**Курсов проект по Бази данни**

**Тема на проект : Фитнес**

1. Моника Палова - 62323
2. Виктория Лазарова - 62351
3. Костадин Динков - 62330
4. Борислав Карапанов - 62280
5. Кирилка Дончева – 62289

Github repository на проекта: <https://github.com/Borislav-K/fmi-BD-project>

|  |  |
| --- | --- |
| **Задача** | **Фак. №** |
| 1.Създаване на схемите и релациите | 62280 |
| 2. Добавяне на съдържание | 62351 |
| 3. Примерни прости заявки и заявки върху две и повече релации | 62289 |
| 4. Примери с подзаявки | 62323 |
| 5. Примери със съединения | 62330 |
| 6. Примери с групиране и аграгация | 62323 |
| 7. Примери с ограничения | 62280 |
| 8. Примери с изгледи и индекси | 62289 |
| 9. Примери с тригери | 62330 |
| 10. Подготовка на презентацията | 62351 |

I. Описание на предметната област и заданието

Целта на базата, която проектираме е да улесни съхранението и качественото използване на данните в една съвременна фитнес зала. Фокусът е върху управление на наличния персонал, абонаментите на клиентите, както и покупките (тези, извършени от фитнеса, както и клиентските покупки).

II. Дефиниране на схемата на релациите

Базата се състои от 8 таблици, които можем да абстрахираме в 3 групи. Първата група, състояща се от EmployeeTypes и Employees съхраняват информация за персонала във фитнес залата. Втората група таблици за Invoices, Purchases и Products, които съхраняват данните за продуктите във фитнеса, като invoces са покупките, извършени от фитнеса, а purchases са клиентски покупки. Третата група таблици – subscriptions и subscriptiontypes представляват данни за клиентските абонаменти. Таблицата clients съхранява информация за клиентите на фитнеса и е свързана с покупките и абонаментите на всеки клиент.

Ето и графична схема на таблиците:

SQL statement-и за създаване на таблиците:

CREATE DATABASE fitness

GO

USE fitness

GO

CREATE TABLE PRODUCTS (

ID INT NOT NULL IDENTITY(1,1) PRIMARY KEY,

NAME VARCHAR(64) NOT NULL,

DESCRIPTION VARCHAR(MAX),

PRICE FLOAT NOT NULL,

QUANTITY INT,

EXPIRYDATE DATE

);

CREATE TABLE CLIENTS (

ID INT NOT NULL IDENTITY(1,1) PRIMARY KEY,

NAME VARCHAR(MAX) NOT NULL,

EMAIL VARCHAR(64),

PHONENUMBER VARCHAR(16),

TOTALPURCHASEVALUE FLOAT DEFAULT 0,

DISCOUNTONBAR FLOAT DEFAULT 0

);

CREATE TABLE PURCHASES (

ID INT NOT NULL IDENTITY(1,1) PRIMARY KEY,

PRODUCTID INT NOT NULL,

CLIENTID INT NOT NULL,

QUANTITY int DEFAULT 1,

price float not null,

PLACEDATE DATETIME DEFAULT GETDATE()

);

CREATE TABLE SUBSCRIPTIONTYPES (

ID INT NOT NULL IDENTITY(1,1) PRIMARY KEY,

NAME VARCHAR(64) NOT NULL,

PRICE FLOAT NOT NULL,

DURATION INT,

);

CREATE TABLE SUBSCRIPTIONS (

ID INT NOT NULL IDENTITY(1,1) PRIMARY KEY,

TYPEID INT NOT NULL,

CLIENTID INT NOT NULL,

INSTRUCTORID INT,

EXPIRATIONDATE DATETIME not null

);

CREATE TABLE EMPLOYEES (

ID INT NOT NULL IDENTITY(1,1) PRIMARY KEY,

NAME VARCHAR(MAX) NOT NULL,

EMAIL VARCHAR(64),

PHONENUMBER VARCHAR(16),

ADDRESS VARCHAR(255),

EMPLOYEETYPEID INT NOT NULL,

HIREDATE DATETIME DEFAULT GETDATE(),

WORKINGHOURS INT DEFAULT 8,

BONUSSALARY FLOAT DEFAULT 0

);

CREATE TABLE EMPLOYEETYPES (

ID INT NOT NULL IDENTITY(1,1) PRIMARY KEY,

NAME VARCHAR(32) NOT NULL,

SALARY FLOAT NOT NULL

);

create table invoices(

id int not null identity(1,1) primary key,

productid int not null,

supplier VARCHAR(64) not null,

quantity int not null,

price float not null

);

