Title

Database class project 2022/2023

Student Name in English	Student Name in Arabic	Student ID	Section	Work percentage
Eba jamal mohammad khalil	اباء جمال محمد خليل	12027796	اثنين-اربعا 11:30-9	50%
Samah asem hasan sheeha	سماح عصام حسن شيحة	12029877	اثنين-اربعا 11:30-9	50%

Date/time

----------This section is intended for the Instructor----------

Topic	**Mark**
Project Requirements and Modeling	
Correctness of Database mapping	
Functional Dependency and Normalization	
Project Tools	
Project Discussion	
Project Completeness	
Project Output Results or reporting (JasperReport, charts, graphs, etc.)	
Project Administration and Management	
Project Report	
Project Idea	
Project Complexity	
Team work	

List of contents

19- Normalization

20- Boyce-Codd Normal Form (BCNF)

21- Gym management system requirements

22- SQL DDL and DML statements

23- Functional Dependency

24- discussion about project GUI work, what each interface do

25- Jasper report

List of tables

1-table trainer

2-table admin

3-table employee

4-table device

5-table exercise

6-table extra info member

7-table member

Abstract

Gym management system is fitness centers to operate the members in an easier way. The administrator, is able to view all the members of fitness center as well as their details. The basic module of the system as follows. This project is a offline based program and it manages the gym members, the personal and

the admin. This system also maintains the student's details, to provides the valuable reports regarding the progress of the gym member.

Introduction

Gyms are become the essential part of our lives, providing best exercise and body building facilities to our society. Therefore, at the management end there are some necessary steps to maintain the records of every individual including trainer, trainees and staff but maintaining the records on paper is very difficult so, it is necessary to have a computerized system that manages all these issues.

Thus working on the management system for gym industry are the basis of our project. We have developed an automated version of the manual system, named as gym management system. This system also provides excellent security of data at to its user.

Project problem

Many gyms suffer from many problems in the management of the club ,and these problems are represented in managing and storing the data of the trainees participating in the club. It is stored in paper records that are at risk of damage and loss, and the problem of searching with difficulty inside the files, which leads to the loss of a lot of time.

Project importance

the importance of searching in a complete management, storing and displaying the data of the trainees participating in the club and managing the data of the employees working in the club and the coach responsible for the participants, in addition to managing the equipment in the club with a statistic display.

Existing system and comparison

Existing system

1- Existing system is a static application so you have to use file system to save record

2-time consuming system

3-no proper system to make announcements

4-importatnt information can be missed while writing on page

5-inefficient and slow search

Proposed system

1-new proposed system allow to user to save record is database

2-fast and easy to use

3-an easy way to make a announcements

4-full fledge information system

5- efficient and proper availability of data

Project objective

The program aims to provide improvements and device to the trainee after measuring the body mass of each player by taking the height and weight to know his physical condition.

System idea

The idea of the simple and easy to use system enables the employee to work and use the program without any difficulties or complications, due to the program's beautiful graphic interface with consistent colors.

Goal

1-The project is basically targeted at those people who would like online management, to make a database that is consistent, reliable and secure

2-To provide correct, complete, ongoing information

3-To develop a well organized information storage system

4-To make good documentation so as to facilitate possible future enhancements

Need of the system

there is always a need of a system that will help gym owner to manage our gym from anywhere at anytime

tools

1-sql developer

2-netbeans

3- Draw.IO

4-jasper soft studio

SQL developer

structured query language(SQL) is a standardized programming language that is used to manage relational database and perform various operations on the data in them. Initially created in the 1970s, SQL is regularly used not only by database administrators, but also by developers writing data integration scripts and data analysts looking to set up and run analytical queries.

The term SQL is pronounced less-Kew-ell or sequel.

SQL is used for the following:

1-modifying database table and index structures

2-adding, updating and deleting rows of data

3-retrive subsets if information from within relational database management system (RDBMSes) this information can be used for transaction processing.

Analytics applications and other applications that require communicating with a relational database

SQL queries and other operations take the form of commands written as statements and are aggregated into programs that enable users to add, modify or retrieve data from database tables.

A table is the most basic unit of a database and consists of rows and columns of data. A single table holds records, and each record is stored in a row of the table. Tables are the most used type of database objects, or structures that hold or reference data in a relational database. Other types of database objects include the following :

1-views are logical representations of data assembled from one or more database tables

2-indexes are lookup tables that help speed up database lookup functions

3-reports consist of data retrieved from one or more tables, usually a subset of that data that is selected based on search criteria.

Each column in a table corresponds to a category of data, for example, costumer name or address while each row contains a data value for the intersecting column.

Relational database are relational because they are composed of tables that relate to each other. For example, a SQL database used for customer service can have one table for costumer names and address and other tables that hold information about specific purchases, product codes and customer contacts.

A table used to track customer contacts usually uses a unique customer identifier called a key or primary key to reference the customer's record in a separate table used to store customer data, such as name and contact information.

SQL commands and syntax

SQL is fundamentally, a programming language designed for accessing, modifying and extracting information from relational database. As a programming language, SQL has commands and a syntax for issuing those commands

SQL commands are divided into several different types, including the following:

1-data definition language(DDL) commands are also called data definition commands because they are used to define data tables.

2-data manipulation language(DML) commands are used to manipulate data in existing tables by adding, changing or removing data, unlike DDL commands that define how data is stored, DML commands operate in the tables defined with DDL commands.

3-data query language consists of just one command, select, used to get specific data from tables, this command is sometimes grouped with the DML commands.

4-data control language commands are used to grant or revoke user access privileges

5-transaction control language commands are used to change the state of some data for example, to commit transaction changes or rollback transaction changes.

SQL syntax, the set of rules for how SQL statements are written and formatted, is similar to other programming languages. Some components of SQL syntax include the following:

SQL statements start with a SQL command and end with a semicolon(;), for example

SELECT * from customers;

This select statement extracts all of the contents of a table called customers

Requirements to run the program

1-Windows 10-64 bit

2-ram 4 GB

3-hard :20 GB free

4-cpu: core i

Techniques

Java

Reference

1-stack over flow

https://stackoverflow.com/

2-Oracle Help Centre

https://docs.oracle.com/en/

3-DesignCap

https://www.designcap.com/

4-Convertio

https://convertio.co/ar/mp4-mp3/

5-removeby

https://www.remove.bg/

conclusion

based on this project entitled "Gym Management System", which was developed and had solved the problems that were identified in the current system as follows:

Gym Management System hopefully could solve the problem of member registration by performing the produce online which is more efficient and less expensive

1-the development of

2-the development of Gym Management System could also solve the problem of membership renewal by renewing

membership dates faster and searching process is done more effectively

3- the development of Gym Management System hopefully could solve the problem of non-member transaction by payment records be done quickly and reduce errors

4- the development of Gym Management System could solve the problem of attendance by recording date and exact time of every gym user automatically from the time of entry to the time of exit

5- the development of Gym Management System hopefully could solve the problem of financial reports by generating proof of output produced per period in the form of a report from the processing of transactions that can facilitate the process of checking the income of member registration, membership renewal and non-member transactions.

Future works

Gym Management System can be developed by uploading the database on the server so that the club owner can follow up on the club's status from a distance. This can be used in clubs with more branches because the owner suffers from being present in all branches at the same time, while he can do so in case of using the program connected to the same database that manages all club data with ease.

Normalization

- all tables are in first noraml form

(Each table cell should contain a single value.

(Each record needs to be unique.)

- all tables are in second normal form

(Be in 1NF , Single Column Primary Key that does not

functionally dependant on any subset of candidate key relation)

- all tables are in third normal form

(Be in 2NF , Has no transitive functional dependencies)

Table admin

1NF : it achieves first normal form rule : becauese it dose not contains multivalued attributes .

2NF : it is in 1NF but we can't apply it because it does not contain composite keys.

3NF : it is on 3NF because it is does not contain transitive dependency attributes.

Table member

1NF : it achieves first normal form rule : becauese it dose not contains multivalued attributes .

2NF : it is in 1NF but we can't apply it because it does not contain composite keys.

3NF : it is on 3NF because it is does not contain transitive dependency attributes.

Table employee

1NF : it achieves first normal form rule : becauese it dose not contains multivalued attributes .

2NF : it is in 1NF but we can't apply it because it does not contain composite keys.

3NF : it is on 3NF because it is does not contain transitive dependency attributes.

Table exercise

1NF : it achieves first normal form rule : becauese it dose not contains multivalued attributes .

2NF : it is in 1NF but we can't apply it because it does not contain composite keys.

3NF : it is on 3NF because it is does not contain transitive dependency attributes.

Table device

1NF : it achieves first normal form rule : becauese it dose not contains multivalued attributes .

2NF : it is in 1NF but we can't apply it because it does not contain composite keys.

3NF : it is on 3NF because it is does not contain transitive dependency attributes.

Table extra info member

1NF : it achieves first normal form rule : becauese it dose not contains multivalued attributes .

2NF : it is in 1NF but we can't apply it because it does not contain composite keys.

3NF : it is on 3NF because it is does not contain transitive dependency attributes.

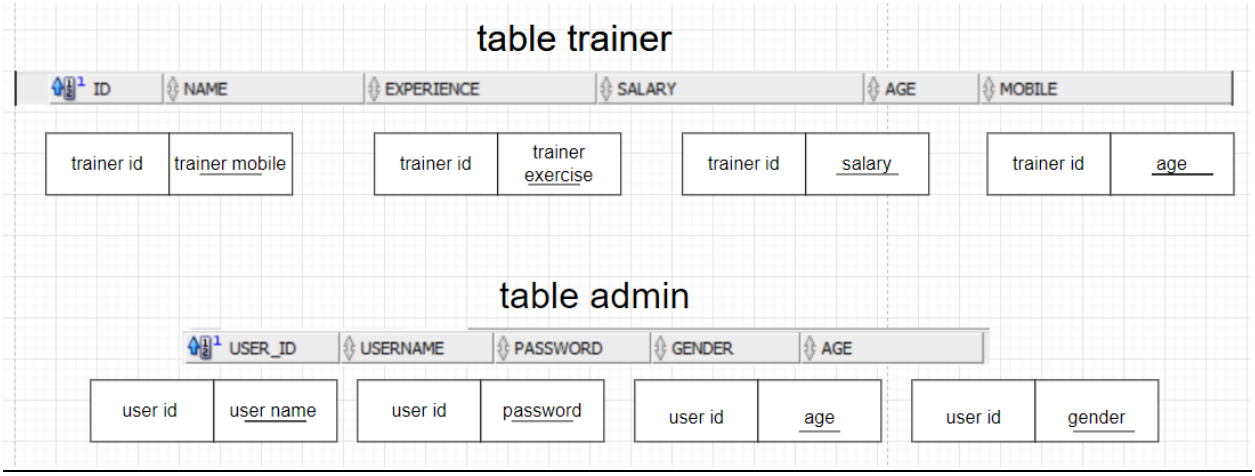
Table trainer

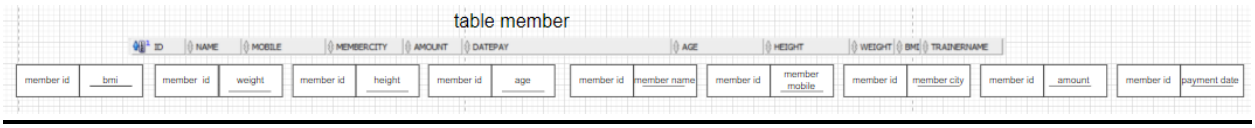
1NF : it achieves first normal form rule : becauese it dose not contains multivalued attributes .

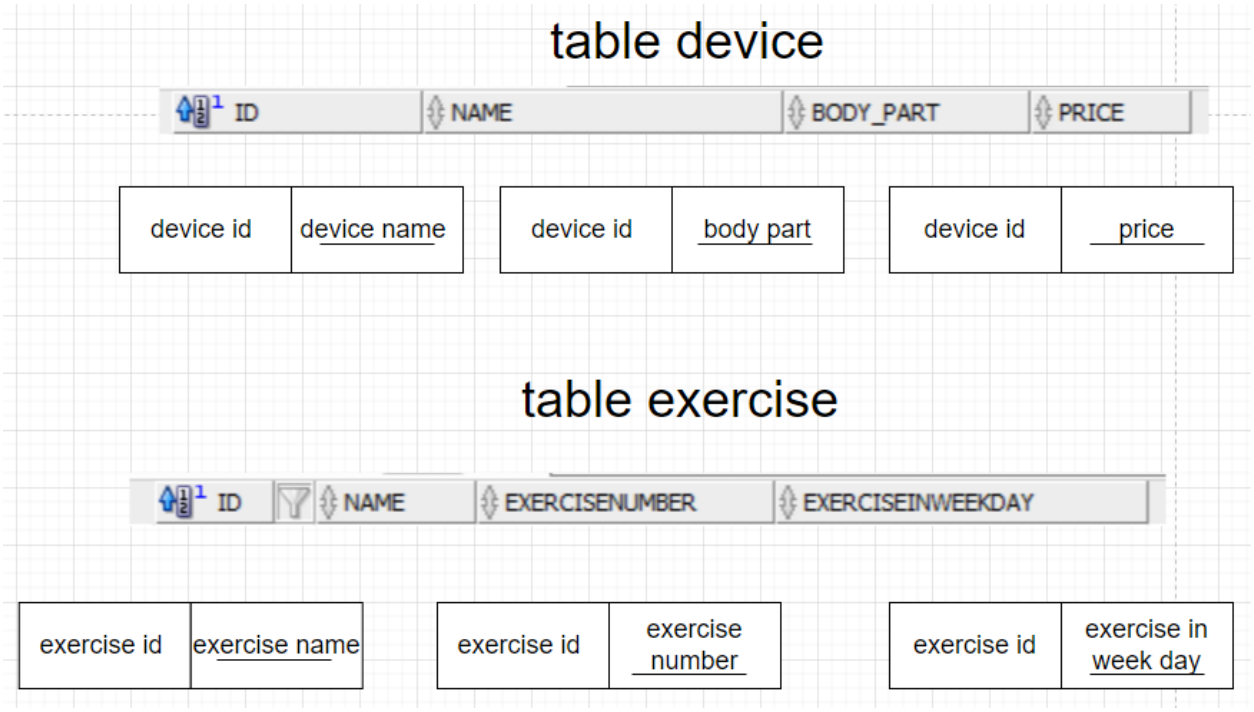
2NF : it is in 1NF but we can't apply it because it does not contain composite keys.

3NF : it is on 3NF because it is does not contain transitive dependency attributes

Boyce-Codd Normal Form (BCNF)







Gym management system requirements

- the gym contains members each member has a member Id, member City, member Name, member Contact,

status, password, and contains trainers each trainer has a trainer Id, trainer Name, Trainer Experience, trainer Contact.

- the gym contains employee each member has a employee Id, employee City, employee Name, employee mobile, salary .

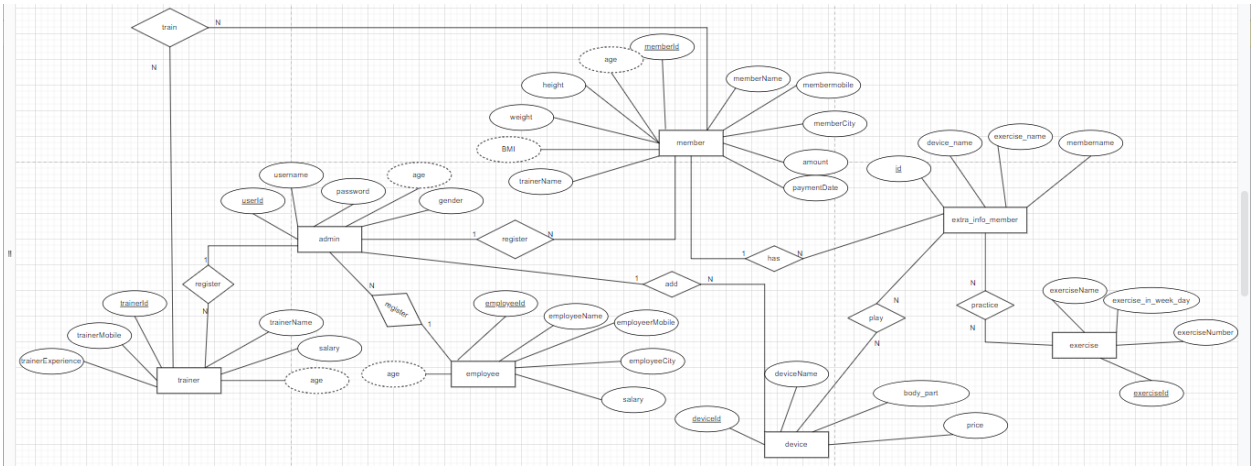
- a administration who is the owner of the gym and controls all processes has a user Id, password, the administration can register any member and manages all payment each administration gives a fee to several trainers.

