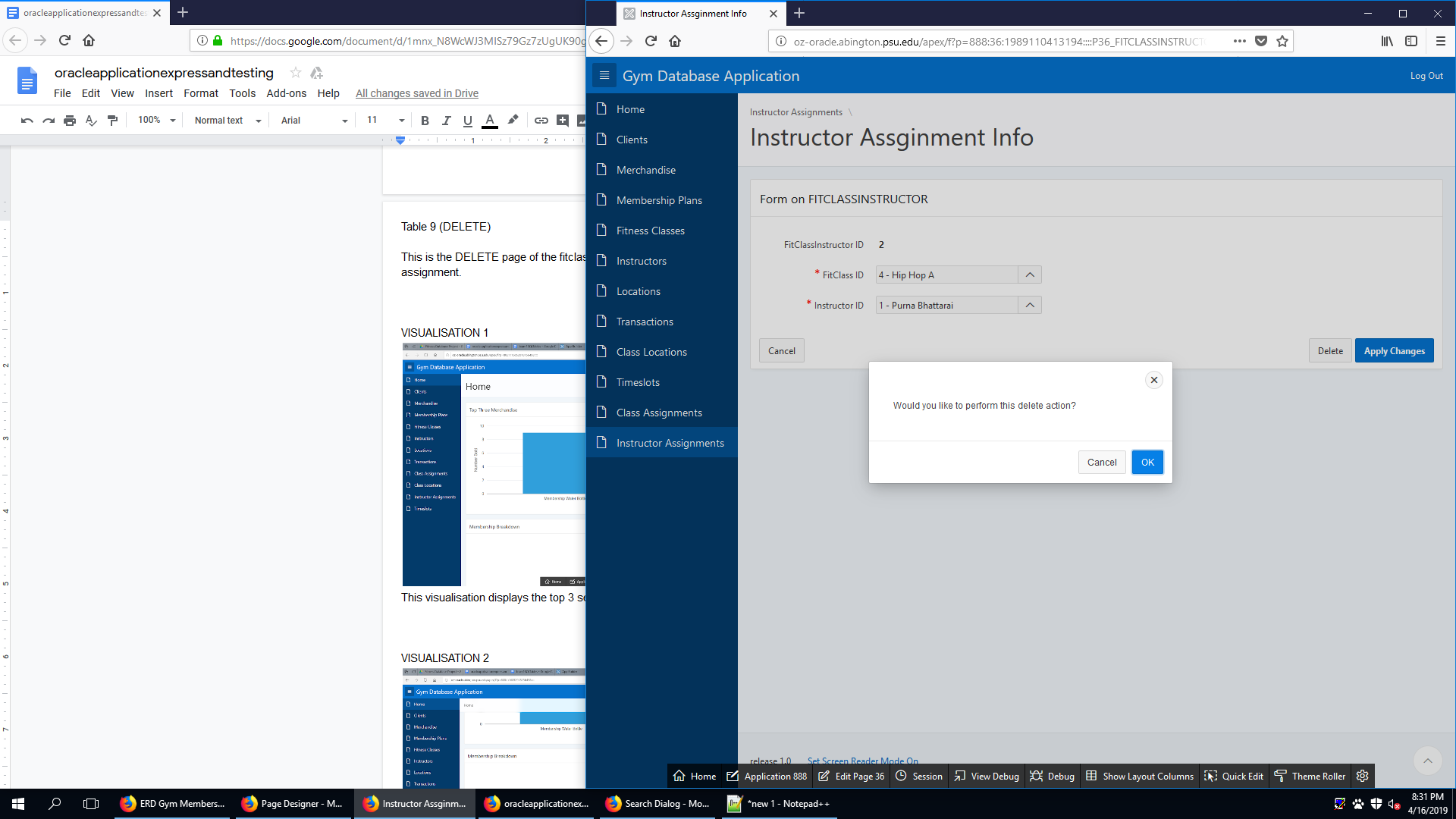
This is the READ page of the fitclassinstructor assignment table. It allows the user to see existing instructor assignment.

FITCLASSINSTRUCTOR (UPDATE)