III. Добавяне на примерно съдържание

USE fitness

INSERT INTO PRODUCTS (NAME, DESCRIPTION, PRICE, QUANTITY, EXPIRYDATE)

VALUES ('SAN Performance Glutamine 600gr', 'Excellent choice for everyone' , 54.00, 10, '2021/3/3');

INSERT INTO PRODUCTS (NAME, DESCRIPTION, PRICE, QUANTITY, EXPIRYDATE)

VALUES ('OPTIMUM NUTRITION Essential Amino Energy 558', NULL, 36.00, 16, '2021/8/3');

INSERT INTO PRODUCTS (NAME, DESCRIPTION, PRICE, QUANTITY, EXPIRYDATE)

VALUES ('YAMAMOTO NUTRITION Amino 120pcs', 'Tablets', 52.80, 30, '2022/1/22');

INSERT INTO PRODUCTS (NAME, DESCRIPTION, PRICE, QUANTITY, EXPIRYDATE)

VALUES ('Protein Cookie 75gr', NULL, 3.97, 100, '2020/10/15');

INSERT INTO PRODUCTS (NAME, DESCRIPTION, PRICE, QUANTITY, EXPIRYDATE)

VALUES ('Bezzo protein bar 80gr', 'Choco flavour', 2.40, 70, '2021/2/1');

INSERT INTO PRODUCTS (NAME, DESCRIPTION, PRICE, QUANTITY, EXPIRYDATE)

VALUES ('ANIMAL Animal Rage 44pack', NULL, 69.00, 7, '2023/1/3');

INSERT INTO PRODUCTS (NAME, DESCRIPTION, PRICE, QUANTITY, EXPIRYDATE)

VALUES ('Protein shake', NULL, 69.00, 7, '2020/5/13');

INSERT INTO PRODUCTS (NAME,DESCRIPTION,PRICE,QUANTITY,EXPIRYDATE)

VALUES ('Myprotein Impact Whey','A whey protein - chocolate brownie',34.00,15,'2024/1/1');

INSERT INTO CLIENTS VALUES ('Georgi Georgiev', DEFAULT, '0817263612', DEFAULT, DEFAULT);

INSERT INTO CLIENTS VALUES ('Spas Naidenov', 'spas@abv.bg', '0815267612', DEFAULT, DEFAULT);

INSERT INTO CLIENTS VALUES ('Razvigor Stoyanov', 'rasto@mail.bg', '0873563612', DEFAULT, DEFAULT);

INSERT INTO CLIENTS VALUES ('Nataliya Simeonova', DEFAULT, '0817266612', DEFAULT, DEFAULT);

INSERT INTO CLIENTS VALUES ('Dimitar Popstoykov', DEFAULT, '0817526312', DEFAULT, 10);

INSERT INTO CLIENTS VALUES ('Gergana Marinova', 'geritO@gmail.com', '0803863612', DEFAULT, DEFAULT);

INSERT INTO CLIENTS VALUES ('Stoyanka Stoeva', DEFAULT, '0817263513', DEFAULT, DEFAULT);

INSERT INTO CLIENTS VALUES ('Angel Manahilov', 'achomustacho@gmail.com', '0817251612', DEFAULT, 7.5);

INSERT INTO PURCHASES VALUES (1, 2, 7,10, DEFAULT);

INSERT INTO PURCHASES VALUES (1, 4, 2,11, DEFAULT);

INSERT INTO PURCHASES VALUES (5, 1, 3,12, DEFAULT);

INSERT INTO PURCHASES VALUES (3, 3, 3,12, DEFAULT);

INSERT INTO PURCHASES VALUES (2, 7, 5,12, DEFAULT);

INSERT INTO PURCHASES VALUES (4, 2, 1,12.5, DEFAULT);

INSERT INTO PURCHASES VALUES (5, 8, 4,12, DEFAULT);

INSERT INTO SUBSCRIPTIONTYPES VALUES ('Daily', 4.50, 1);

INSERT INTO SUBSCRIPTIONTYPES VALUES ('Weekly', 10, 7);

INSERT INTO SUBSCRIPTIONTYPES VALUES ('Monthly', 30, 30);

INSERT INTO SUBSCRIPTIONTYPES VALUES ('Yearly', 350.00, 365);

INSERT INTO SUBSCRIPTIONTYPES VALUES ('Daily(with instructor)',8,1)

INSERT INTO SUBSCRIPTIONTYPES VALUES ('Weekly(with instructor)',20,7)

INSERT INTO SUBSCRIPTIONTYPES VALUES ('Monthly(with instructor)',50.00,30)

INSERT INTO SUBSCRIPTIONTYPES VALUES ('Yearly(with instructor)',500,365)