-the gym contain devices each device have device id, device name, body part, price.

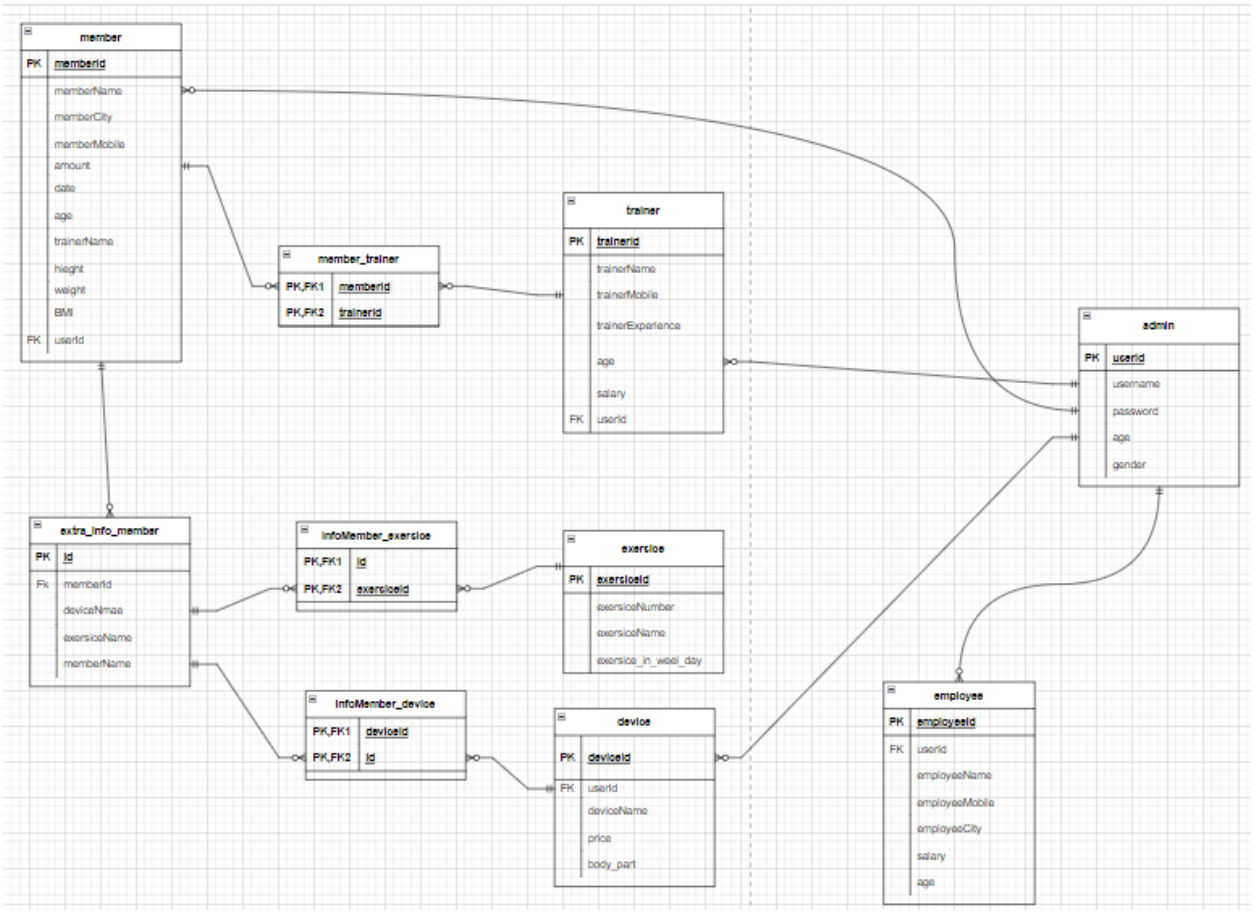
-each trainer give member some exercises each exercises hava exercise name, exercise id, exercise number, exercise in week day

- body type contains body type information of the member such as Age, height, weight, body Id, each member has a body type.