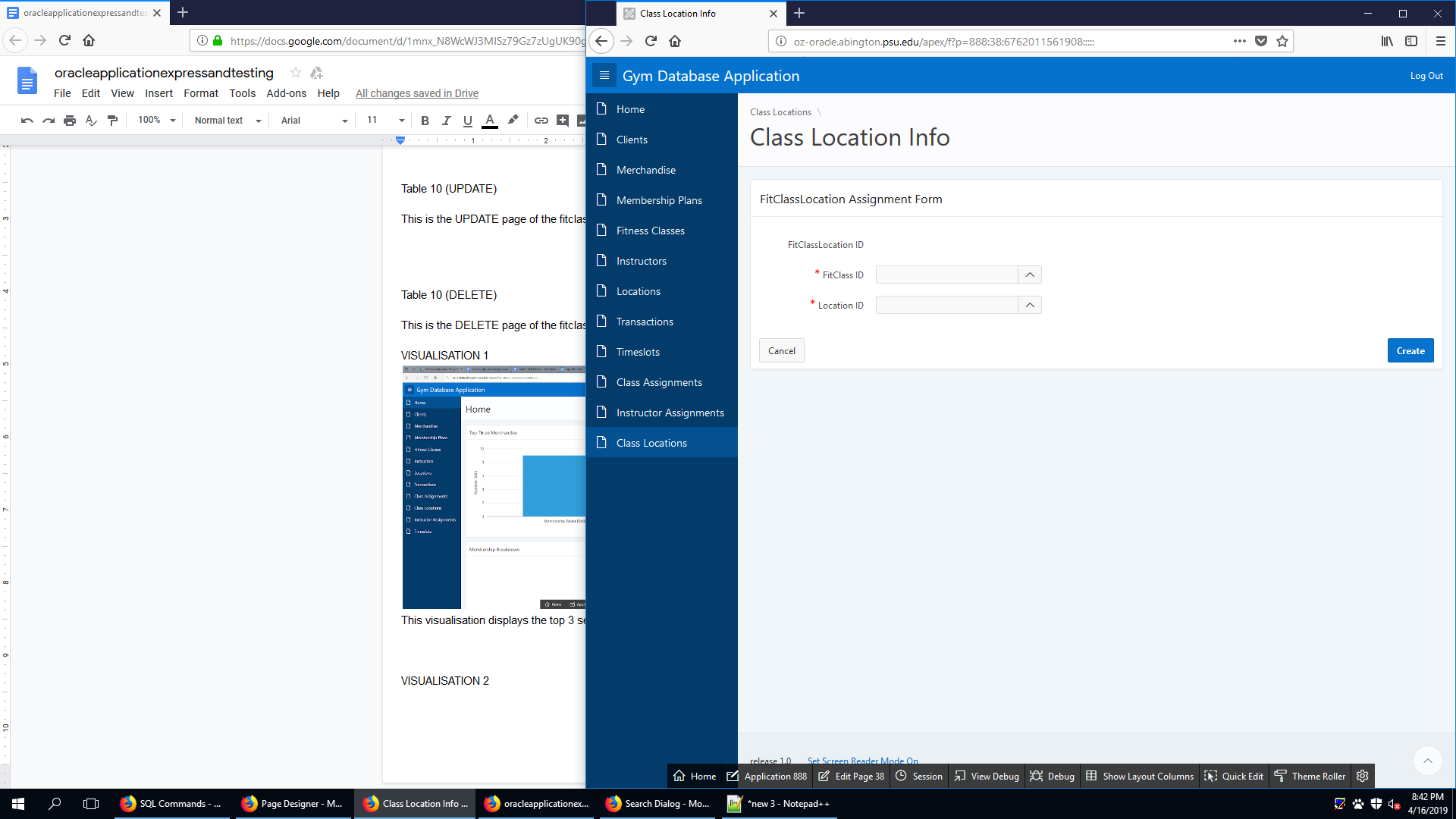
This is the UPDATE page of the fitclassinstructor assignment table. It allows the user to update an existing instructor assignment.

FITCLASSINSTRUCTOR (DELETE)



This is the DELETE page of the fitclassinstructor assignment table. It allows the user to delete an existing instructor assignment.

FITCLASSLOCATION(CREATE)



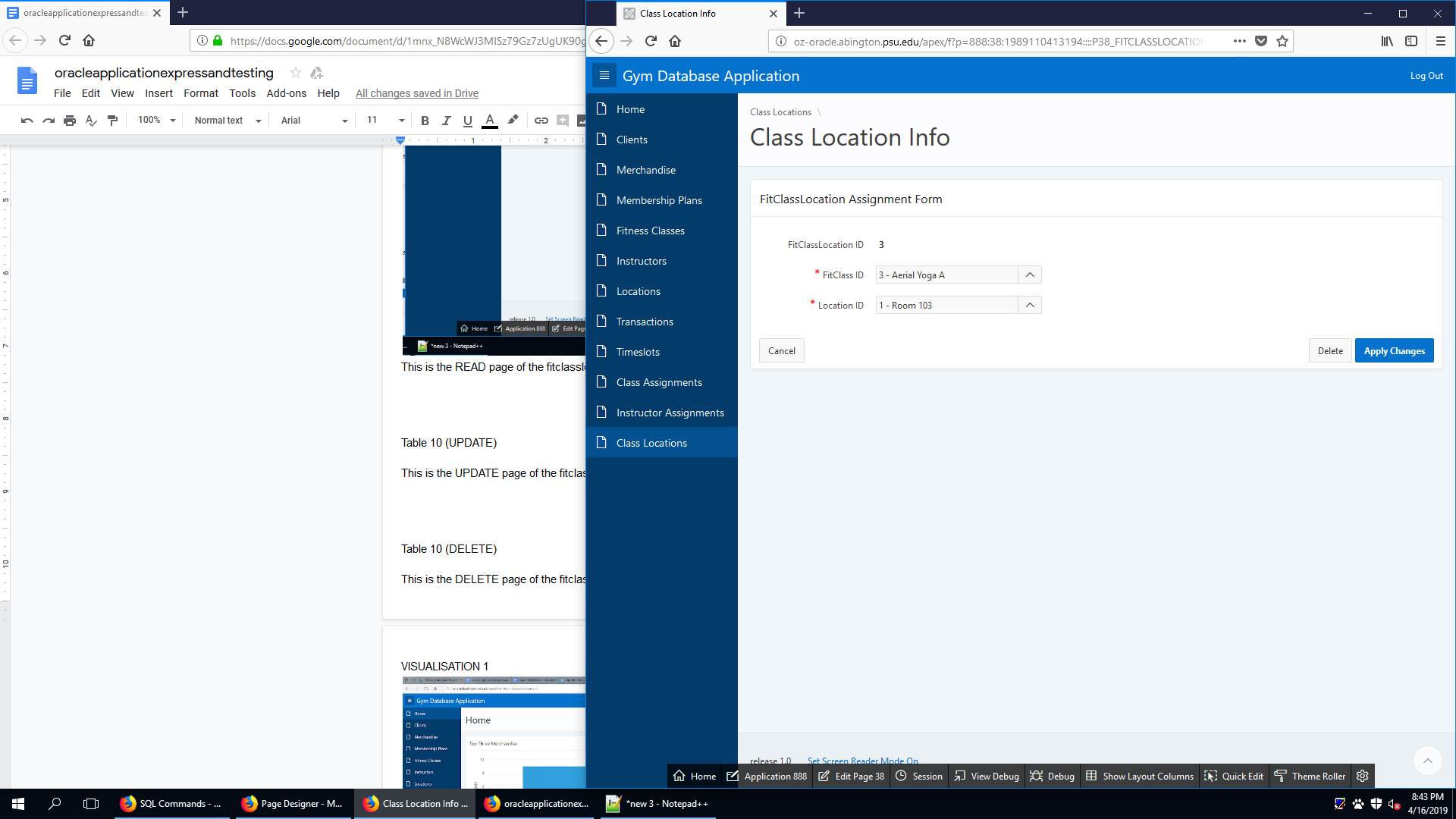
This is the CREATE page of the fitclasslocation table. It allows the user to create new class location.

