**[CREAZIONE TABLE]**

CREATE DATABASE progettoSSD;

USE progettoSSD;

CREATE TABLE Utente (

ID int(10) NOT NULL AUTO\_INCREMENT,

Nome varchar(255),

Cognome varchar(255),

PRIMARY KEY (ID)

);

CREATE TABLE Prodotto (

ID int(10) NOT NULL AUTO\_INCREMENT,

ListaDesideriID int(10) NOT NULL,

Nome varchar(255),

PRIMARY KEY (ID)

);

CREATE TABLE ListaDesideri (

ID int(10) NOT NULL AUTO\_INCREMENT,

UtenteID int(10) NOT NULL,

Nome varchar(255),

PRIMARY KEY (ID)

);

CREATE TABLE Gruppo (

ID int(10) NOT NULL AUTO\_INCREMENT,

Nome varchar(255),

PRIMARY KEY (ID)

);

CREATE TABLE Partecipazione (

ID int(10) NOT NULL AUTO\_INCREMENT,

UtenteProprietario tinyint(1) NOT NULL,

UtenteID int(10) NOT NULL,

GruppoID int(10) NOT NULL,

PRIMARY KEY (ID)

);

CREATE TABLE Gruppo\_IDListeCondivise (

GruppoID int(10) NOT NULL,

GruppoIndex int(10) NOT NULL,

IDListeCondivise int(10) NOT NULL,

PRIMARY KEY (GruppoID, GruppoIndex)

);

ALTER TABLE Prodotto ADD CONSTRAINT FKProdotto471876 FOREIGN KEY (ListaDesideriID) REFERENCES ListaDesideri (ID);

ALTER TABLE ListaDesideri ADD CONSTRAINT FKListaDesid479326 FOREIGN KEY (UtenteID) REFERENCES Utente (ID);

ALTER TABLE Partecipazione

ADD CONSTRAINT FKPartecipaz792957 FOREIGN KEY (UtenteID) REFERENCES Utente (ID);

ALTER TABLE Partecipazione ADD CONSTRAINT FKPartecipaz714368 FOREIGN KEY (GruppoID) REFERENCES Gruppo (ID);

ALTER TABLE Gruppo\_IDListeCondivise ADD CONSTRAINT FKGruppo\_IDL170216 FOREIGN KEY (GruppoID) REFERENCES Gruppo (ID);

**SOSTITUZIONE RAPIDA**

CREATE database progettoSSD;

USE progettoSSD;

CREATE TABLE Utente (ID int(10) NOT NULL AUTO\_INCREMENT, Nome varchar(255), Cognome varchar(255), PRIMARY KEY (ID));

CREATE TABLE Prodotto (ID int(10) NOT NULL AUTO\_INCREMENT, ListaDesideriID int(10) NOT NULL, Nome varchar(255), PRIMARY KEY (ID));

CREATE TABLE ListaDesideri (ID int(10) NOT NULL AUTO\_INCREMENT, UtenteID int(10) NOT NULL, Nome varchar(255), PRIMARY KEY (ID));

CREATE TABLE Gruppo (ID int(10) NOT NULL AUTO\_INCREMENT, Nome varchar(255), PRIMARY KEY (ID));

CREATE TABLE Partecipazione (ID int(10) NOT NULL AUTO\_INCREMENT, UtenteProprietario tinyint(1) NOT NULL, UtenteID int(10) NOT NULL, GruppoID int(10) NOT NULL, PRIMARY KEY (ID));

CREATE TABLE Gruppo\_ListaDesideri (GruppoID int(10) NOT NULL, ListaDesideriID int(10) NOT NULL, PRIMARY KEY (GruppoID, ListaDesideriID));

ALTER TABLE Prodotto ADD CONSTRAINT FKProdotto471876 FOREIGN KEY (ListaDesideriID) REFERENCES ListaDesideri (ID);

ALTER TABLE ListaDesideri ADD CONSTRAINT FKListaDesid479326 FOREIGN KEY (UtenteID) REFERENCES Utente (ID);

ALTER TABLE Partecipazione ADD CONSTRAINT FKPartecipaz792957 FOREIGN KEY (UtenteID) REFERENCES Utente (ID);

ALTER TABLE Partecipazione ADD CONSTRAINT FKPartecipaz714368 FOREIGN KEY (GruppoID) REFERENCES Gruppo (ID);

ALTER TABLE Gruppo\_ListaDesideri ADD CONSTRAINT FKGruppo\_Lis633240 FOREIGN KEY (GruppoID) REFERENCES Gruppo (ID);

ALTER TABLE Gruppo\_ListaDesideri ADD CONSTRAINT FKGruppo\_Lis683055 FOREIGN KEY (ListaDesideriID) REFERENCES ListaDesideri (ID);

**### DASHBOARD ###**

**---- RECUPERA ID E NOME LISTE PERSONALI (@ GetPersonalLists)----**

SELECT ID, Nome FROM listadesideri AS lista WHERE lista.UtenteID = "1";

**---- RECUPERA ID E NOME GRUPPI PERSONALI (@ GetPersonalGroups) ----**

SELECT ID, Nome FROM gruppo AS g WHERE EXISTS (SELECT UtenteID, GruppoID FROM partecipazione AS p WHERE g.ID = p.GruppoID AND p.UtenteID = '2');

---- @NewList ----

INSERT INTO listadesideri (UtenteID, Nome) VALUES ('1', 'La Mia Nuova Lista');

---- @NewGroup ----

INSERT INTO gruppo (Nome) VALUES ('Nuovo Gruppo');

SELECT last\_insert\_id() AS id\_nuovo\_gruppo;

* INSERT INTO partecipazione (UtenteProprietario, UtenteID, GruppoID) VALUES ('1','2',id\_nuovo\_gruppo);

**### LISTA ###**

**---- SELEZIONA I PRODOTTI DI UNA LISTA DEI DESIDERI (@ GetList) ----**

SELECT \* FROM prodotto AS pr WHERE EXISTS(SELECT \* FROM listadesideri AS lista WHERE pr.ListaDesideriID = lista.ID AND pr.ListaDesideriID = '1');

* **SELEZIONA NOME LISTA**
* SELECT Nome FROM listadesideri AS lista WHERE lista.ID = '3';

^^^ Così stiamo facendo due query. Se ne può fare anche una sola.

**---- AGGIUNGI PRODOTTO ALLA LISTA(@ AddElement)----**

INSERT INTO prodotto (ListaDesideriID, Nome) VALUES ('1','Mango');

**---- ELIMINAZIONE PRODOTTO DALLA LISTA (@RemoveElement) ----**

DELETE FROM prodotto WHERE ID='5';

---- @RenameList

**---- @RemoveList ----**

DELETE FROM listadesideri AS lista WHERE lista.ID = '3';

**### GRUPPO ###**

**---- SELEZIONA I PARTECIPANTI E LE LISTE CONDIVISE DI UN GRUPPO (@ GetGroup) ----**

SELECT \* FROM utente AS u WHERE EXISTS (SELECT GruppoID FROM partecipazione AS par WHERE u.ID = par.UtenteID AND par.GruppoID = '2');

SELECT \* FROM listadesideri AS lista WHERE EXISTS (SELECT GruppoID FROM gruppo\_listadesideri AS gl WHERE lista.ID = gl.ListaDesideriID AND gl.GruppoID = '2');

* SELECT Nome FROM gruppo AS g WHERE g.ID = '1';

---- @RenameGroup

**---- @RemoveGroup ----**

DELETE FROM gruppo AS g WHERE g.ID = '3';

**----** (@AddPartecipant)