Project EER

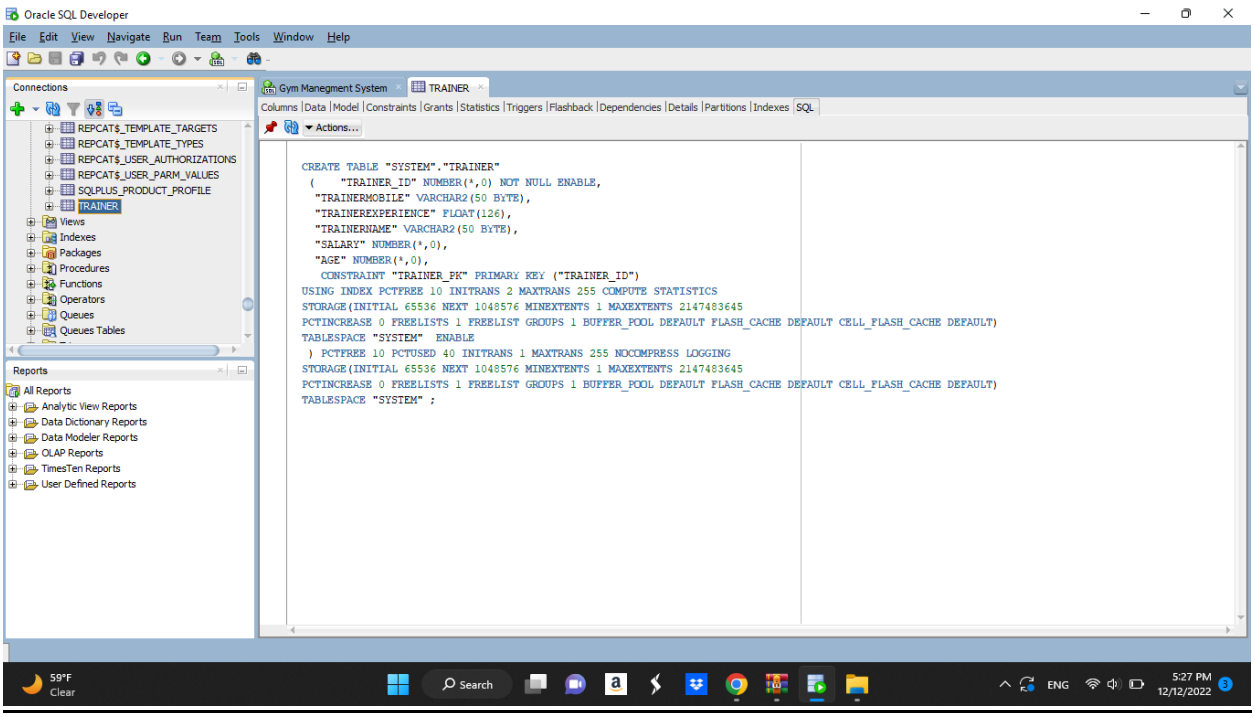


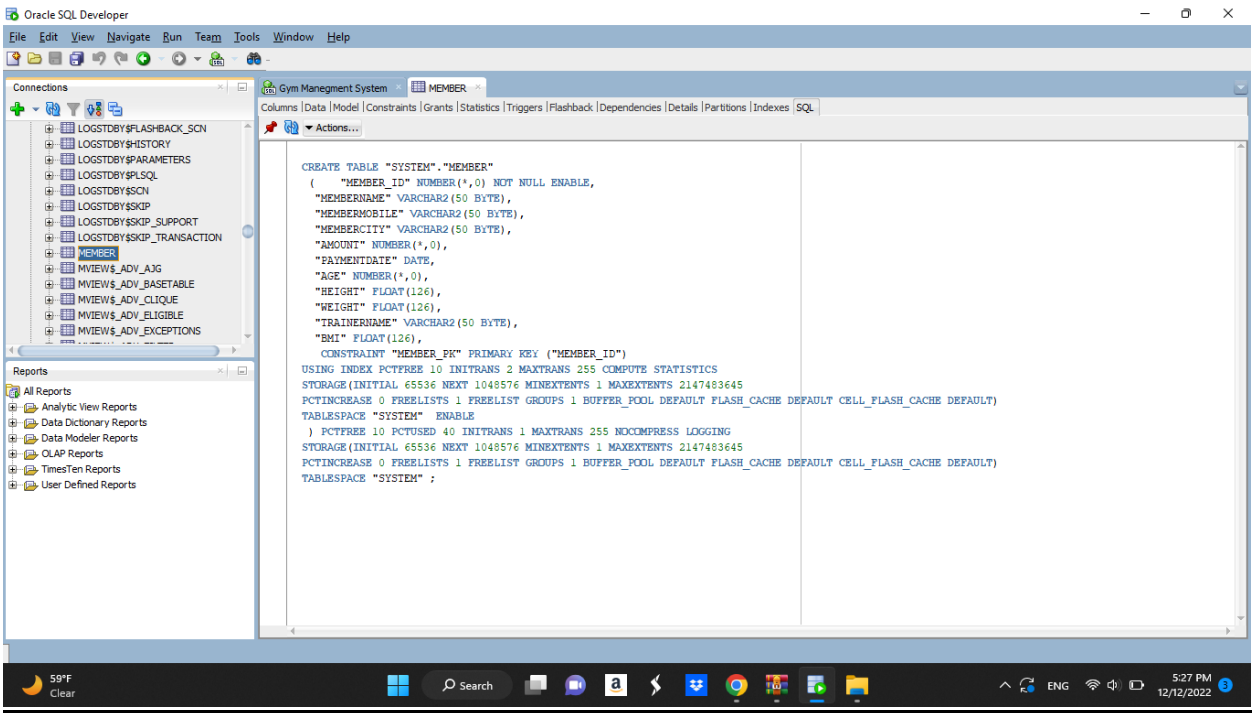
Project Uml

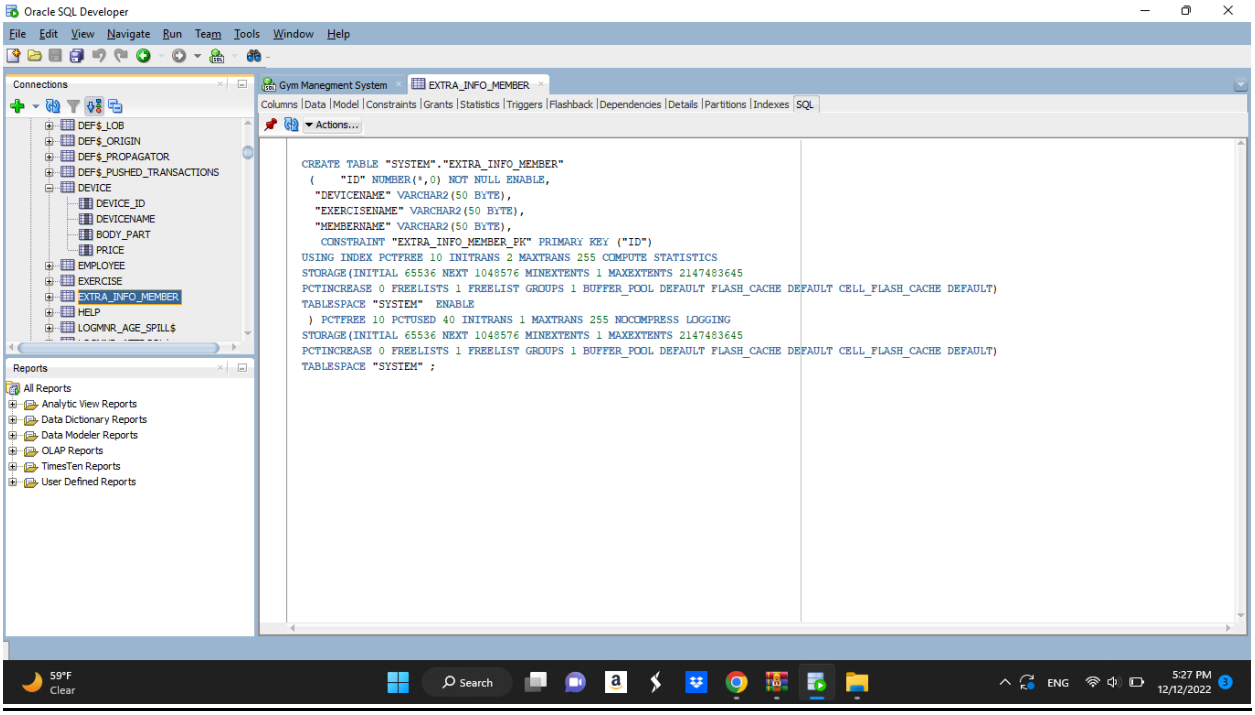


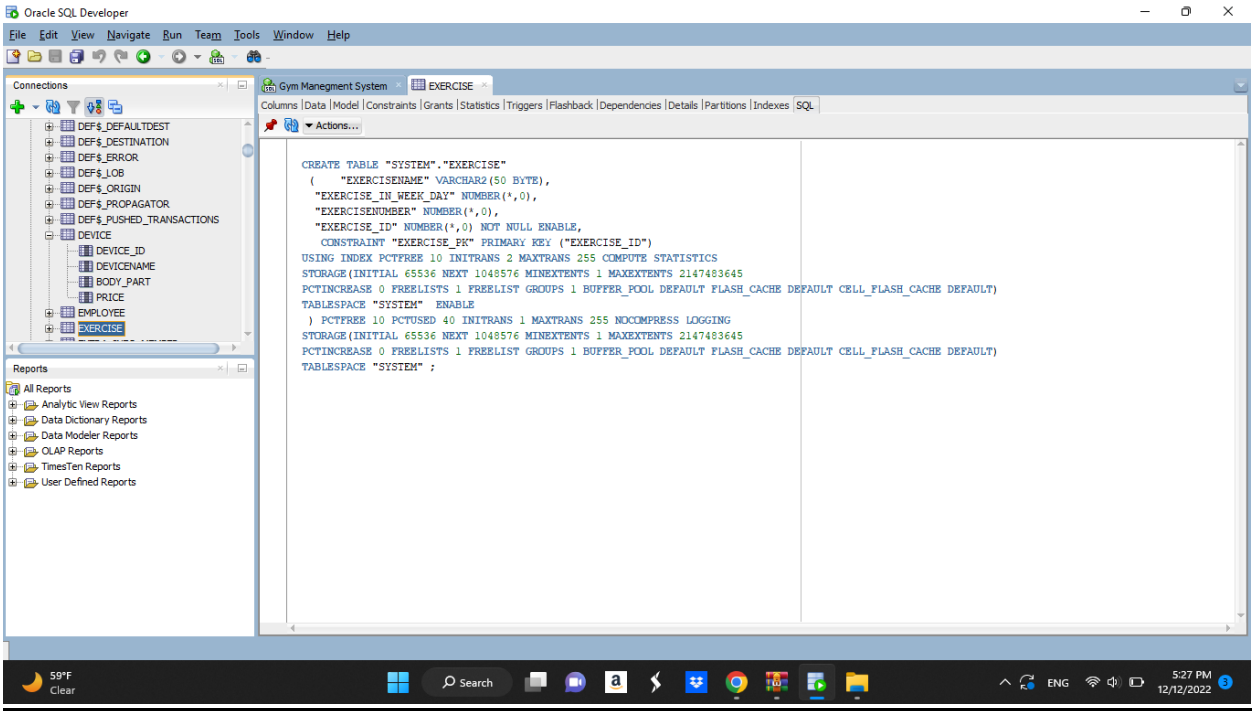
SQL DDL and DML statements

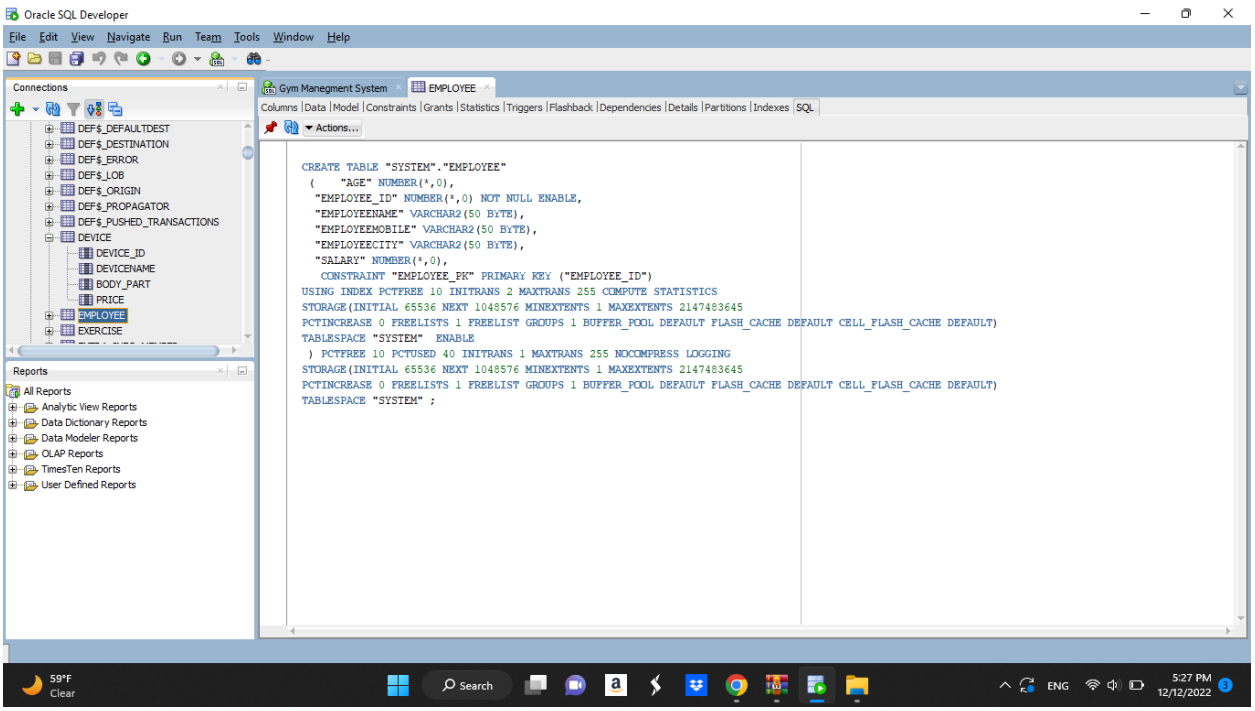
Create statement

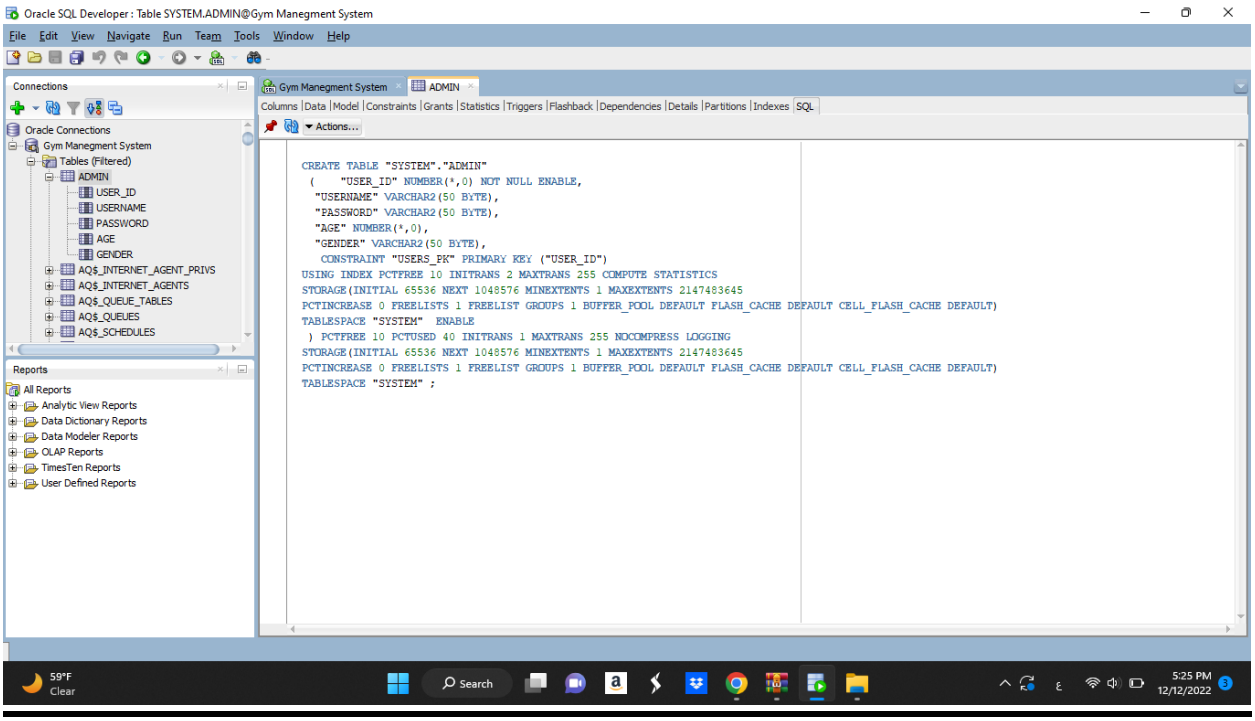


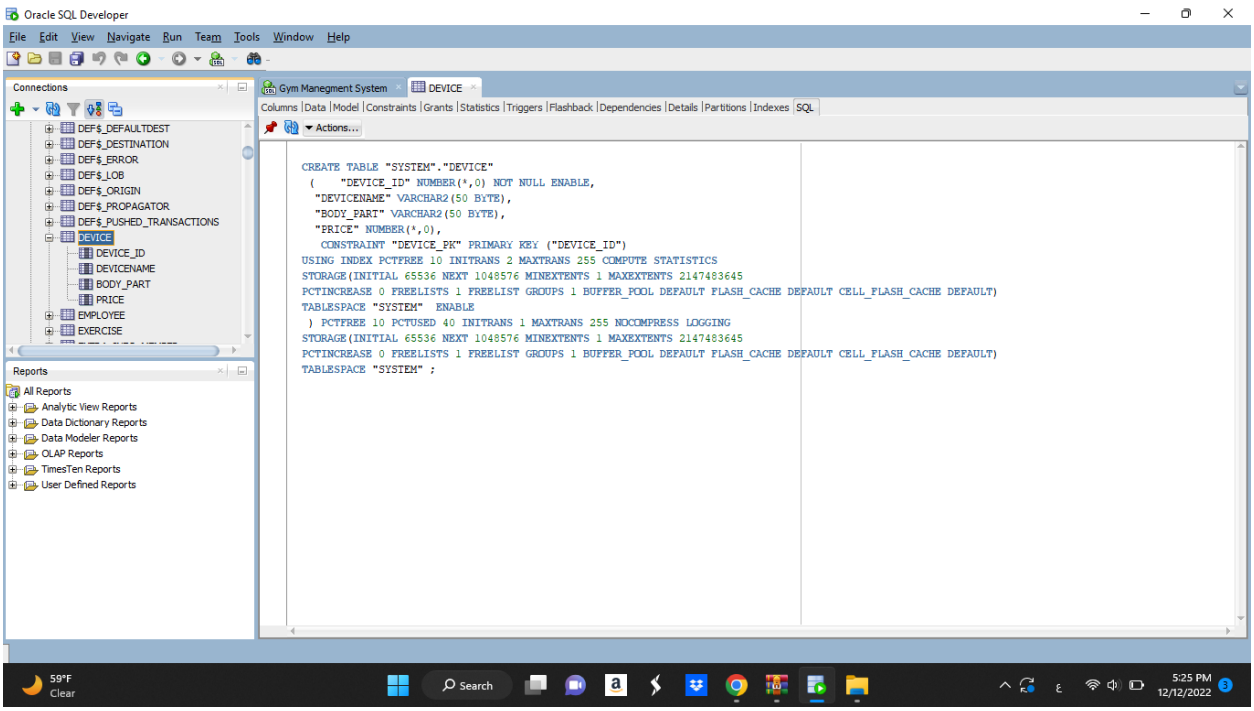



Oracle SQL Developer
File Edit View Navigate Run Team Tools Window Help
Connections
DEF$_LOB
DEF$_ORIGIN
DEF$_PROPAGATOR
DEF$_PUSHED_TRANSACTIONS
DEVICE
DEVICE_ID
DEVICENAME
BODY_PART
PRICE
EMPLOYEE
EXERCISE
EXTRA_INFO_MEMBER
HELP
LOGMNR_AGE_SPILL$
Reports
All Reports
Analytic View Reports
Data Dictionary Reports
Data Modeler Reports
OLAP Reports
TimesTen Reports
User Defined Reports
Gym Managment System
EXTRA_INFO_MEMBER
Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes SQL
Actions...
CREATE TABLE "SYSTEM"."EXTRA_INFO_MEMBER"
("ID" NUMBER(*,0) NOT NULL ENABLE,
"DEVICENAME" VARCHAR2(50 BYTE),
"EXERCISENAME" VARCHAR2(50 BYTE),
"MEMBERNAME" VARCHAR2(50 BYTE),
CONSTRAINT "EXTRA_INFO_MEMBER_PK" PRIMARY KEY ("ID")
USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
TABLESPACE "SYSTEM" ENABLE
) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
TABLESPACE "SYSTEM" ;
59°F
Clear
Search
ENG
5:27 PM
12/12/2022


Oracle SQL Developer
File Edit View Navigate Run Team Tools Window Help
Connections
DEF$_DEFAULTDEST
DEF$_DESTINATION
DEF$_ERROR
DEF$_LOB
DEF$_ORIGIN
DEF$_PROPAGATOR
DEF$_PUSHED_TRANSACTIONS
DEVICE
DEVICE_ID
DEVICENAME
BODY_PART
PRICE
EMPLOYEE
EXERCISE
Reports
All Reports
Analytic View Reports
Data Dictionary Reports
Data Modeler Reports
OLAP Reports
TimesTen Reports
User Defined Reports
Gym Managment System
EXERCISE
Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes SQL
Actions...
CREATE TABLE "SYSTEM"."EXERCISE"
("EXERCISENAME" VARCHAR2(50 BYTE),
"EXERCISE_IN_WEEK_DAY" NUMBER(*,0),
"EXERCISENUMBER" NUMBER(*,0),
"EXERCISE_ID" NUMBER(*,0) NOT NULL ENABLE,
CONSTRAINT "EXERCISE_PK" PRIMARY KEY ("EXERCISE_ID")
USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
TABLESPACE "SYSTEM" ENABLE
) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
TABLESPACE "SYSTEM" ;
59°F
Clear
Search
ENG
5:27 PM
12/12/2022


Oracle SQL Developer
File Edit View Navigate Run Team Tools Window Help
Connections
DEF$_DEFAULTDEST
DEF$_DESTINATION
DEF$_ERROR
DEF$_LOB
DEF$_ORIGIN
DEF$_PROPAGATOR
DEF$_PUSHED_TRANSACTIONS
DEVICE
DEVICE_ID
DEVICENAME
BODY_PART
PRICE
EMPLOYEE
EXERCISE
Reports
All Reports
Analytic View Reports
Data Dictionary Reports
Data Modeler Reports
OLAP Reports
TimesTen Reports
User Defined Reports
Gym Managment System
EMPLOYEE
Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes SQL
Actions...
CREATE TABLE "SYSTEM"."EMPLOYEE"
("AGE" NUMBER(*,0),
"EMPLOYEE_ID" NUMBER(*,0) NOT NULL ENABLE,
"EMPLOYEENAME" VARCHAR2(50 BYTE),
"EMPLOYEEMOBILE" VARCHAR2(50 BYTE),
"EMPLOYEECITY" VARCHAR2(50 BYTE),
"SALARY" NUMBER(*,0),
CONSTRAINT "EMPLOYEE_PK" PRIMARY KEY ("EMPLOYEE_ID")
USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
TABLESPACE "SYSTEM" ENABLE
) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
TABLESPACE "SYSTEM" ;
59°F
Clear
Search
ENG
5:27 PM
12/12/2022


Oracle SQL Developer : Table SYSTEM.ADMIN@Gym Managment System
File Edit View Navigate Run Team Tools Window Help
Connections
Oracle Connections
Gym Managment System
Tables (Filtered)
ADMIN
USER_ID
USERNAME
PASSWORD
AGE
GENDER
AQ$_INTERNET_AGENT_PRIVS
AQ$_INTERNET_AGENTS
AQ$_QUEUE_TABLES
AQ$_QUEUES
AQ$_SCHEDULES
Reports
All Reports
Analytic View Reports
Data Dictionary Reports
Data Modeler Reports
OLAP Reports
TimesTen Reports
User Defined Reports
Gym Managment System
ADMIN
Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes SQL
Actions...
CREATE TABLE "SYSTEM"."ADMIN"
("USER_ID" NUMBER(*,0) NOT NULL ENABLE,
"USERNAME" VARCHAR2(50 BYTE),
"PASSWORD" VARCHAR2(50 BYTE),
"AGE" NUMBER(*,0),
"GENDER" VARCHAR2(50 BYTE),
CONSTRAINT "USERS_PK" PRIMARY KEY ("USER_ID")
USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
TABLESPACE "SYSTEM" ENABLE
) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
TABLESPACE "SYSTEM" ;
59°F
Clear
Search
5:25 PM
12/12/2022



Insert into table trainer

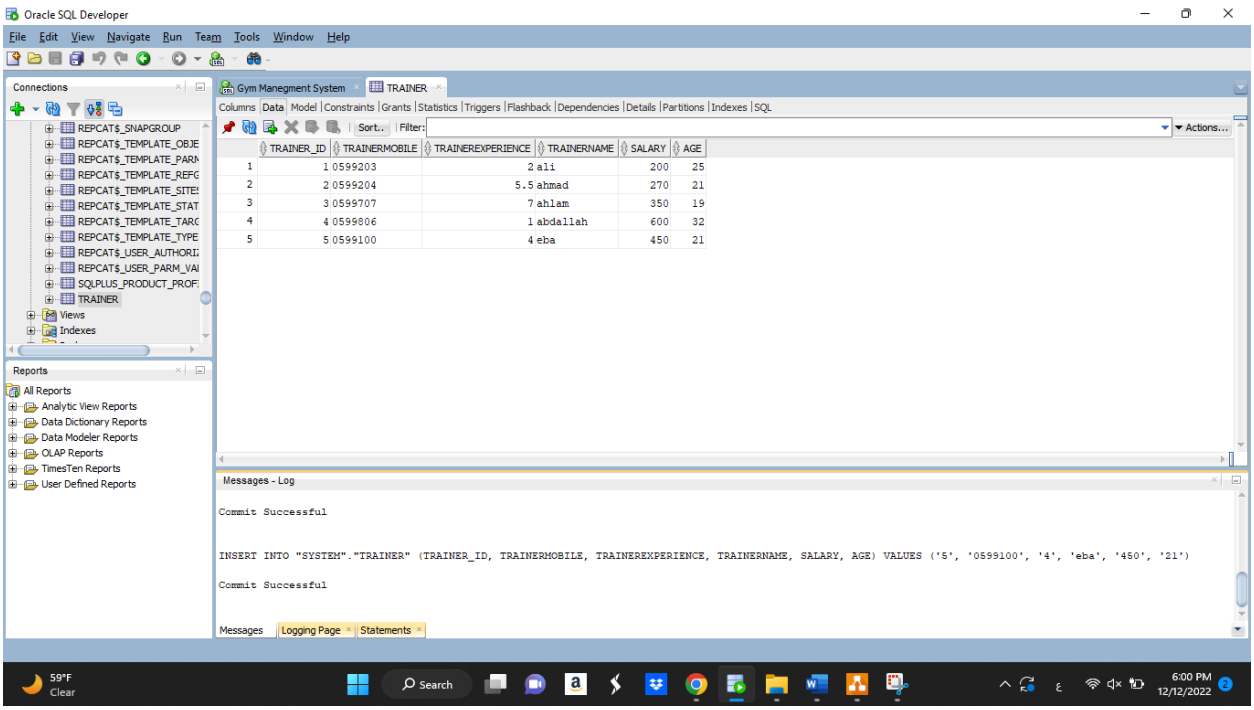
INSERT INTO "SYSTEM"."TRAINER" (TRAINER_ID, TRAINERMOBILE, TRAINEREXPERIENCE, TRAINERNAME, SALARY, AGE) VALUES ('1', '0599203', '2', 'ali', '200', '25')

INSERT INTO "SYSTEM"."TRAINER" (TRAINER_ID, TRAINERMOBILE, TRAINEREXPERIENCE, TRAINERNAME, SALARY, AGE) VALUES ('2', '0599204', '5.5', 'ahmad', '270', '21')

INSERT INTO "SYSTEM"."TRAINER" (TRAINER_ID, TRAINERMOBILE, TRAINEREXPERIENCE, TRAINERNAME, SALARY, AGE) VALUES ('3', '0599707', '7', 'ahlam', '350', '19')

INSERT INTO "SYSTEM"."TRAINER" (TRAINER_ID, TRAINERMOBILE, TRAINEREXPERIENCE, TRAINERNAME, SALARY, AGE) VALUES ('4', '0599806', '1', 'abdallah', '600', '32')

INSERT INTO "SYSTEM"."TRAINER" (TRAINER_ID, TRAINERMOBILE, TRAINEREXPERIENCE, TRAINERNAME, SALARY, AGE) VALUES ('5', '0599100', '4', 'eba', '450', '21')