FITCLASSLOCATION (READ)



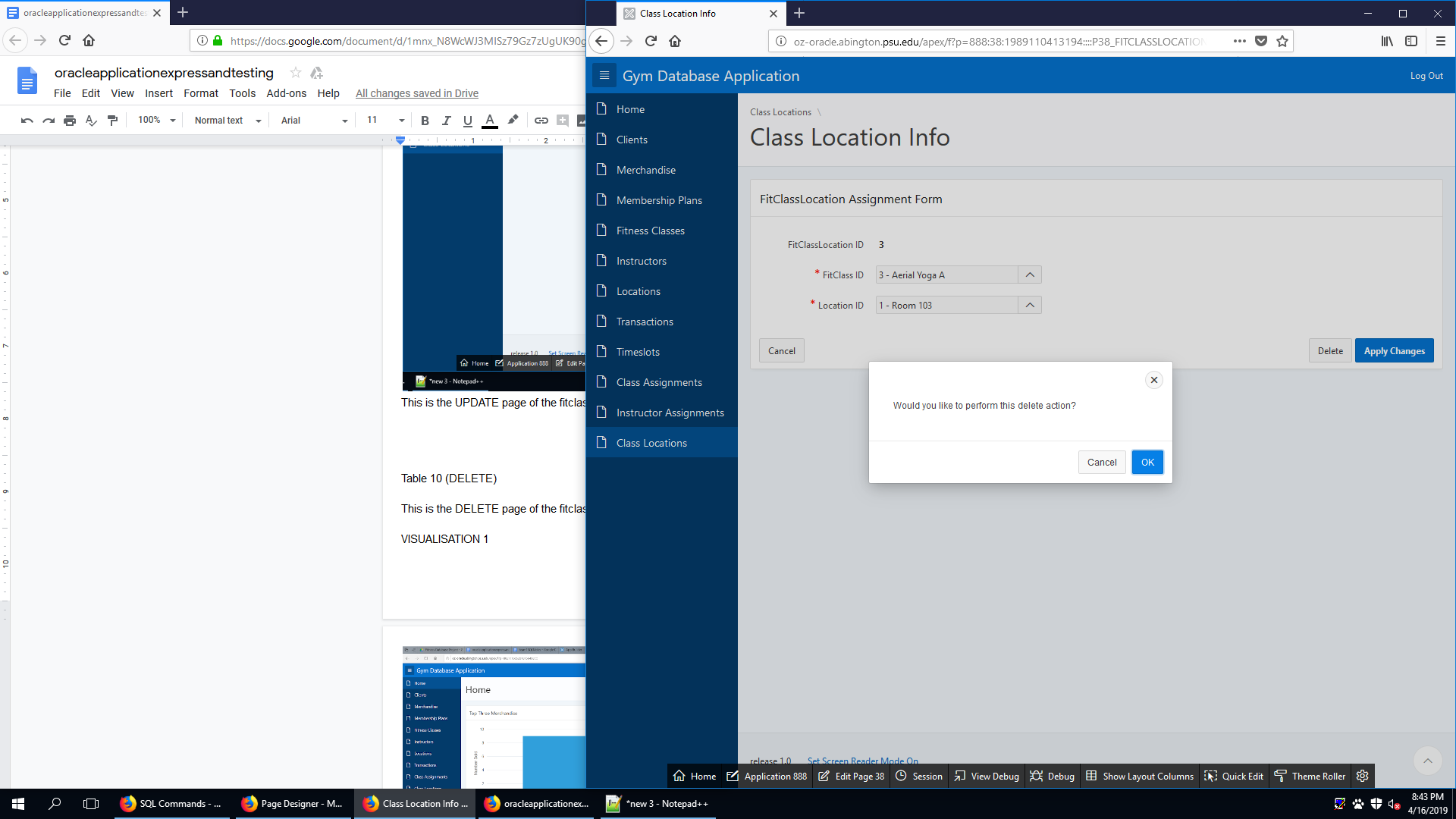
This is the READ page of the fitclasslocation table. It allows the user to see existing class locations.

FITCLASSLOCATION (UPDATE)