INSERT INTO EMPLOYEETYPES VALUES ('Manager', 8.50);

INSERT INTO EMPLOYEETYPES VALUES ('Instructor', 6.50);

INSERT INTO EMPLOYEETYPES VALUES ('Janitor', 4.00);

INSERT INTO EMPLOYEETYPES VALUES ('Equipment support', 7.00);

INSERT INTO EMPLOYEETYPES VALUES ('Bartender', 5.00)

INSERT INTO EMPLOYEES VALUES ('Kiril Mutafchiev', 'kimut@gmail.com', '0861253713', 'Troyan, "Iglika" 8', 1, DEFAULT, DEFAULT, DEFAULT);

INSERT INTO EMPLOYEES VALUES ('Samuil Krumov', 'samuil@abv.bg', '0861258922', 'Karnobat, "Plodorodie" 13', 2, DEFAULT, 6, DEFAULT);

INSERT INTO EMPLOYEES VALUES ('Asen Momchilov', DEFAULT, '0834292649', 'Sliven', 3, DEFAULT, 4, DEFAULT);

INSERT INTO EMPLOYEES VALUES ('Ivan Milev', 'vanko82@gmail.com', '0861926493', 'Krichim, "Bor" 5', 2, DEFAULT, 6, DEFAULT);

INSERT INTO EMPLOYEES VALUES ('Momchil Zapryanov', DEFAULT, '0861925481', 'Blagoevgrad, "Vazrajdane" 52', 4, DEFAULT, 4, DEFAULT);

INSERT INTO EMPLOYEES VALUES ('Penka Zikatanova', DEFAULT, '0873122121', 'Studentski grad 22B', 5, DEFAULT, 8, DEFAULT);

INSERT INTO SUBSCRIPTIONS VALUES (2, 1, DEFAULT, '2020/05/10');

INSERT INTO SUBSCRIPTIONS VALUES (1, 2, DEFAULT, '2021/05/7');

INSERT INTO SUBSCRIPTIONS VALUES (3, 3, DEFAULT, '2020/05/10');

INSERT INTO SUBSCRIPTIONS VALUES (3, 7, DEFAULT, '2020/05/10');

INSERT INTO SUBSCRIPTIONS VALUES (1, 5, DEFAULT, '2020/05/10');

INSERT INTO SUBSCRIPTIONS VALUES (4, 4, DEFAULT, '2020/05/10');

INSERT INTO SUBSCRIPTIONS VALUES (3, 4, DEFAULT,'2020/05/7');

INSERT INTO SUBSCRIPTIONS VALUES (3, 2, 5,'2020/08/7');

INSERT INTO SUBSCRIPTIONS VALUES (8, 2, 2, '2021/1/10')

INSERT INTO SUBSCRIPTIONS VALUES (6, 3, 2, '2020/10/10')

*//TODO add info for invoices*

IV. Примерни заявки

IV-1. Прости заявки

Select \*

from EMPLOYEES

where EMAIL is null;

select ID, name, EXPIRYDATE

from PRODUCTS

where DATEADD(DAY,-10,EXPIRYDATE)<=GETDATE();

select ID, name, quantity

from PRODUCTS

where QUANTITY<20;

select id,name,totalpurchasevalue

from CLIENTS

where DISCOUNTONBAR>0;

select ID,name

from CLIENTS

where TOTALPURCHASEVALUE>1000 and DISCOUNTONBAR=0;

select \*

from PURCHASES

where CLIENTID=3;

select \*

from invoices

where price>2000;

select \*

from SUBSCRIPTIONTYPES

where DURATION>7;

IV-2. Заявки върху 2 или повече релации

select CLIENTS.NAME as client\_name,CLIENTS.PHONENUMBER as client\_phone\_number, TYPEid as subscription\_type, EXPIRATIONDATE as expiration\_date

from SUBSCRIPTIONS, CLIENTS

where CLIENTID=CLIENTS.ID and (typeid=3 or TYPEID=2) and DATEADD(DAY,-5,EXPIRATIONDATE)<=GETDATE();