Insert into table member

INSERT INTO "SYSTEM"."MEMBER" (MEMBER_ID, MEMBERNAME, MEMBERMOBILE, MEMBERCITY, AMOUNT, PAYMENTDATE, AGE, HEIGHT, WEIGHT, TRAINERNAME, BMI) VALUES ('1', 'sara', '0599201', 'nablus', '50', TO_DATE('2022-12-03 18:05:01', 'YYYY-MM-DD HH24:MI:SS'), '18', '1.60', '60', 'eba', '23.4')

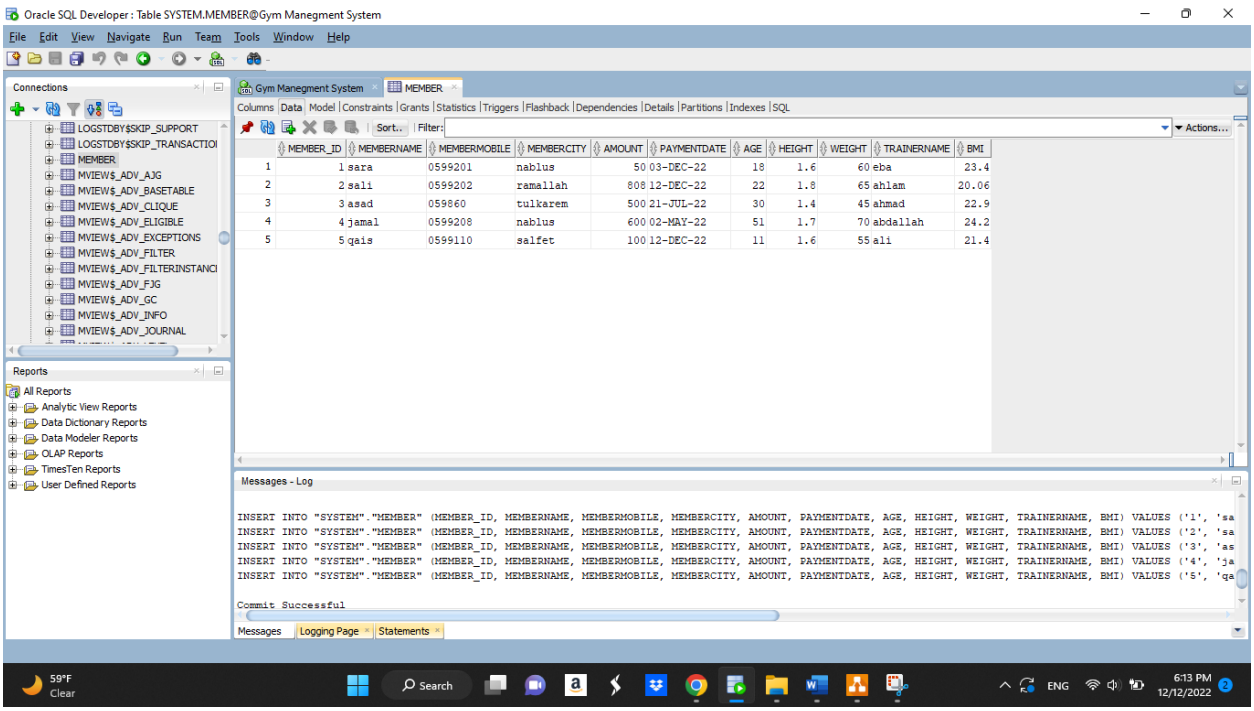
INSERT INTO "SYSTEM"."MEMBER" (MEMBER_ID, MEMBERNAME, MEMBERMOBILE, MEMBERCITY, AMOUNT, PAYMENTDATE, AGE, HEIGHT, WEIGHT, TRAINERNAME, BMI) VALUES ('2', 'sali', '0599202', 'ramallah', '808', TO_DATE('2022-

12-12 18:04:57', 'YYYY-MM-DD HH24:MI:SS'), '22', '1.8', '65', 'ahlam', '20.06')

INSERT INTO "SYSTEM"."MEMBER" (MEMBER_ID, MEMBERNAME, MEMBERMOBILE, MEMBERCITY, AMOUNT, PAYMENTDATE, AGE, HEIGHT, WEIGHT, TRAINERNAME, BMI) VALUES ('3', 'asad', '059860', 'tulkarem', '500', TO_DATE('2022-07-21 18:05:57', 'YYYY-MM-DD HH24:MI:SS'), '30', '1.4', '45', 'ahmad', '22.9')

INSERT INTO "SYSTEM"."MEMBER" (MEMBER_ID, MEMBERNAME, MEMBERMOBILE, MEMBERCITY, AMOUNT, PAYMENTDATE, AGE, HEIGHT, WEIGHT, TRAINERNAME, BMI) VALUES ('4', 'jamal', '0599208', 'nablus', '600', TO_DATE('2022-05-02 18:06:51', 'YYYY-MM-DD HH24:MI:SS'), '51', '1.7', '70', 'abdallah', '24.2')

INSERT INTO "SYSTEM"."MEMBER" (MEMBER_ID, MEMBERNAME, MEMBERMOBILE, MEMBERCITY, AMOUNT, PAYMENTDATE, AGE, HEIGHT, WEIGHT, TRAINERNAME, BMI) VALUES ('5', 'qais', '0599110', 'salfet', '100', TO_DATE('2022-12-12 18:08:06', 'YYYY-MM-DD HH24:MI:SS'), '11', '1.6', '55', 'ali', '21.4')



Insert into table extra info member

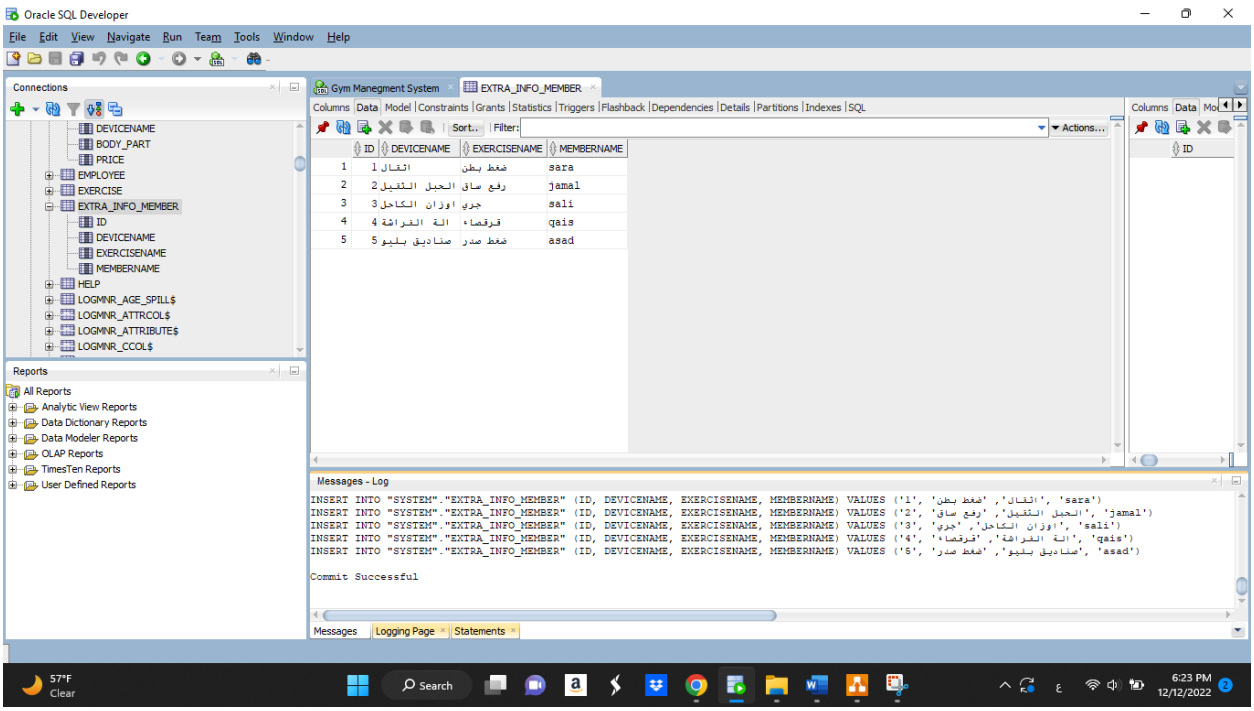
INSERT INTO "SYSTEM"."EXTRA_INFO_MEMBER" (ID, DEVICENAME, EXERCISENAME, MEMBERNAME) VALUES('sara' ,'ضغط بطن' ,'اثقال' ,'1')

INSERT INTO "SYSTEM"."EXTRA_INFO_MEMBER" (ID, DEVICENAME, EXERCISENAME, MEMBERNAME) VALUES('jamal' ,'رفع ساق' ,'الحبل الثقيل' ,'2')

INSERT INTO "SYSTEM"."EXTRA_INFO_MEMBER" (ID, DEVICENAME, EXERCISENAME, MEMBERNAME) VALUES('sali' ,'جري' ,'اوزان الكاحل' ,'3')

INSERT INTO "SYSTEM"."EXTRA_INFO_MEMBER" (ID, DEVICENAME, EXERCISENAME, MEMBERNAME) VALUES('qais' ,'قرقصاء' ,'الة الفراشة' ,'4')

INSERT INTO "SYSTEM"."EXTRA_INFO_MEMBER" (ID, DEVICENAME, EXERCISENAME, MEMBERNAME) VALUES('sara' ,'ضغط بطن' ,'اثقال' ,'1')



Insert into table exercise

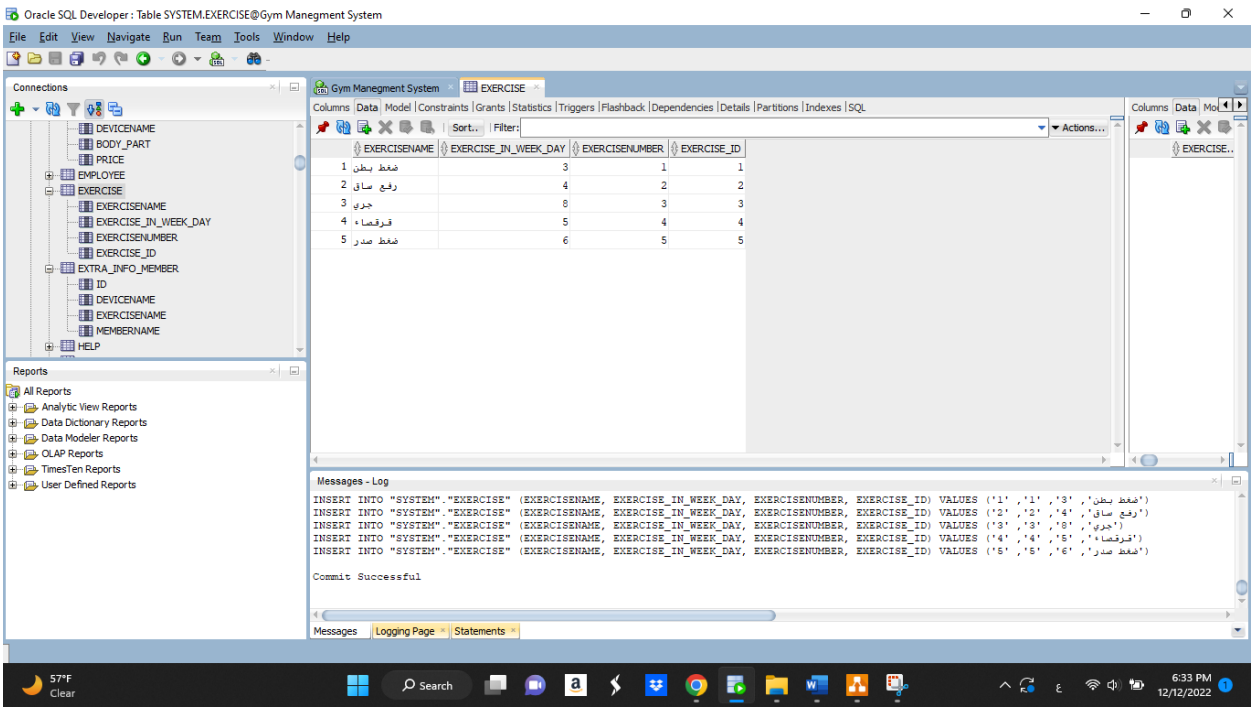
INSERT INTO "SYSTEM"."EXERCISE" (EXERCISENAME, EXERCISE_IN_WEEK_DAY, EXERCISENUMBER, EXERCISE_ID)
VALUES('ضغط بطن', '3' ,'1' ,'1')

INSERT INTO "SYSTEM"."EXERCISE" (EXERCISENAME, EXERCISE_IN_WEEK_DAY, EXERCISENUMBER, EXERCISE_ID)
VALUES('رفع ساق', '4' ,'2' ,'2')

INSERT INTO "SYSTEM"."EXERCISE" (EXERCISENAME, EXERCISE_IN_WEEK_DAY, EXERCISENUMBER, EXERCISE_ID)
VALUES('جري', '8' ,'3' ,'3')

INSERT INTO "SYSTEM"."EXERCISE" (EXERCISENAME, EXERCISE_IN_WEEK_DAY, EXERCISENUMBER, EXERCISE_ID)
VALUES('قرقصاء', '5' ,'4' ,'4')

INSERT INTO "SYSTEM"."EXERCISE" (EXERCISENAME, EXERCISE_IN_WEEK_DAY, EXERCISENUMBER, EXERCISE_ID)
VALUES('ضغط صدر', '6' ,'5' ,'5')



Insert into table employee

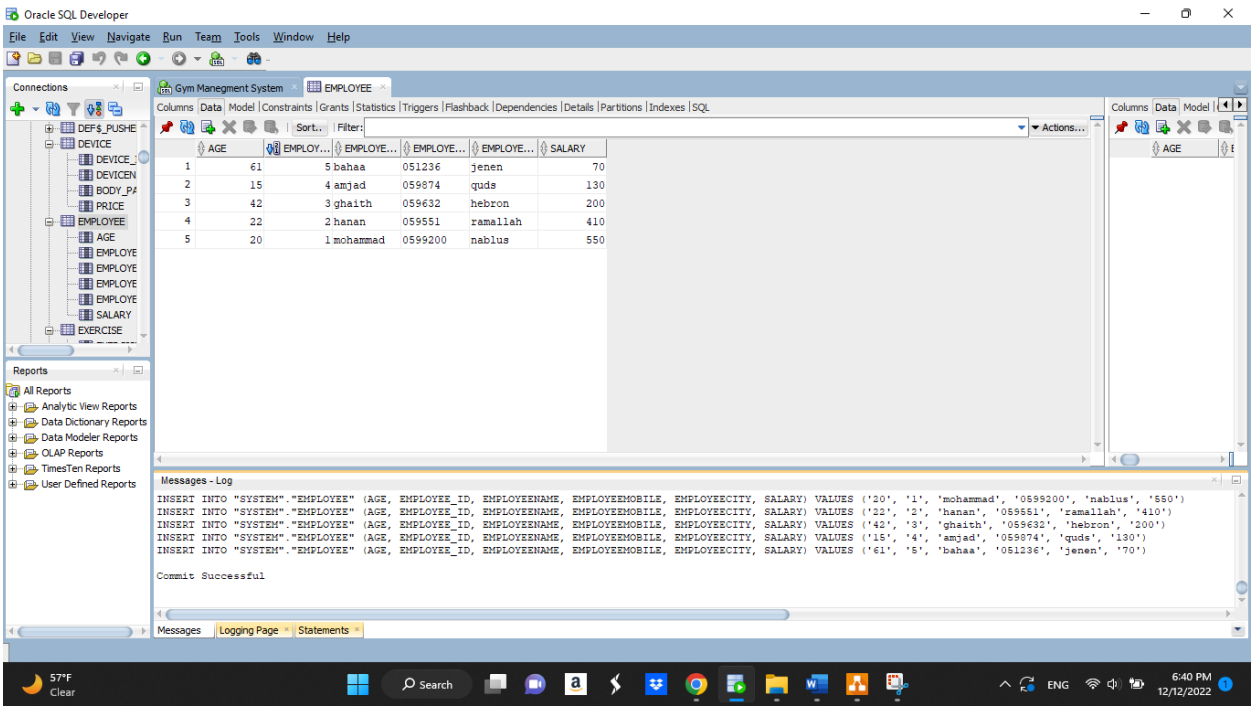
INSERT INTO "SYSTEM"."EMPLOYEE" (AGE, EMPLOYEE_ID, EMPLOYEENAME, EMPLOYEEMOBILE, EMPLOYEECITY, SALARY) VALUES ('20', '1', 'mohammad', '0599200', 'nablus', '550')

INSERT INTO "SYSTEM"."EMPLOYEE" (AGE, EMPLOYEE_ID, EMPLOYEENAME, EMPLOYEEMOBILE, EMPLOYEECITY, SALARY) VALUES ('22', '2', 'hanan', '059551', 'ramallah', '410')