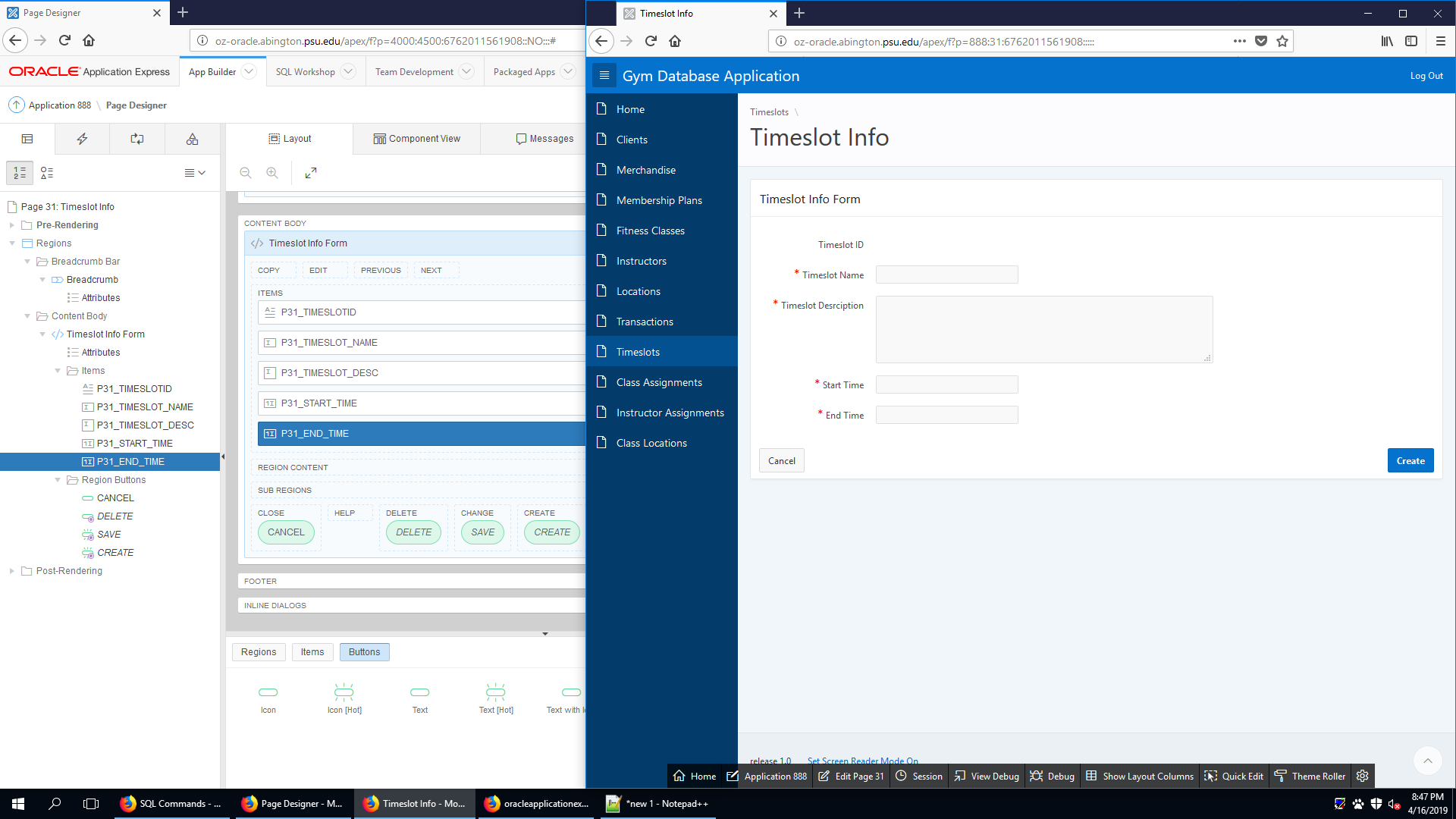
This is the UPDATE page of the fitclasslocation table. It allows the user to update an existing class location.

FITCLASSLOCATION (DELETE)



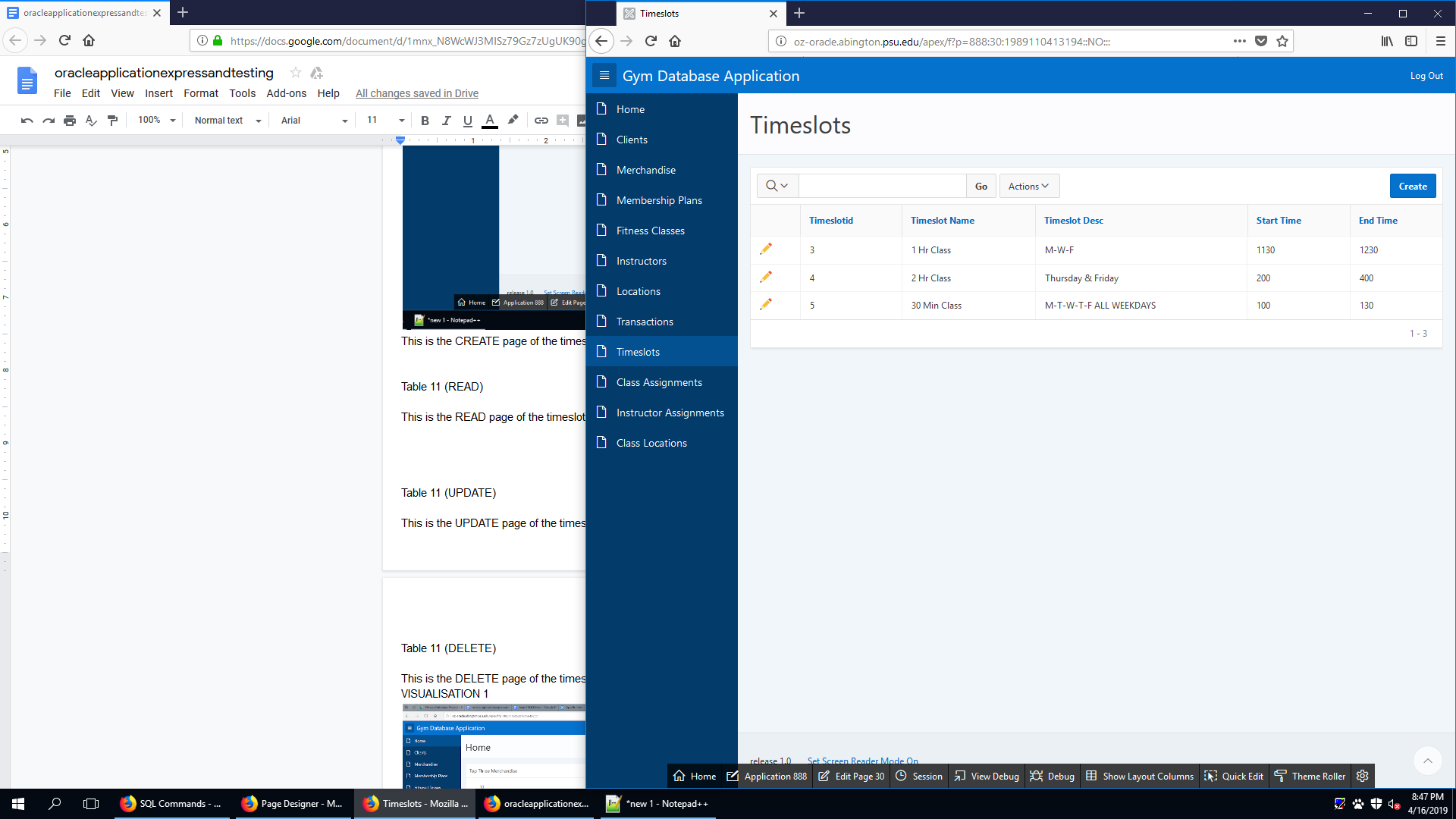
This is the DELETE page of the fitclasslocation table. It allows the user to delete an existing class location.

