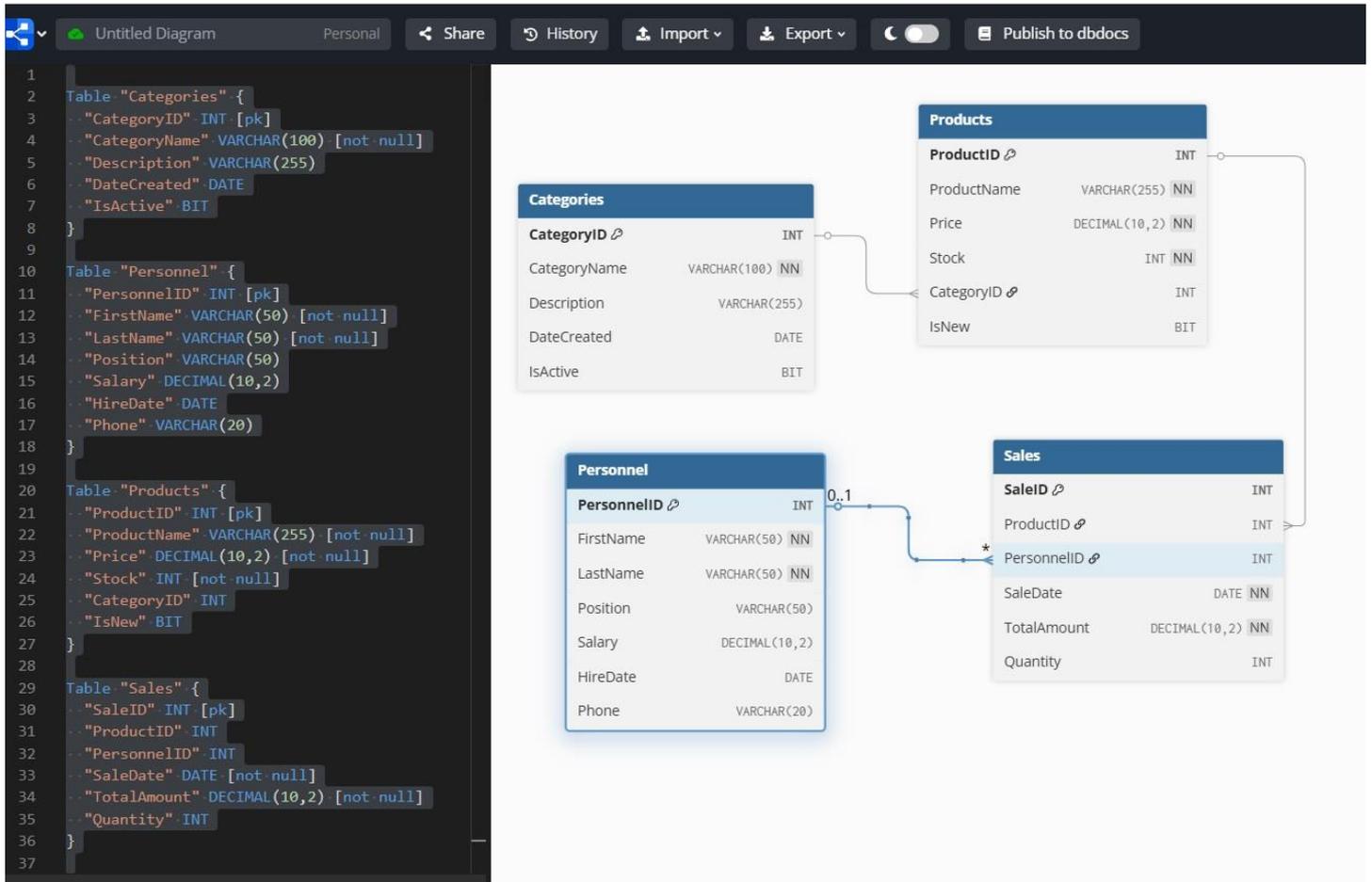


# ЛОВНО-РИБАРСКИ МАГАЗИН



-- Ловно-рибарски магазин - продукти по категории, цени, персонал и др.

-- използвани типове данни:

-- INT - за цели числа без десетична част, използван за уникални ID-та.

-- VARCHAR(N)- Текст със променлива дължина, използван най-често за имена, глъжноти и т.н.

-- DECIMAL(p,s)- числа с фиксирана точност, използвано за парични стойности(цени, заплати).

-- DATE - само за дата (година, месец, ден), използвано за дати на наемане и продажби.