INSERT INTO "SYSTEM"."EMPLOYEE" (AGE, EMPLOYEE_ID, EMPLOYEENAME, EMPLOYEEMOBILE, EMPLOYEECITY, SALARY) VALUES ('42', '3', 'ghaith', '059632', 'hebron', '200')

INSERT INTO "SYSTEM"."EMPLOYEE" (AGE, EMPLOYEE_ID, EMPLOYEENAME, EMPLOYEEMOBILE, EMPLOYEECITY, SALARY) VALUES ('15', '4', 'amjad', '059874', 'quds', '130')

INSERT INTO "SYSTEM"."EMPLOYEE" (AGE, EMPLOYEE_ID, EMPLOYEENAME, EMPLOYEEMOBILE, EMPLOYEECITY, SALARY) VALUES ('61', '5', 'bahaa', '051236', 'jenen', '70')



Insert into table device

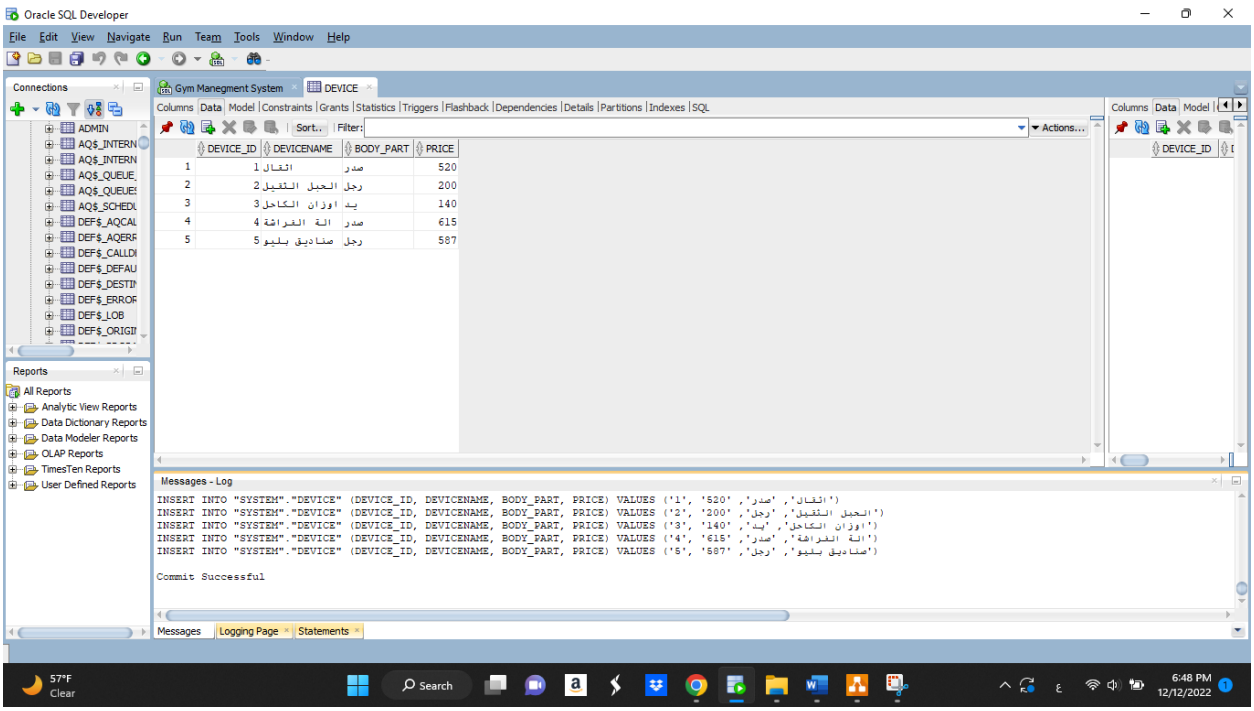
INSERT INTO "SYSTEM"."DEVICE" (DEVICE_ID, DEVICENAME, BODY_PART, PRICE) VALUES('520' ,'صدر' ,'اثقال' ,'1')

INSERT INTO "SYSTEM"."DEVICE" (DEVICE_ID, DEVICENAME, BODY_PART, PRICE) VALUES('200' ,'رجل' ,'الحبل الثقيل' ,'2')

INSERT INTO "SYSTEM"."DEVICE" (DEVICE_ID, DEVICENAME, BODY_PART, PRICE) VALUES('140' ,'يد' ,'اوزان الكاحل' ,'3')

INSERT INTO "SYSTEM"."DEVICE" (DEVICE_ID, DEVICENAME, BODY_PART, PRICE) VALUES('615' ,'صدر' ,'الة الفراشة' ,'4')

INSERT INTO "SYSTEM"."DEVICE" (DEVICE_ID, DEVICENAME, BODY_PART, PRICE) VALUES('587' ,'رجل' ,'صناديق بليو' ,'5')



Insert into table admin

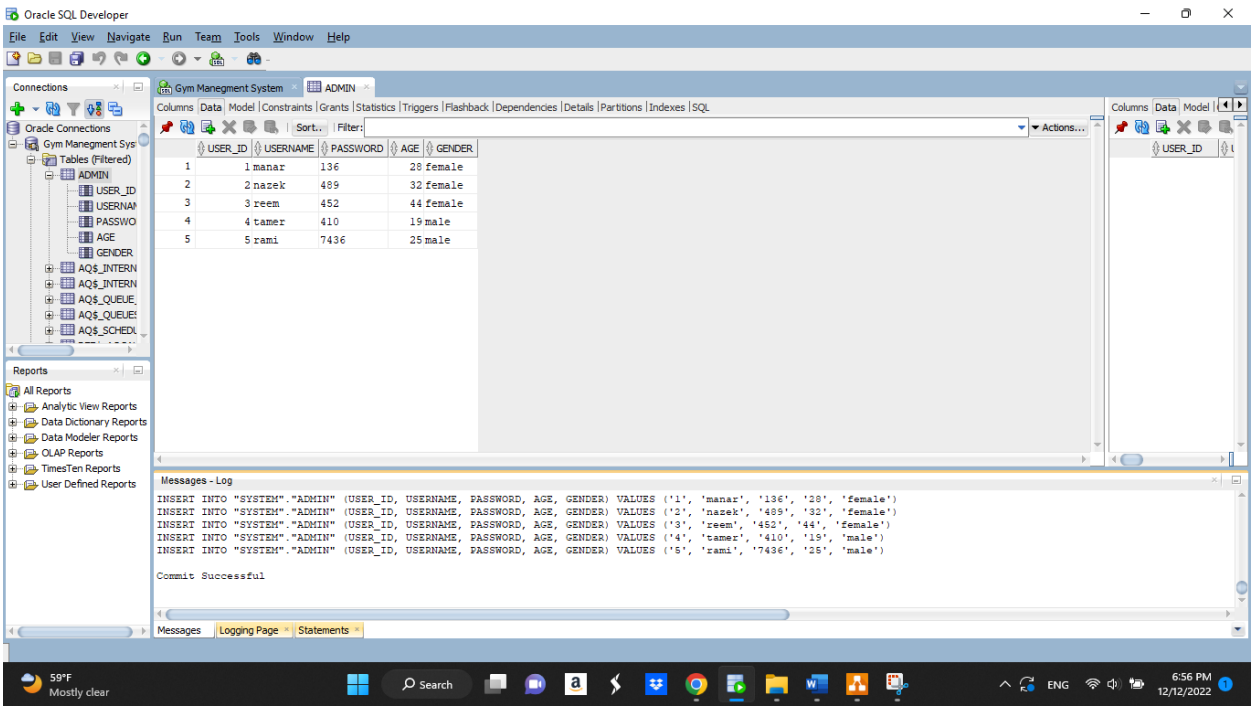
INSERT INTO "SYSTEM"."ADMIN" (USER_ID, USERNAME, PASSWORD, AGE, GENDER) VALUES ('1', 'manar', '136', '28', 'female')

INSERT INTO "SYSTEM"."ADMIN" (USER_ID, USERNAME, PASSWORD, AGE, GENDER) VALUES ('2', 'nazek', '489', '32', 'female')

INSERT INTO "SYSTEM"."ADMIN" (USER_ID, USERNAME, PASSWORD, AGE, GENDER) VALUES ('3', 'reem', '452', '44', 'female')

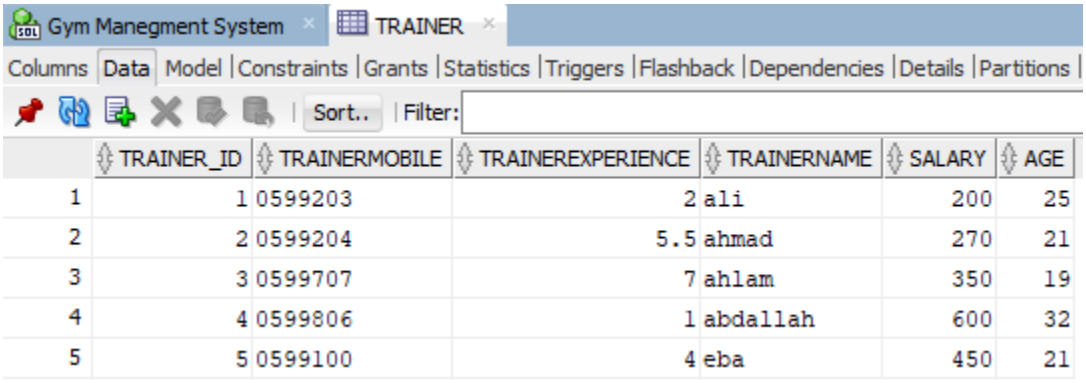
INSERT INTO "SYSTEM"."ADMIN" (USER_ID, USERNAME, PASSWORD, AGE, GENDER) VALUES ('4', 'tamer', '410', '19', 'male')

INSERT INTO "SYSTEM"."ADMIN" (USER_ID, USERNAME, PASSWORD, AGE, GENDER) VALUES ('5', 'rami', '7436', '25', 'male')



Functional Dependency

*First We Will Check Functional Dependency For trainer table according to Data:



	TRAINER_ID	TRAINERMOBILE	TRAINEREXPERIENCE	TRAINERNAME	SALARY	AGE
1	1	0599203	2	ali	200	25
2	2	0599204	5.5	ahmad	270	21
3	3	0599707	7	ahlam	350	19
4	4	0599806	1	abdallah	600	32
5	5	0599100	4	eba	450	21

Check FD for trainer id and trainer mobile: → (Have a FD)

Check FD for trainer id and trainer exercise: → (Have a FD)

Check FD for trainer id and trainer name: → (Have a FD)

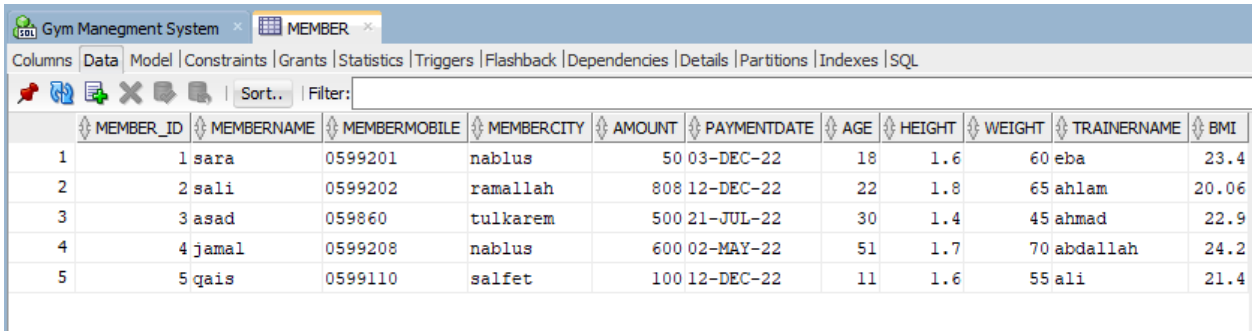
Check FD for trainer id and salary : → (Have a FD)

Check FD for trainer id and age: → (don't have a FD)

Functional Dependency for trainer table :

1) for trainer id → trainer mobile
2) trainer id → trainer exercise
3) trainer id → trainer name
4) trainer id → salary
5) trainer name→ trainer mobile

* We Will Check Functional Dependency For member table according to Data:



	MEMBER_ID	MEMBERNAME	MEMBERMOBILE	MEMBERCITY	AMOUNT	PAYMENTDATE	AGE	HEIGHT	WEIGHT	TRAINERNAME	BMI
1	1	sara	0599201	nablus	50	03-DEC-22	18	1.6	60	eba	23.4
2	2	sali	0599202	ramallah	808	12-DEC-22	22	1.8	65	ahlam	20.06
3	3	asad	059860	tulkarem	500	21-JUL-22	30	1.4	45	ahmad	22.9
4	4	jamal	0599208	nablus	600	02-MAY-22	51	1.7	70	abdallah	24.2
5	5	qais	0599110	salfet	100	12-DEC-22	11	1.6	55	ali	21.4

Check FD for member id and member mobile: → (Have a FD)

Check FD for member id and member city: → (don't Have a FD)

Check FD for member id and member name: → (Have a FD)

Check FD for member id and amount : → (Have a FD)

Check FD for member id and age: → (have a FD)

Check FD for member id and paymentDate: → (have a FD)

Check FD for member id and weight: → (have a FD)

Check FD for member id and height: → (don't have a FD)

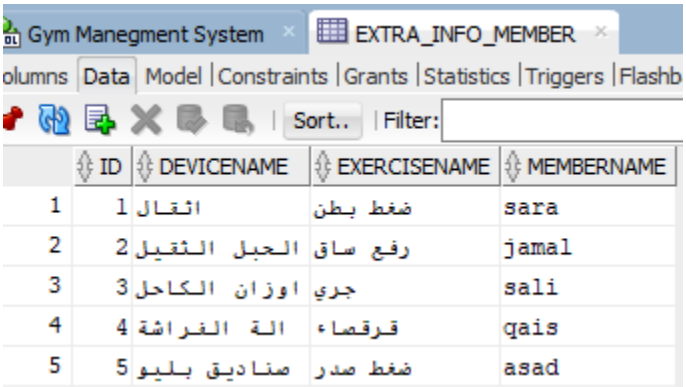
Check FD for member id and trainer name: → (have a FD)

Check FD for member id and bmi: → (have a FD)

Functional Dependency for trainer table :

1) for member id → member mobile
2) member id → member name
3) member id → amount
4) member id → age
5) member id → paymentDate
6) member id → weight
7) member id → trainer name
8) member id → bmi

*We Will Check Functional Dependency For extra info member table according to Data:



	ID	DEVICENAME	EXERCISENAME	MEMBERNAME
1	1	اثقال	ضغط بطن	sara
2	2	الحبل الثقيل	رفع ساق	jamal
3	3	اوزان الكاحل	جري	sali
4	4	الة الفراشة	قرقصاء	qais
5	5	صناديق بليو	ضغط صدر	asad

Check FD for id and device name: → (Have a FD)

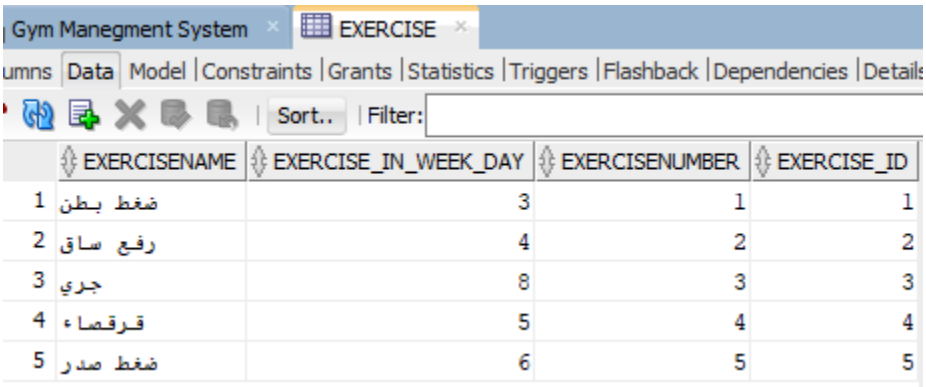
Check FD for id and exercise name: → (Have a FD)

Check FD for member id and member name: → (Have a FD)

Functional Dependency for extra info member table :

1) for id → device name
2) id → exercise name
3) id → member name
4) member name → exercise name

*We Will Check Functional Dependency For exercise table according to Data:



	EXERCISENAME	EXERCISE_IN_WEEK_DAY	EXERCISENUMBER	EXERCISE_ID
1	ضغط بطن	3	1	1
2	رفع ساق	4	2	2
3	جري	8	3	3
4	قرفصاء	5	4	4
5	ضغط صدر	6	5	5