TIMESLOT (CREATE)



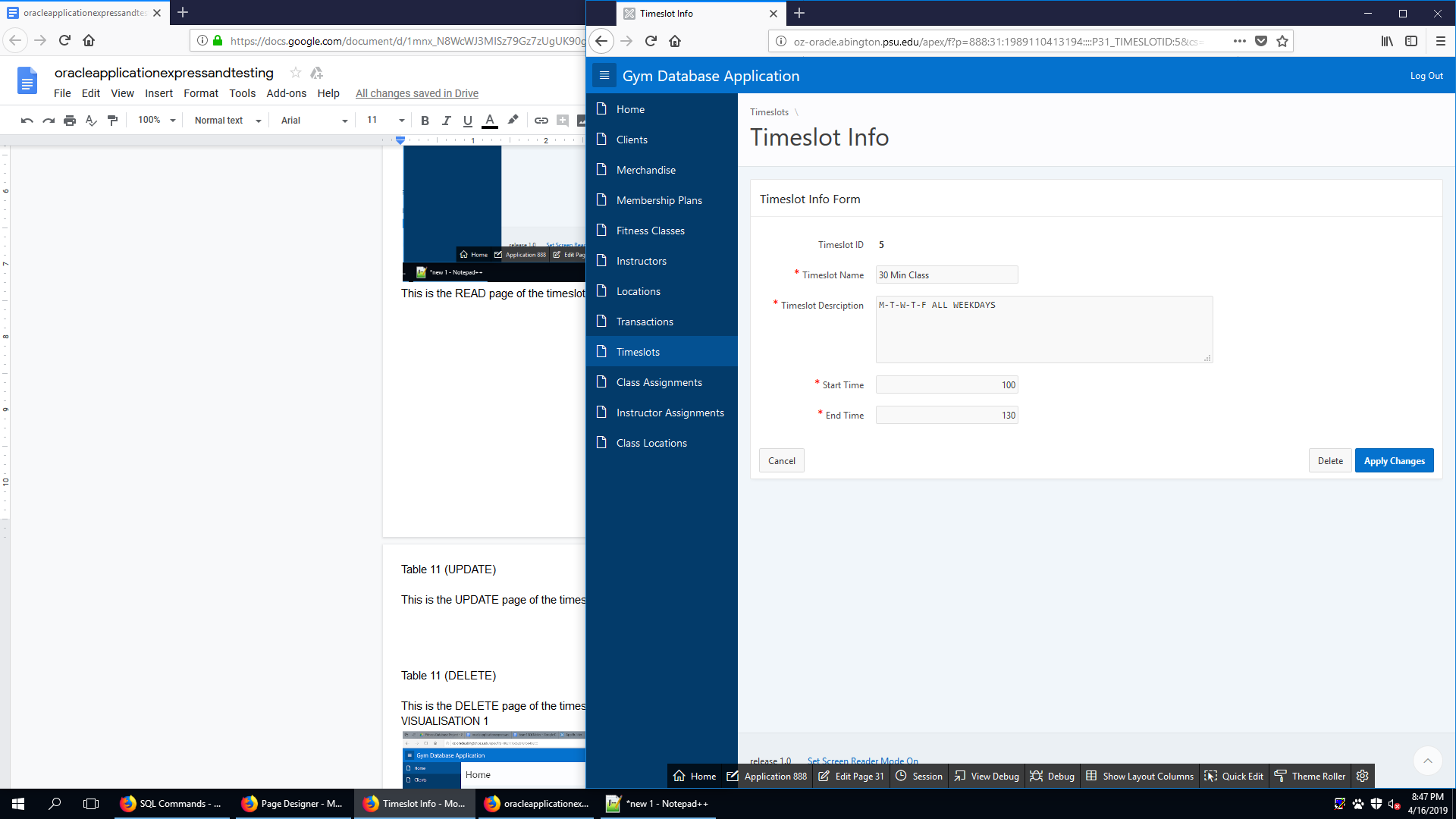
This is the CREATE page of the timeslot table. It allows the user to create new timeslot.

TIMESLOT (READ)



This is the READ page of the timeslot table. It allows the user to see existing timeslot.

TIMESLOT (UPDATE)



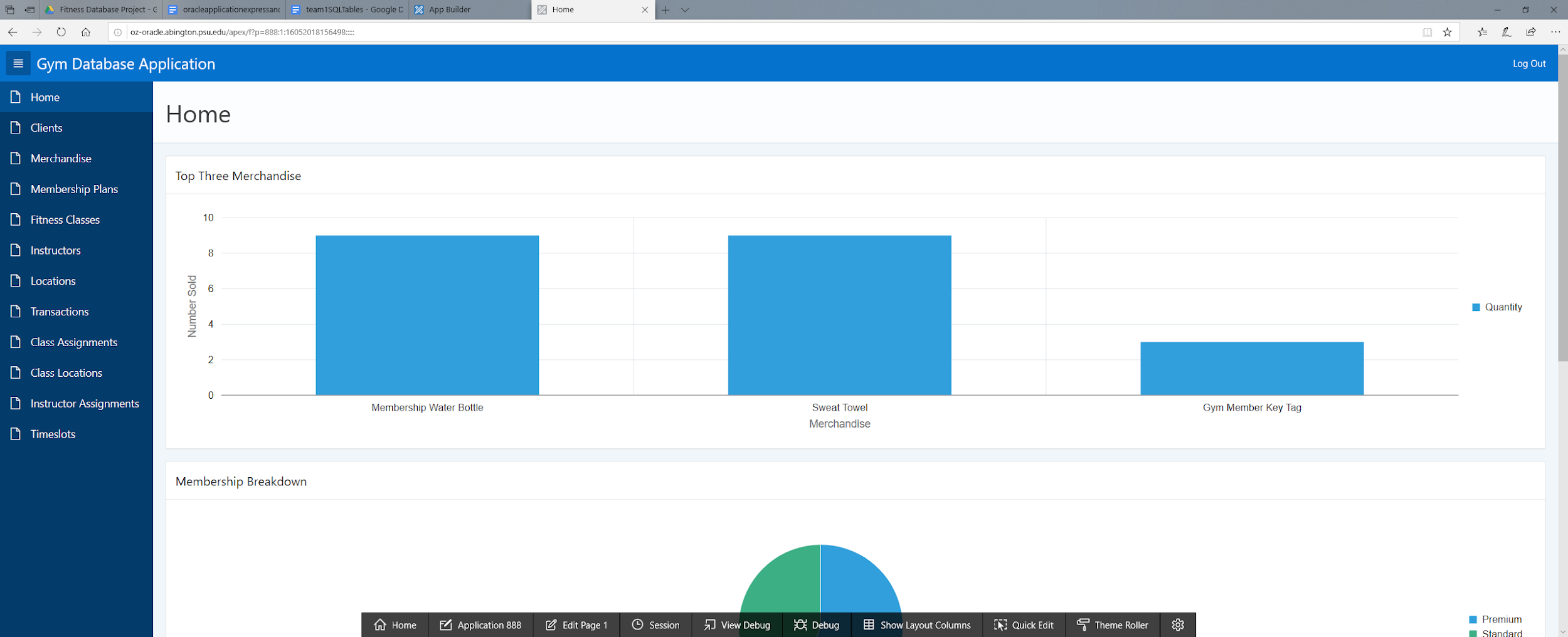
This is the UPDATE page of the timeslot table. It allows the user to update an existing timeslot.