(select id

from employees

where workinghours = 4)

intersect

(select instructorid

from subscriptions

where expirationdate>getdate());

select EMPLOYEES.ID,EMPLOYEES.NAME,EMPLOYEETYPES.SALARY,EMPLOYEES.WORKINGHOURS

from EMPLOYEES,EMPLOYEETYPES

where EMPLOYEES.EMPLOYEETYPEID=EMPLOYEETYPES.ID and EMPLOYEES.BONUSSALARY=0

order by salary;

select CLIENTS.ID as client\_id, CLIENTS.name as client\_name, PURCHASES.price, PURCHASES.PLACEDATE

from CLIENTS,PURCHASES

where CLIENTS.ID=PURCHASES.CLIENTID

select CLIENTS.ID as client\_id, CLIENTS.name as client\_name, SUBSCRIPTIONTYPES.NAME, SUBSCRIPTIONS.INSTRUCTORID

from CLIENTS,SUBSCRIPTIONS,SUBSCRIPTIONTYPES

where CLIENTS.ID=SUBSCRIPTIONS.CLIENTID and SUBSCRIPTIONTYPES.ID=SUBSCRIPTIONS.TYPEID and INSTRUCTORID is not null;

IV-3. Подзаявки

IV-4. Съединения

IV-5. Групиране и агрегация

V. Заключение

V-1. Ограничения

Alter table invoices

add constraint invoices\_productid\_fk foreign key(productid) references products(id);

Alter table employees

add constraint employees\_typeid\_fk foreign key(employeetypeid) references employeetypes(id);

Alter table purchases

add constraint purchases\_clientid\_fk foreign key(clientid) references clients(id);

Alter table purchases

add constraint purchases\_productid\_fk foreign key(productid) references products(id);

Alter table subscriptions

add constraint subs\_clientid\_fk foreign key(clientid) references clients(id);

Alter table subscriptions

add constraint subs\_instructorid\_fk foreign key(instructorid) references employees(id);

Alter table subscriptions

add constraint subs\_typeid\_fk foreign key(typeid) references subscriptiontypes(id);

ALTER TABLE products

ADD CONSTRAINT products\_positive\_quantity CHECK (quantity>0)

ALTER TABLE INVOICES

ADD CONSTRAINT invoices\_positive\_quantity CHECK (quantity>0)

V-2. Изгледи, индекси и тригери

Тригери:

USE fitness

GO

CREATE TRIGGER PURCHASES\_AFTER\_INSERT

ON PURCHASES

AFTER INSERT

AS

BEGIN

SET NOCOUNT ON;

UPDATE PRODUCTS

SET QUANTITY = QUANTITY - (SELECT QUANTITY FROM INSERTED)

WHERE ID = (SELECT PRODUCTID FROM INSERTED);

UPDATE CLIENTS

SET TOTALPURCHASEVALUE = TOTALPURCHASEVALUE + (SELECT PRICE FROM PRODUCTS WHERE ID = (SELECT PRODUCTID FROM INSERTED)) \* (SELECT QUANTITY FROM INSERTED)

WHERE ID = (SELECT CLIENTID FROM INSERTED)

END

GO

CREATE TRIGGER SUBSCRIPTIONS\_AFTER\_INSERT

ON SUBSCRIPTIONS

AFTER INSERT

AS

BEGIN

SET NOCOUNT ON;

UPDATE SUBSCRIPTIONS

SET EXPIRATIONDATE = (SELECT DATEADD(DAY, (SELECT DURATION FROM SUBSCRIPTIONTYPES WHERE id = (SELECT TYPEid FROM INSERTED)), GETDATE()))

WHERE ID = (SELECT ID FROM INSERTED);

UPDATE CLIENTS

SET TOTALPURCHASEVALUE = TOTALPURCHASEVALUE + (SELECT PRICE FROM SUBSCRIPTIONTYPES WHERE id= (SELECT TYPEid FROM INSERTED))

WHERE ID = (SELECT CLIENTID FROM INSERTED)

END

GO

Индекси и изгледи:

use fitness

go

-------------------views-----------------------

create view monthly\_subscribed\_clients

as

select subscriptions.clientid, clients.name, clients.phonenumber

from subscriptions, clients

where subscriptions.clientid=clients.id and subscriptions.typeid in (select id from subscriptiontypes where duration=30);

create view instructors\_info

as

select employees.id, employees.name, employees.phonenumber, employees.workinghours

from employees,employeetypes

where employeetypeid=employeetypes.id and employeetypes.name = 'Instructor';

create view products\_needing\_restock

as

select ID, name, quantity

from PRODUCTS

where QUANTITY<10;

create view expired\_subscriptions

as

select clientid,TYPEid,expirationdate

from SUBSCRIPTIONS

where EXPIRATIONDATE<GETDATE();

create view loyal\_clients

as

select id,name,phonenumber,discountonbar

from CLIENTS

where DISCOUNTONBAR>0;

-------------------------indeces-------------------------

create index idx\_clients\_info

on clients(id,phonenumber);

create unique index idx\_subscription\_types

on subscriptiontypes(name);

create index idx\_employees

on employees(id,employeetypeid);

create index idx\_employeetypes

on employeetypes(name);