Check FD for exercise id and exercise in week day: → (Have a FD)

Check FD for exercise id and exercise name: → (Have a FD)

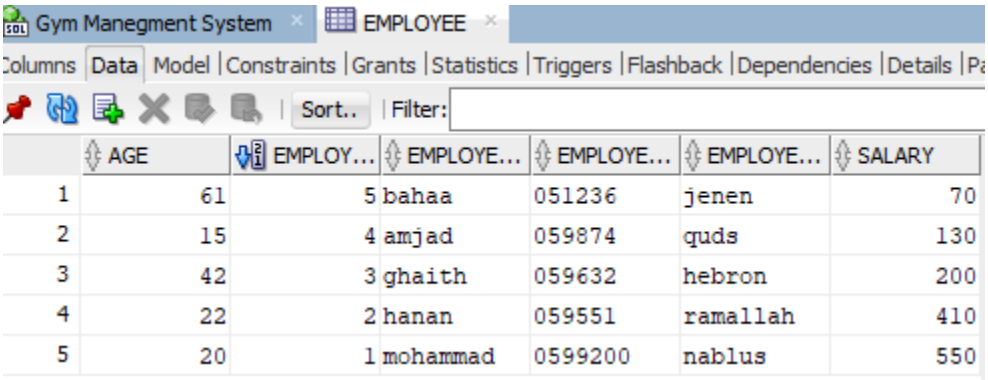
Check FD for exercise id and exercise number: → (Have a FD)

Functional Dependency for extra info member table :

1) for exercise id → exercise in week day
2) exercise id → exercise name
3) exercise id → exercise number

4) exercise in week day→ exercise name

*We Will Check Functional Dependency For employee table according to Data:



	AGE	EMPLOY...	EMPLOYE...	EMPLOYE...	EMPLOYE...	SALARY
1	61	5	bahaa	051236	jenen	70
2	15	4	amjad	059874	quds	130
3	42	3	ghaith	059632	hebron	200
4	22	2	hanan	059551	ramallah	410
5	20	1	mohammad	0599200	nablus	550

Check FD for employee id and age: → (Have a FD)

Check FD for employee id and e employee name: → (Have a FD)

Check FD for employee id and employee mobile: → (Have a FD)

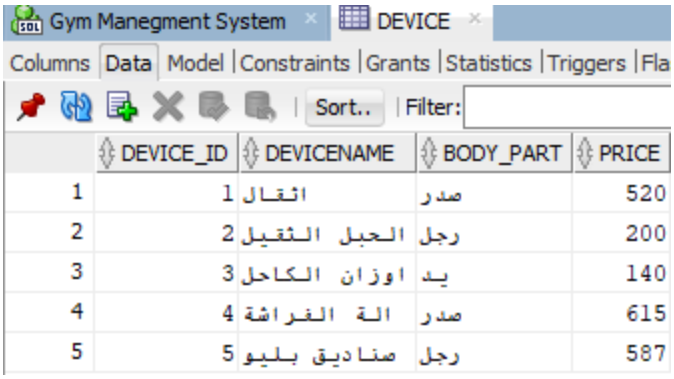
Check FD for employee id and city: → (Have a FD)

Check FD for employee id and salary: → (Have a FD)

Functional Dependency for extra info member table :

1) for employee id → age
2) employee id → employee mobile
3) employee id → employee name
4) employee id → city
5) employee id → salary

*We Will Check Functional Dependency For device table according to Data:



	DEVICE_ID	DEVICENAME	BODY_PART	PRICE
1	1	اثقال	صدر	520
2	2	الحبل الثقيل	رجل	200
3	3	اوزان الكاحل	يد	140
4	4	الة الفراشة	صدر	615
5	5	صناديق بليو	رجل	587

Check FD for device id and device name: → (Have a FD)

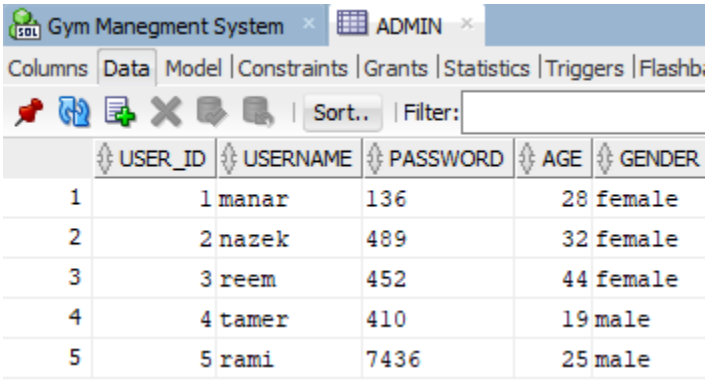
Check FD for device id and body part: → (don't Have a FD)

Check FD for device id and price: → (Have a FD)

Functional Dependency for extra info member table :

1) for device id → device name
2) device id → price
3) device name → price

*We Will Check Functional Dependency For admin table according to Data:



	USER_ID	USERNAME	PASSWORD	AGE	GENDER
1	1	manar	136	28	female
2	2	nazek	489	32	female
3	3	reem	452	44	female
4	4	tamer	410	19	male
5	5	rami	7436	25	male

Check FD for user id and user name: → (Have a FD)

Check FD for user id and password: → (Have a FD)

Check FD for user id and age: → (Have a FD)

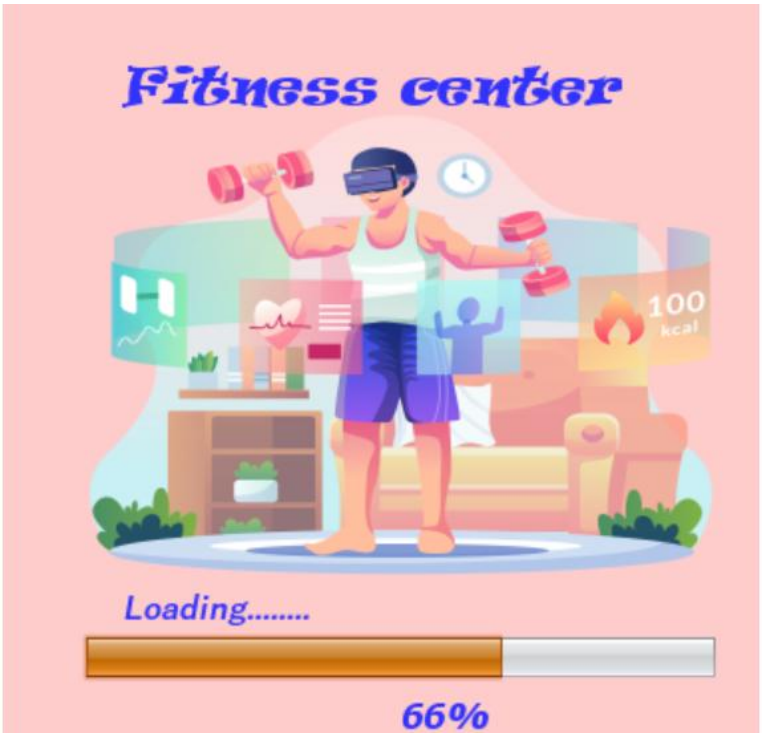
Check FD for user id and gender: → (don't Have a FD)

Functional Dependency for admin table :

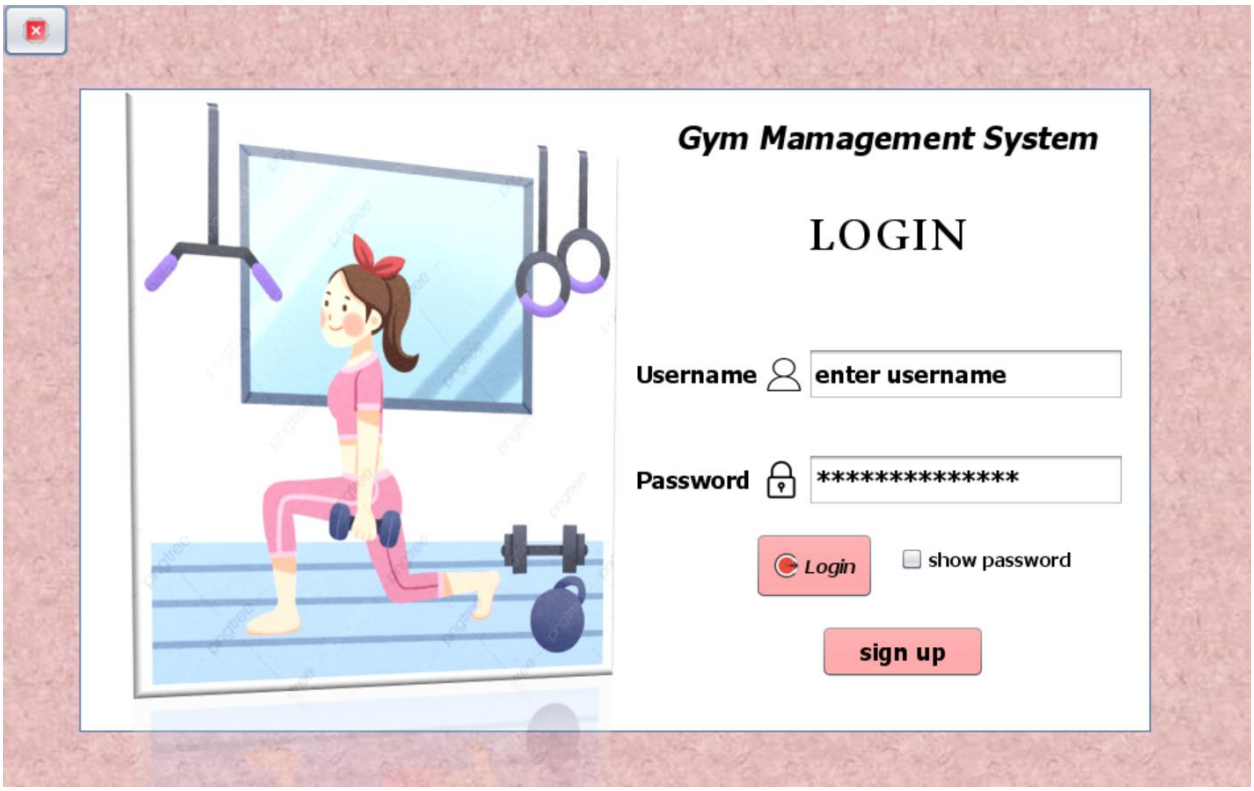
1) for user id → user name
2) user id → age
3) user name → password
4) age → password
5) user id → gender

discussion about project GUI work, what each interface do

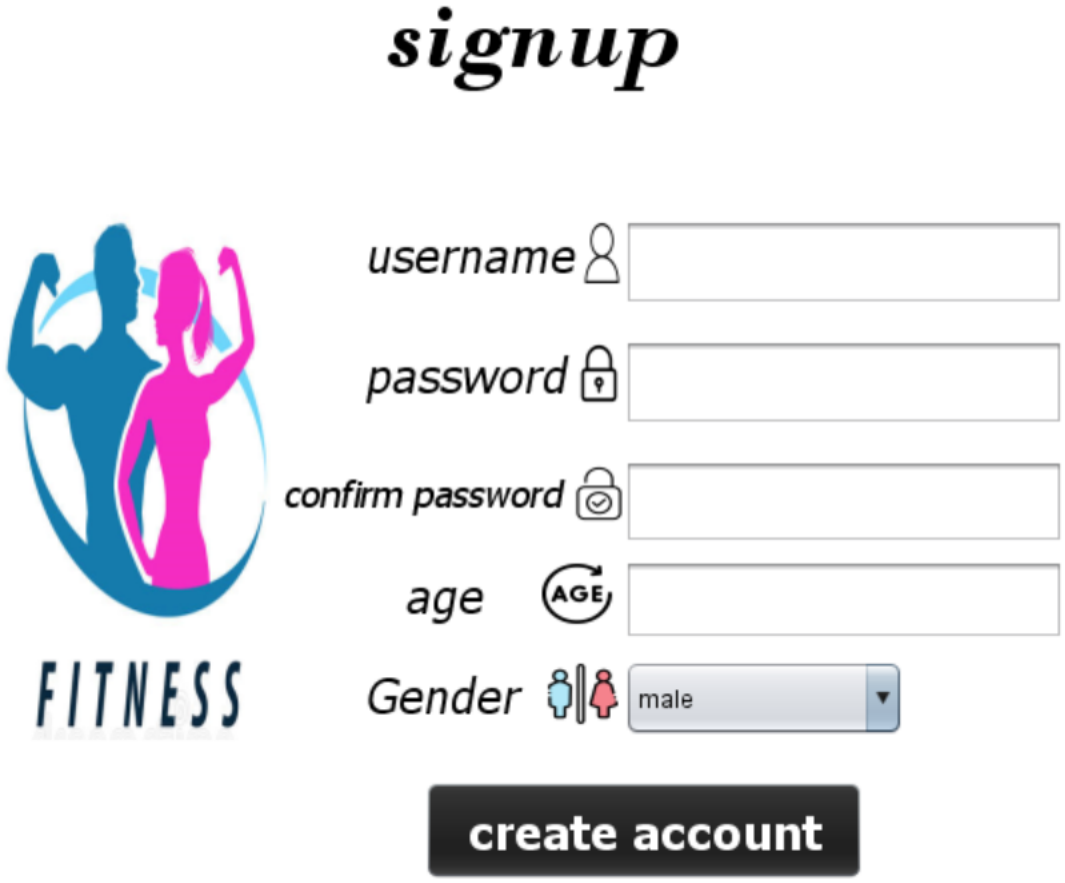
welcome window : it is the first window that appears when the program is running. It contains a progress bar, which is a counter that increases automatically when the counter value becomes 100. the window closes and the next window opens, which is the login window.



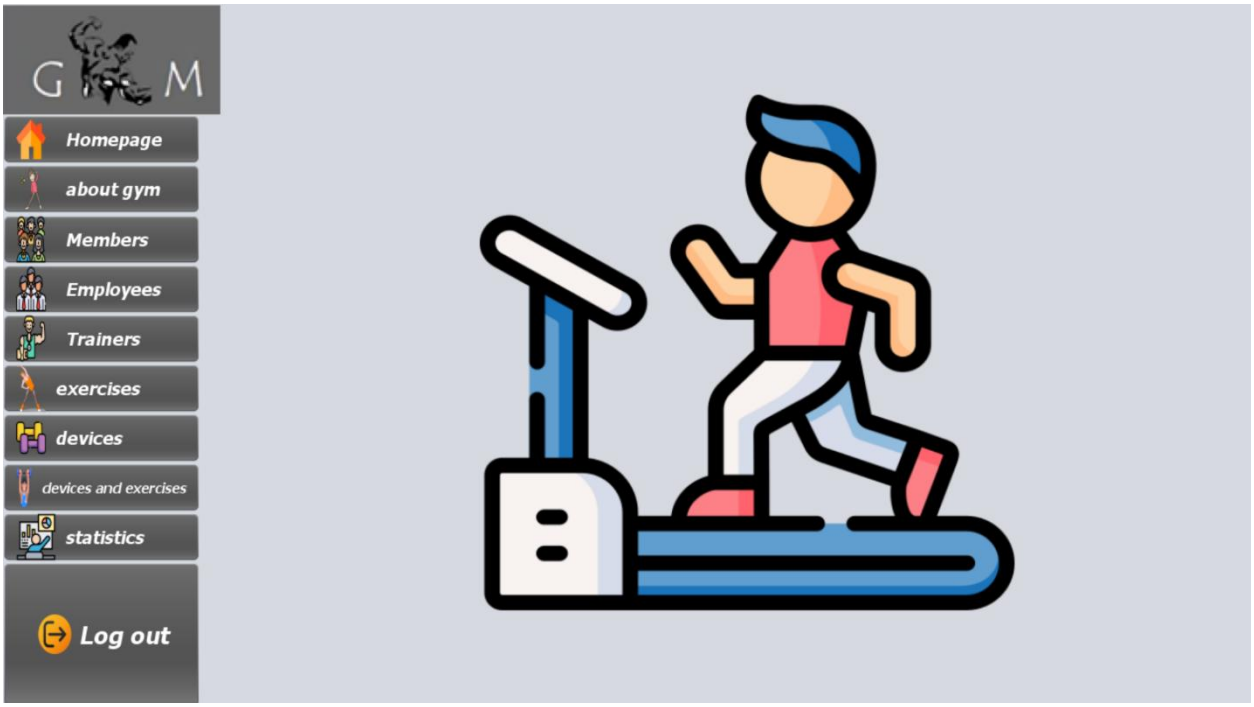
In order to login you have to write the user name and password correctly, when you touch on show password your password will appear, when you touch on button sign up you will go to the next page.