TIMESLOT (DELETE)



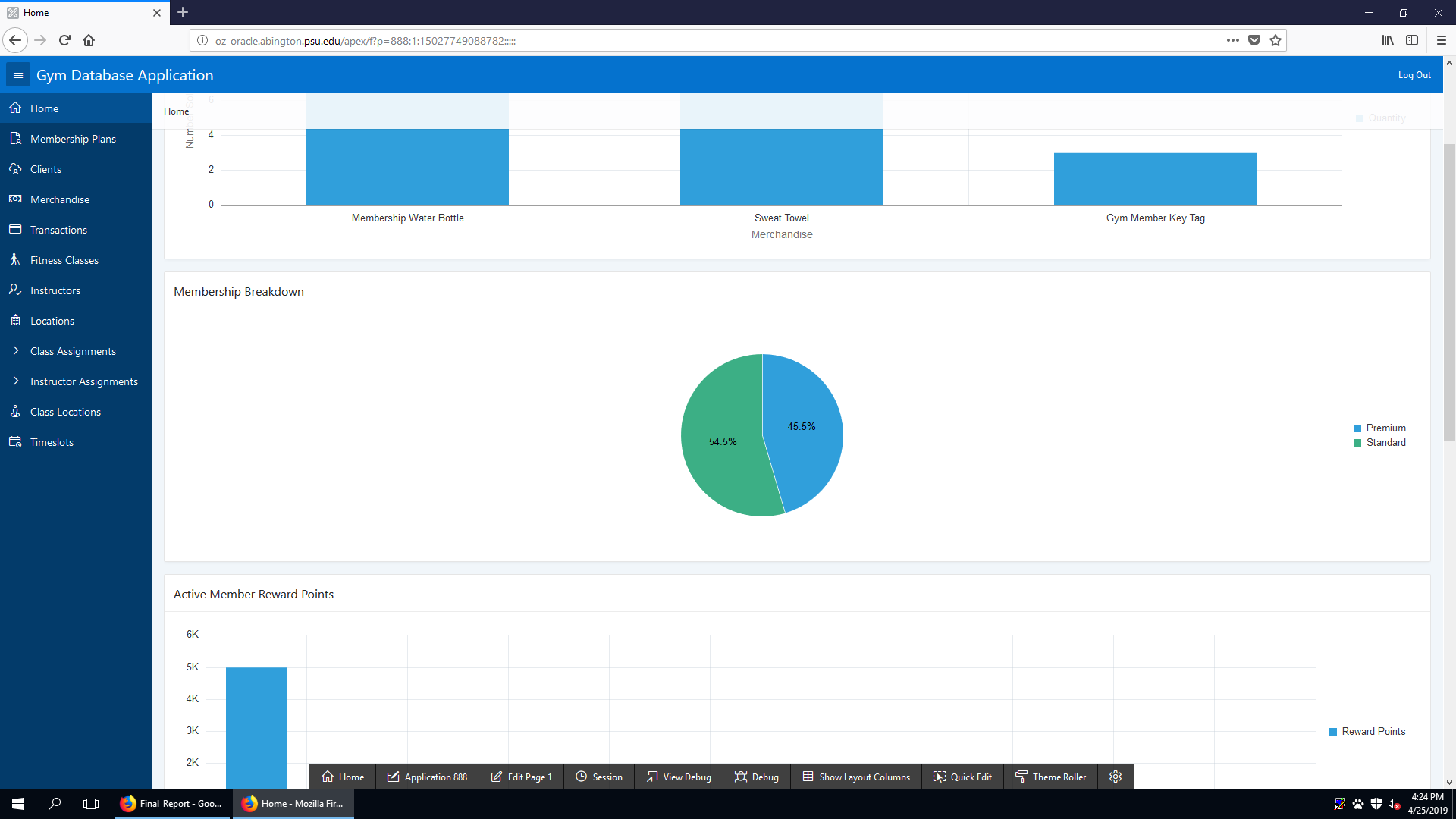
This is the DELETE page of the timeslot table. It allows the user to delete an existing timeslot.

VISUALISATION 1 - TOP THREE MERCHANDISE



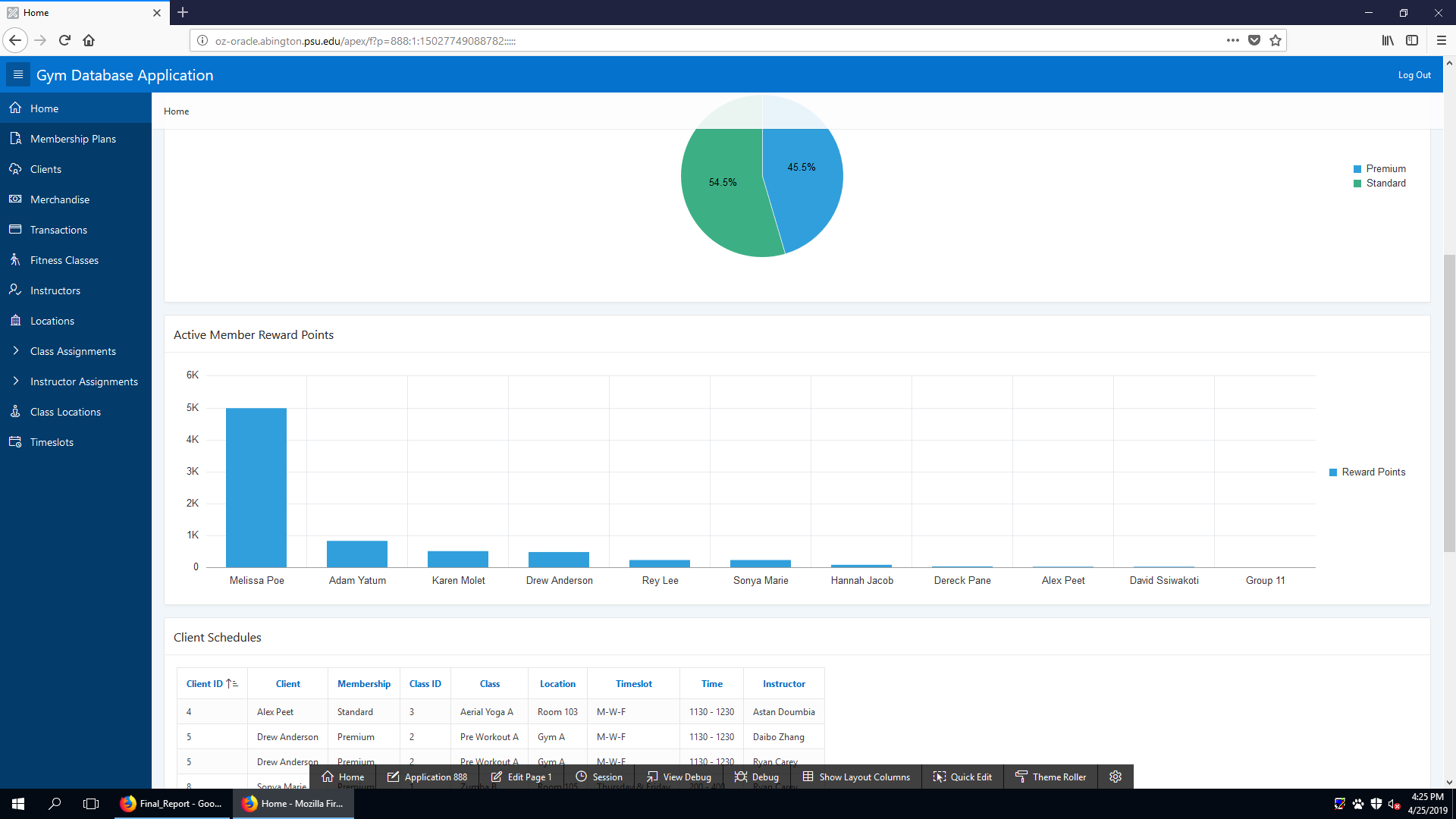
This visualisation displays the top 3 selling merchandise by quantity sold. This allows for an easy visual to see the most popular merchandise sold