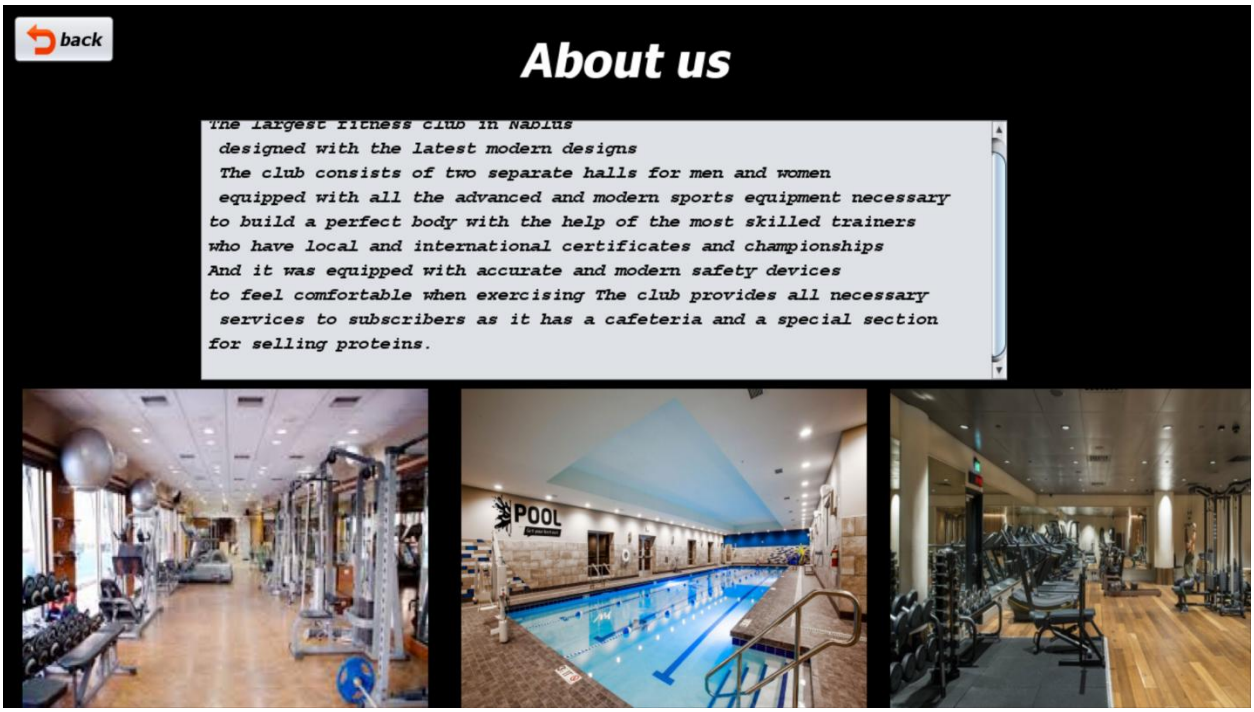
Here you have to write the correct username, password, confirm password, age and choose your gender then press on button create account in order to create an account, if you write your password wrongly while confirming your password you will have a note that your password is incorrect.



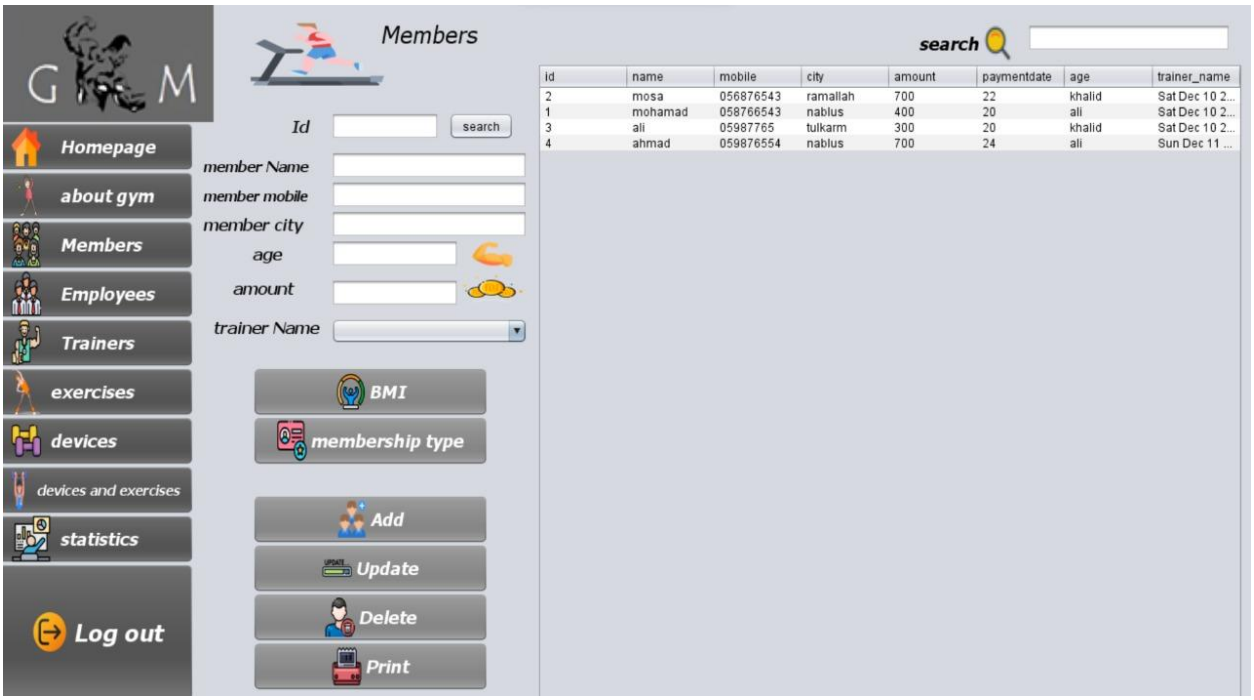
This is the homepage each choice explain much about that choice for example if you press (about gym) it will explain a lot about our gym and shows pictures about the gym, if you press (members) you will be sent to the page of members to know much about them, if you press (employees) you will be sent to the page of employee to know much about them, if you press (trainers) you will be sent to the page of trainers to know much about them, if you press (exercises) you will be sent to the page of exercises to know much about the exercises there, if you press (devices) you will be sent to the page of devices to know much about the devices in our gym, if you press (devices and exercises) you will be sent to the page of devices to know much about the devices and exercises in our gym, if you press (statistics) you will be sent to the page of statistics to know much about statistics in our gym, , if you press (log out) you will be out.



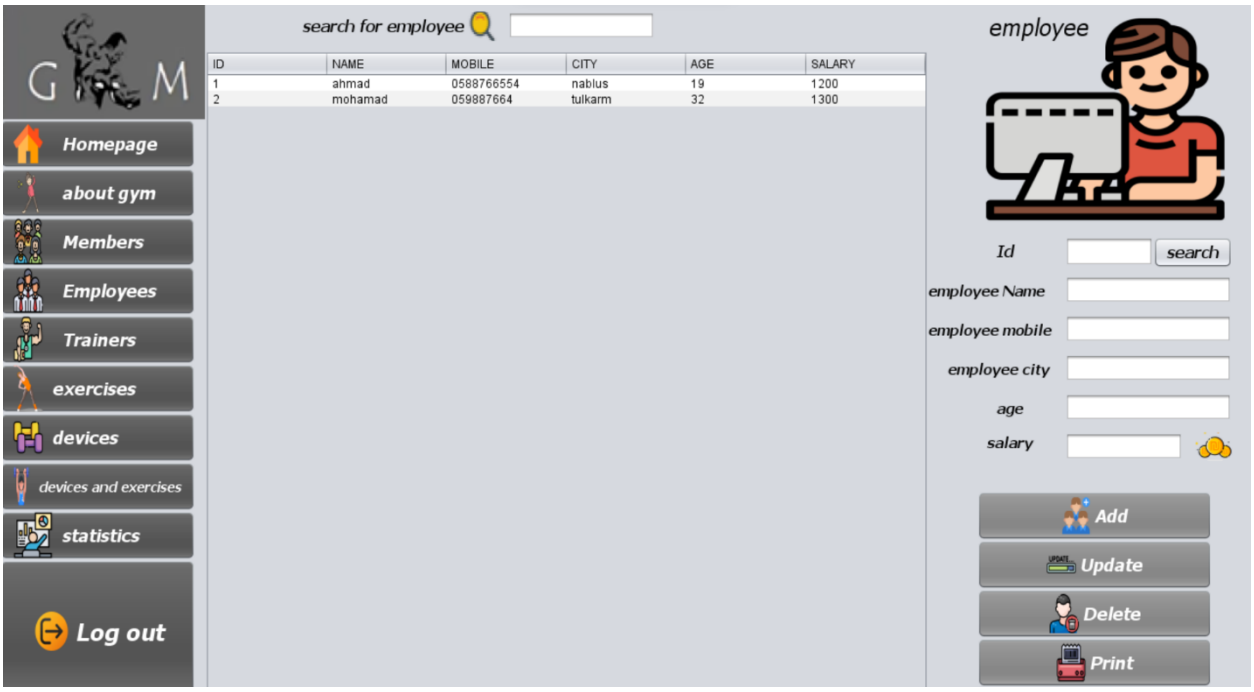
These picture and the information written introduces our gym to all.



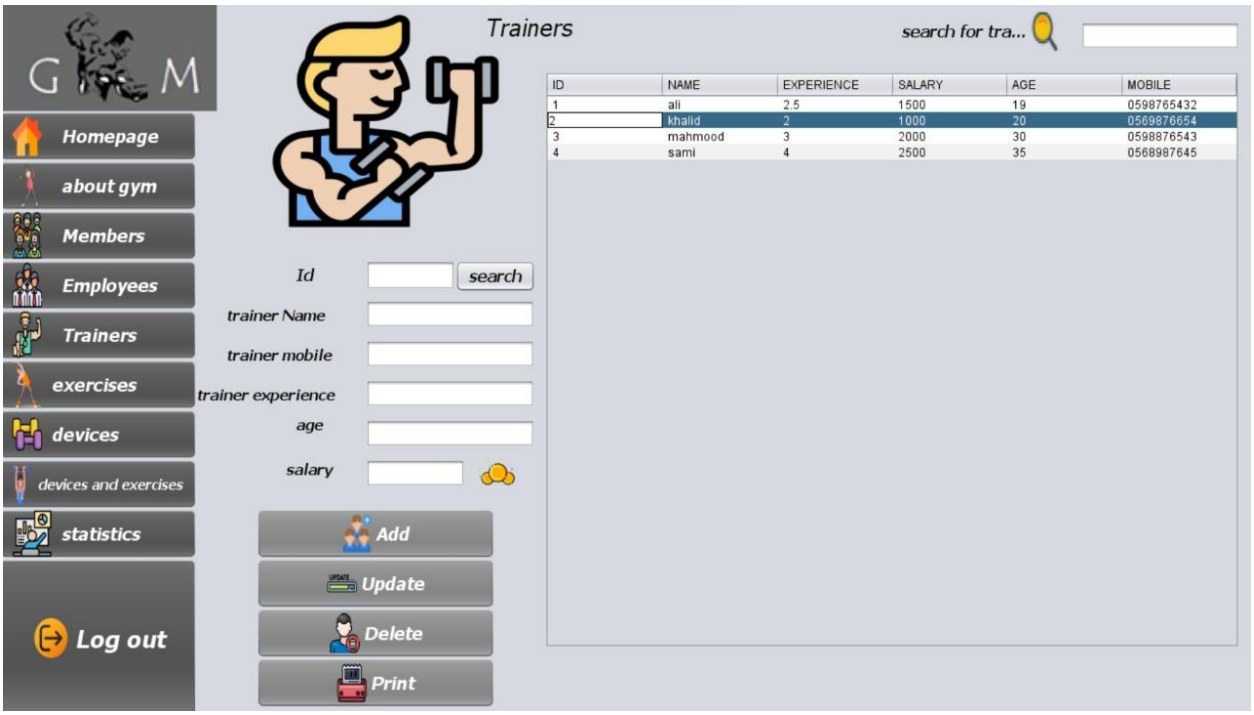
The member can write the id then press on (search) the information down will appear automatically or the member can write his/her name, his/her mobile, his/her city, his/her age, his/her amount and he/she choose the trainer name from list then if press on button (add) all information will be added to the table, if he/she update any information, the information will be updated on table, if he/she delete any information then all information about him/her will be deleted from table, if he/she press on (print) the information about him/her will be printed, if member press on (BMI) the BMI will be calculated, if member press on (membership type) it explain the kind of Gym membership.



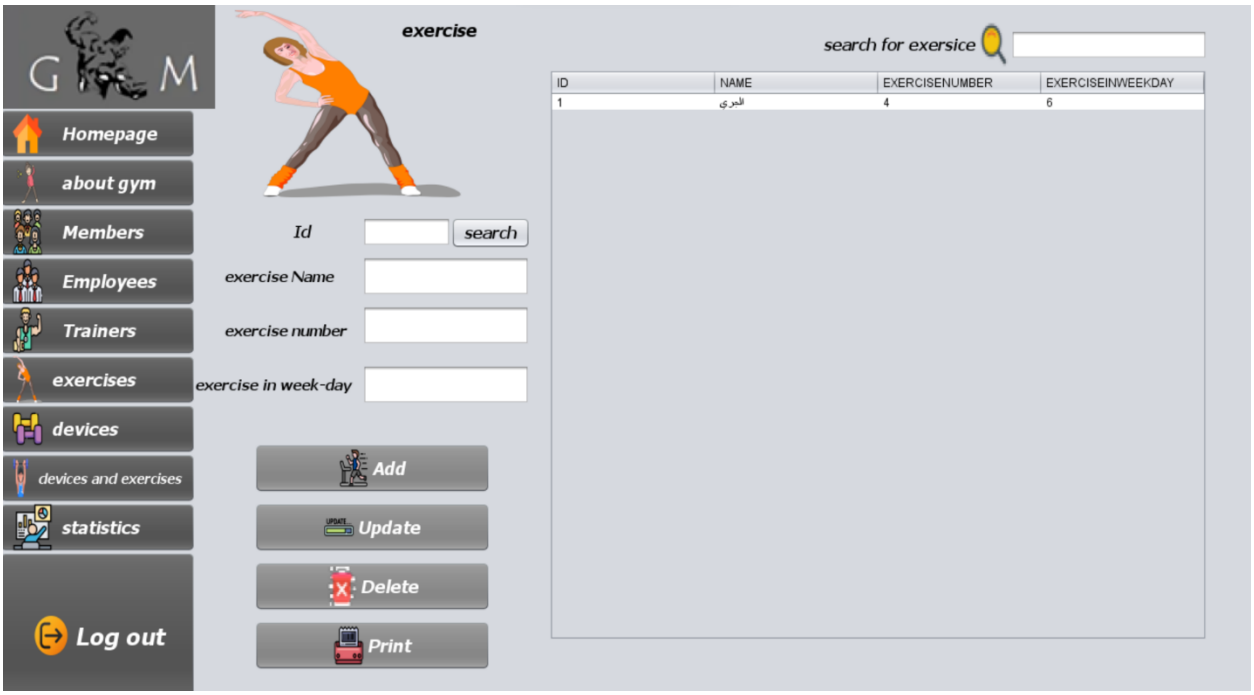
The employee can write the id then press on (search) the information down will appear automatically or the employee can write his/her name, his/her mobile, his/her city, his/her age, his/her salary then if press on button (add) all information will be added to the table, if he/she update any information, the information will be updated on table, if he/she delete any information then all information about him/her will be deleted from table, if he/she press on (print) the information about him/her will be printed.



The trainer can write the id then press on (search) the information down will appear automatically or the trainer can write his/her name, his/her mobile, his/her city, his/her employee experience, his/her age, his/her salary then if press on button (add) all information will be added to the table, if he/she update any information, the information will be updated on table, if he/she delete any information then all information about him/her will be deleted from table, if he/she press on (print) the information about him/her will be printed.

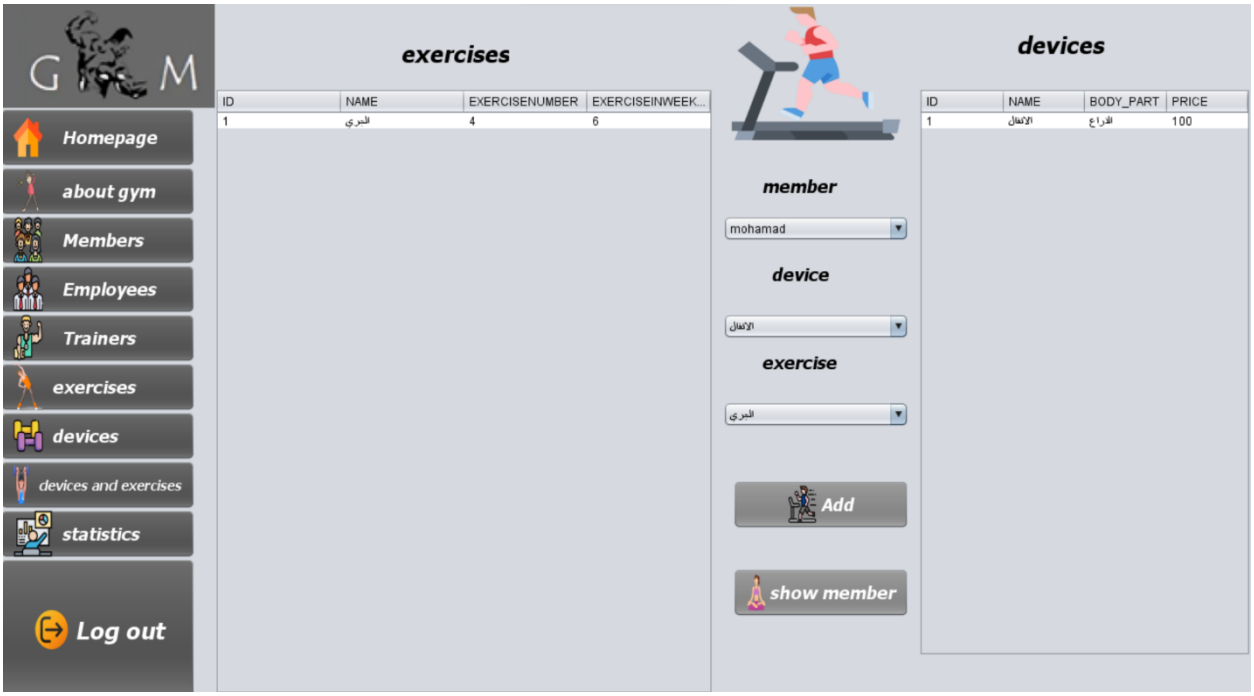


The admin can write the id then press on (search) the information down will appear automatically or the admin can write his/her exercise name, his/her exercise number, his/her exercise in week day then if press on button (add) all information will be added to the table, if he/she update any information, the information will be updated on table, if he/she delete any information then all information about him/her will be deleted from table, if he/she press on (print) the information about him/her will be printed.

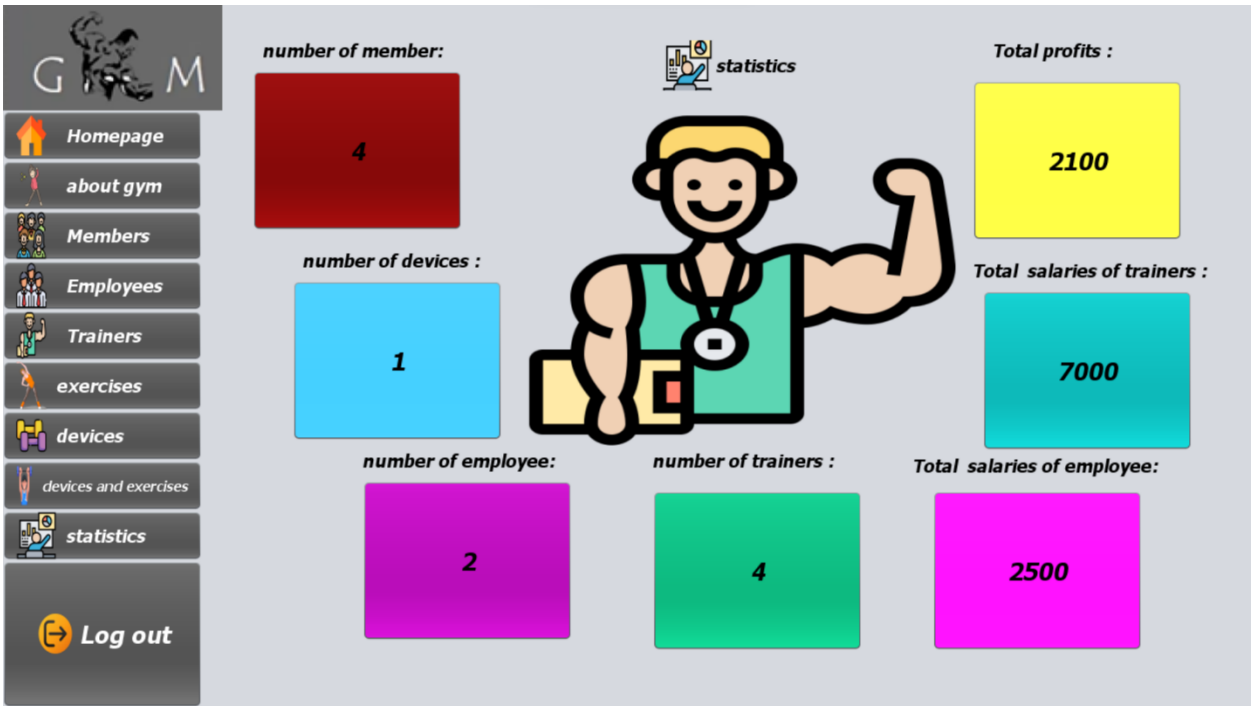


The admin can write the id then press on (search) the information down will appear automatically or the admin can write his/her device name, his/her body part, his/her price then if press on button (add) all information will be added to the table, if he/she update any information, the information will be updated on table, if he/she delete any information then all information about him/her will be deleted from table, if he/she press on (print) the information about him/her will be printed.

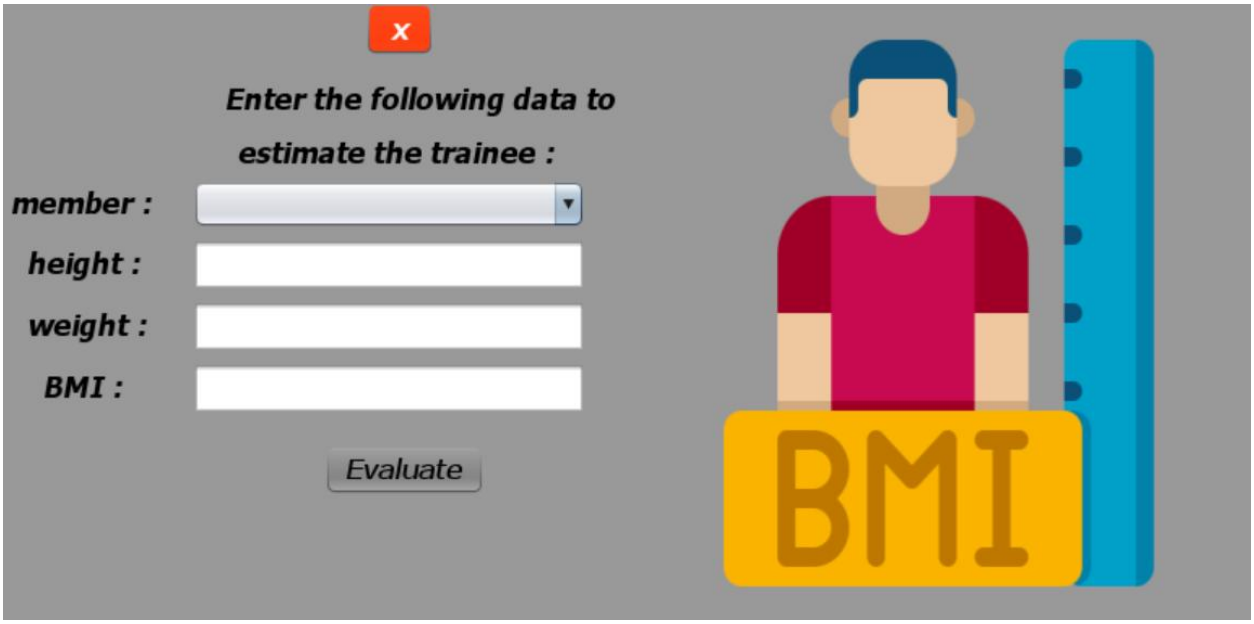
The member chooses the device and exercise he/she prefer, when he/she press on button (add) they will be added, when the admin press on (show member) the exercise, device and trainer chosen by member will appear to the admin.



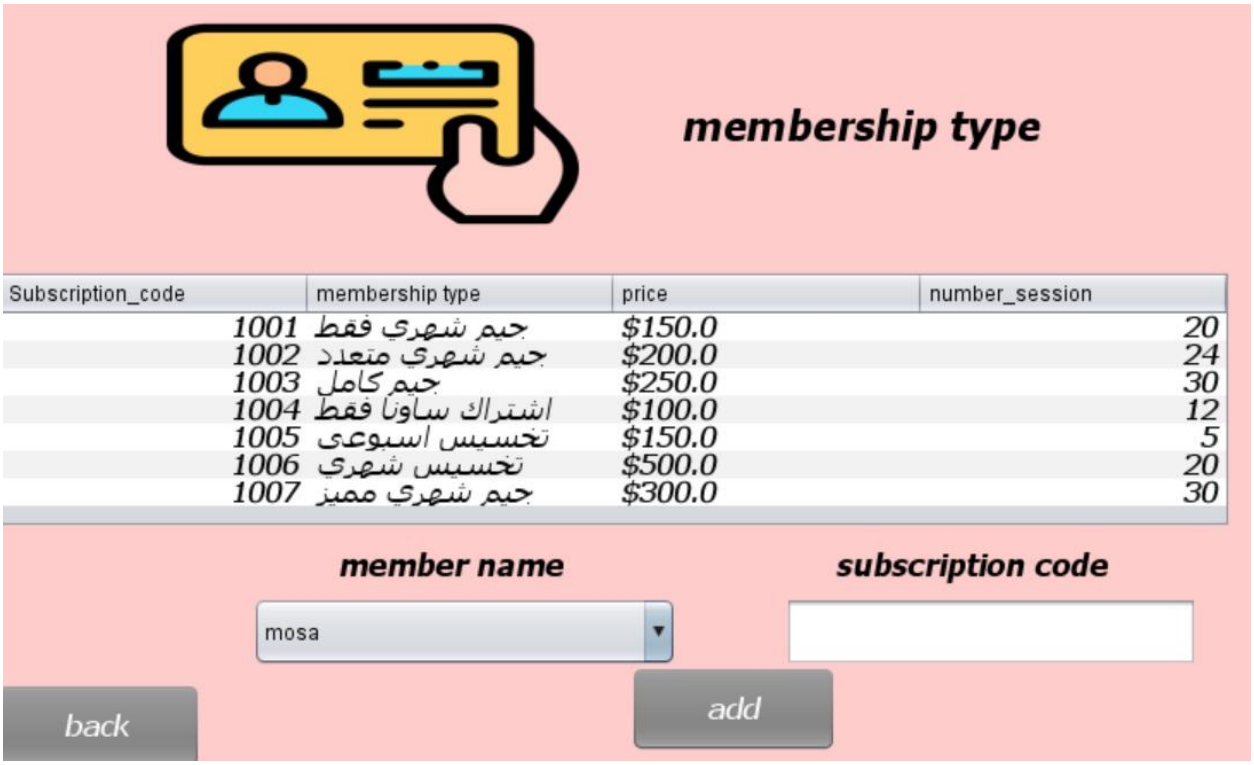
This page explain much about statistics of the gym, for example number of member, total profits, number of devices, total salaries of trainers, number of employee, number of trainers, total salaries of employee.



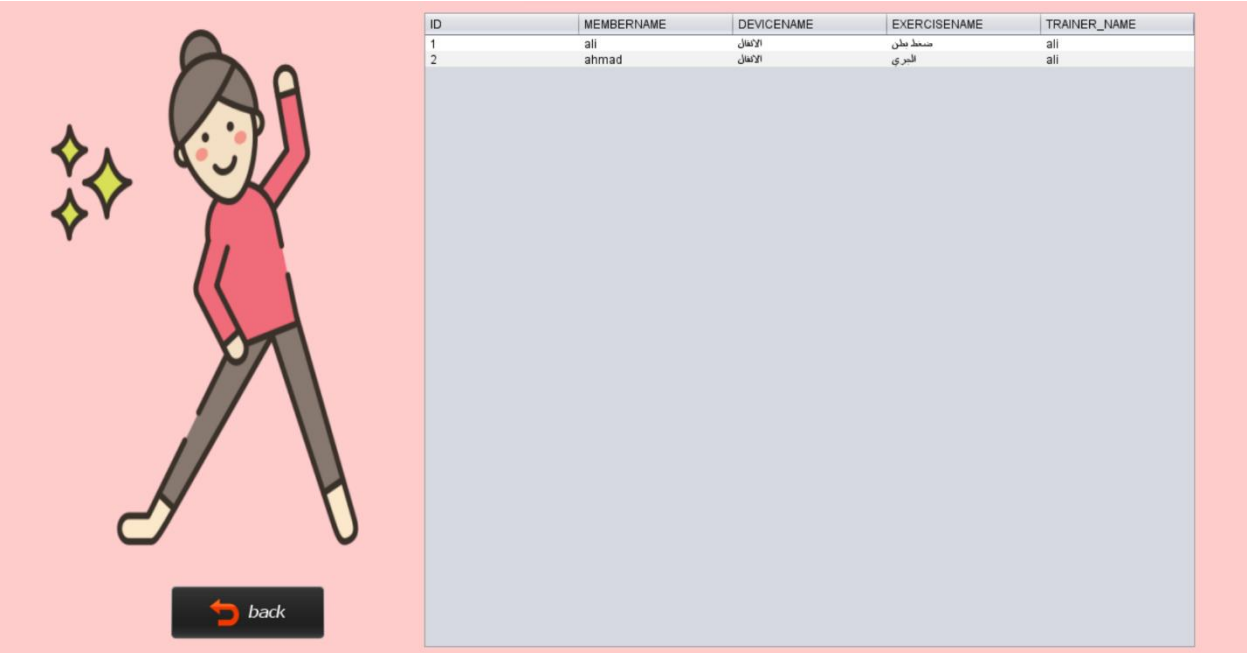
Here we choose the name of member, then write his/her height and weight, after that we press on (evaluate) to get the BMI, then we get a note about the BMI, if its normal or abnormal.



The member choose the kind of membership he/she wants, the price, number of session, when he/she press on mouse exactly on (membership) will get the subscription code in the text, then press on (add), finally when press on (back) return to page (member).



When we pressed on (show member) we will get this page which show member name, device name, exercise name, trainer name, when we press on (back) we return to page (member)



Jasper report

devices

ID	NAME	PRICE	BODY_PART
1	الاثقال	25	الذراع
5	bench press	70	صدر وذراع
2	Dip station	30	صدر
3	Tricep bars	24	كف
4	pull-up-par	39	صدر وذراع
6	الحبال القتالية	60	كتف
7	preacher bench	15	كتف
8	الة الفراشة	33	ساق
9	مقبض التمارين	55	يد
10	مقعد الاوزان المائل	37	صدر
11	جهاز الجذب العالي	80	ساق
12	الة تمرين سمانة القدم	24	كعب القدم
13	صناديق بليو	67	ذراع
14	اوزان الكاحل	200	كاحل
15	الة القرفصاء	300	فخذ وساق
16	جهاز التمرين العكسي	400	ظهر

employee

ID	NAME	CITY	SALARY	AGE	MOBILE
3	ameed	khalil	1600	28	0598765544
4	masa	nablus	1500	20	0567877658
1	ahmad	nablus	1200	19	0588766554
2	mohamad	tulkarm	1300	32	0598766553
5	ali	ramallah	1300	30	0598765442

exercises

EXERCISENUMBER	NAME	ID	EXERCISEINWEEKD
7	الجري	1	15
6	ضغط بطن	2	12
6	ضغط صدر	3	12
5	قرفصاء	4	12
6	رفع ساق	5	12
7	الجري	6	15
4	بنج ذراع	7	8

members

ID	MOBILE	NAME	AMOUNT	HEIGHT	TRAINERNA	MEMBERCI	BMI	WEIGHT	DATEPAY	AGE
2	056876543	mosa	700	1.66	khalid	ramallah	19.959356	55.0	Sat Dec 10 23:33:20 IST 2022	23
1	058766543	mohamad	400	1.4	ali	nablus	20.408165	40.0	Sat Dec 10 23:32:41 IST 2022	20
3	05987765	ali	300	1.77	khalid	tulkarm	19.151585	60.0	Sat Dec 10 23:34:44 IST 2022	20
4	059876554	ahmad	700	1.75	ali	nablus	20.897959	64.0	Sun Dec 11 12:54:32 IST 2022	24

trainers

NAME	EXPERIENCE	MOBILE	AGE	SALARY	ID
ali	2.5	0598765432	19	1500	1
khalid	2.0	0569876654	20	1000	2
mahmood	3.0	0598876543	30	2000	3
sami	4.0	0568987645	35	2500	4
eba	3.0	0599875365	16	900	5