VISUALISATION 2 - MEMBERSHIP DISTRUBUTION

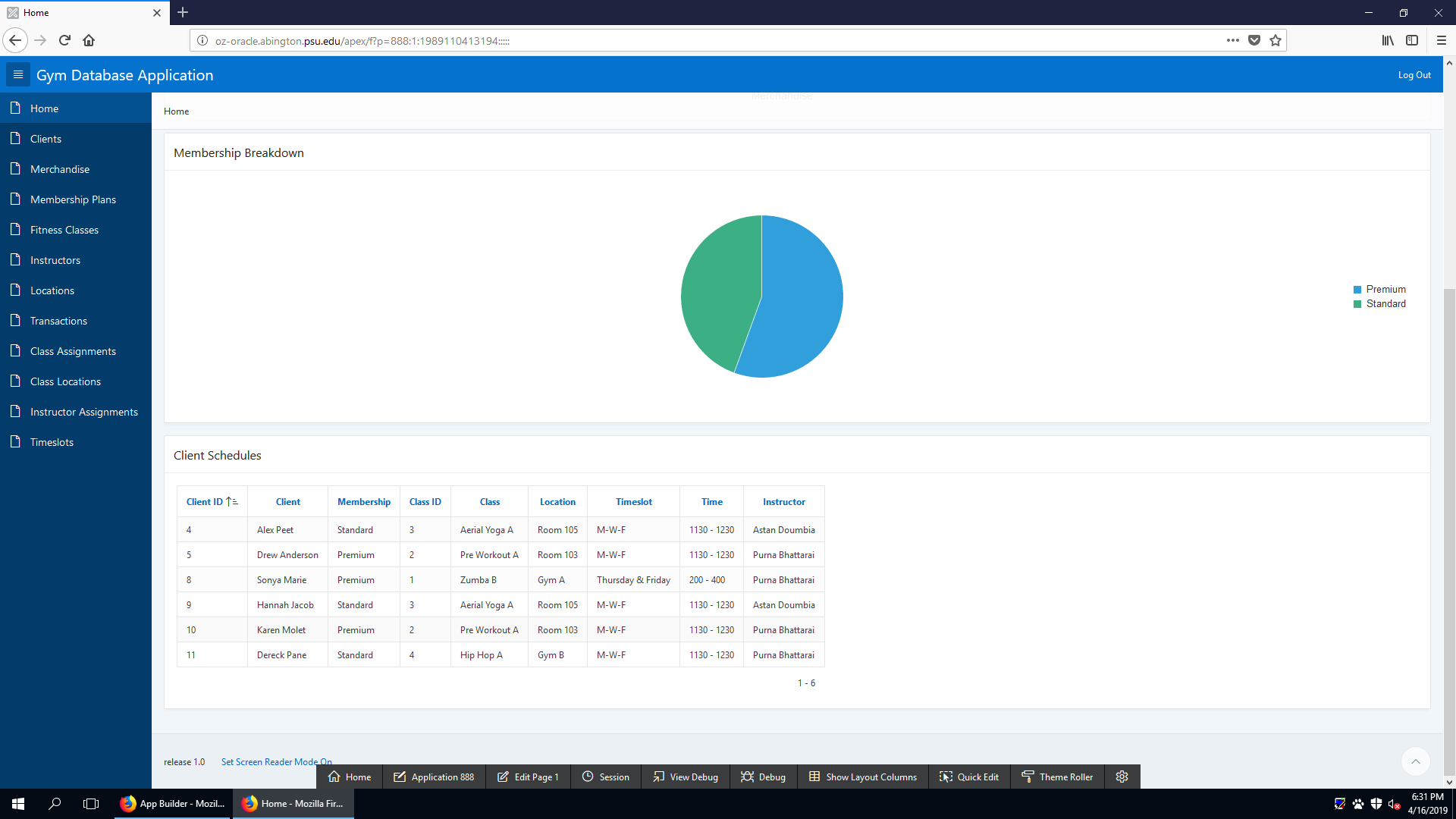


This visualisation shows the proportions of premium versus standard memberships as a pie chart. It allows the user to see which membership is more favored at the moment.

VISUALISATION 3 - REWARD POINTS

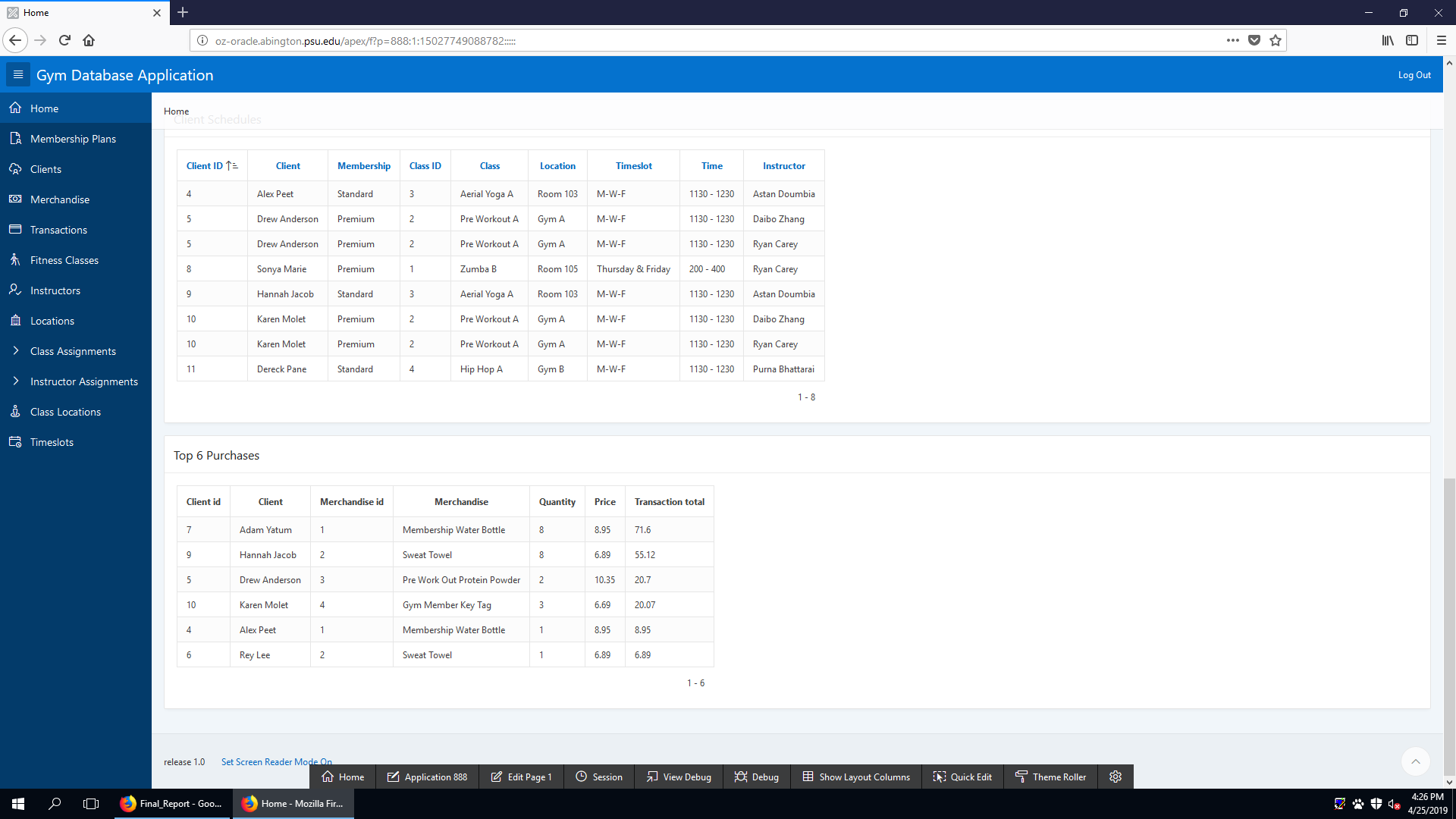


REPORT 1 - CLIENT SCHEDULES



The Client Schedules Report is a one look report that allows the application user to see which clients are assigned to fitness classes. It additionally shows the location and time of each class alongside it’s instructor.

REPORT 2 - TOP 6 PURCHASES



The Top Purchases Report is a one look report that allows the user to see the individual purchases by customer. It shows the customer and merchandise information of the individual purchases sorted by largest purchase descending.