-- BIT - съхранява 1 или 0 (true or false) - използвано за отбелязване дали даден запис е активен.

--изчистване на таблици ако съществуват.

```
DROP TABLE IF EXISTS Sales;
```

```
DROP TABLE IF EXISTS Products;
```

```
DROP TABLE IF EXISTS Personnel;
```

```
DROP TABLE IF EXISTS Categories;
```

-- Създава таблица с 5 полета (1)

```
Create TABLE Categories (
  CategoryID INT PRIMARY KEY,
  CategoryName VARCHAR(100) NOT NULL, --(not null е вид ограничение.)
  Description VARCHAR(255),
  DateCreated DATE,
  IsActive BIT
);
```

--2

```
CREATE TABLE Personnel (
  PersonnelID INT PRIMARY KEY,
  FirstName VARCHAR(50) NOT NULL,
```

```
LastName VARCHAR(50) NOT NULL,  
Position VARCHAR(50),  
Salary DECIMAL(10, 2),  
HireDate DATE,  
Phone VARCHAR(20)
```

```
);
```

```
--3
```

```
create TABLE Products (  
ProductID INT PRIMARY KEY,  
ProductName VARCHAR(255) NOT NULL,  
Price DECIMAL(10, 2) NOT NULL,  
Stock INT NOT NULL,  
CategoryID INT,  
IsNew BIT,  
FOREIGN KEY (CategoryID) REFERENCES Categories(CategoryID) -- създава foreign key
```

```
);
```

```
--4
```

```
CREATE TABLE Sales (  
SaleID INT PRIMARY KEY,  
ProductID INT,  
PersonnelID INT,  
SaleDate DATE NOT NULL,  
TotalAmount DECIMAL(10, 2) NOT NULL, Quantity INT,  
FOREIGN KEY (ProductID) REFERENCES Products(ProductID)  
FOREIGN KEY (PersonnelID) REFERENCES Personnel(PersonnelID) );
```

```
ALTER TABLE Sales  
ADD Quantity INT;
```

```
-- промяна на таблица
```

```
ALTER TABLE Personnel  
ADD Email VARCHAR(100);
```

```
-- 10 ЗАПИСА В Categories
```

```
INSERT INTO Categories (CategoryID, CategoryName, Description, DateCreated, IsActive) VALUES  
(1, 'Риболовни пръчки', 'Въдици и макари', '2023-01-01', 1),  
(2, 'Стръв и захранки', 'Естествена и изкуствена стръв', '2023-01-01', 1),  
(3, 'Ловно облекло', 'Термо облекла и камуфлаж', '2023-02-15', 1),  
(4, 'Оръжия и амуниции', 'Пушки, пистолети и патрони', '2023-03-10', 0),  
(5, 'Акcesoари', 'Фенери, ножове, раници', '2023-04-01', 0),  
(6, 'Ловни кучета', 'Храна и акcesoари', '2023-05-01', 1),  
(7, 'Морски риболов', 'Специализирана екипировка', '2023-06-01', 1),  
(8, 'Къмпинг', 'Палатки и спални чували', '2023-07-01', 1),  
(9, 'Защитна екипировка', 'Очила, ръкавици', '2023-08-01', 1),  
(10, 'Книги и справочници', 'Ръководства за лов и риболов', '2023-09-01', 1);
```

```
-- 10 записа в Personnel
```

```
INSERT INTO Personnel (PersonnelID, FirstName, LastName, Position, Salary, HireDate, Phone, Email)  
VALUES  
(101, 'Иван', 'Петров', 'Мениджър', 2500.00, '2020-05-15', '088111222', 'i.petrov@shop.bg'),  
(102, 'Мария', 'Георгиева', 'Продавач', 1800.00, '2021-08-20', '088222333', 'm.georgieva@shop.bg'),  
(103, 'Георги', 'Димитров', 'Продавач', 1850.00, '2022-01-10', '088333444', 'g.dimitrov@shop.bg'),
```

```
(104, 'Елена', 'Колева', 'Касиев', 1500.00, '2023-03-01', '088444555', 'e.koleva@shop.bg'),
(105, 'Димитър', 'Илиев', 'Склад', 1400.00, '2023-06-01', '088555666', 'd.iliev@shop.bg'),
(106, 'Ана', 'Тодорова', 'Продавач', 1750.00, '2023-10-01', '088666777', 'a.todorova@shop.bg'),
(107, 'Николай', 'Христов', 'Продавач', 1900.00, '2024-01-01', '088777888', 'n.hristov@shop.bg'),
(108, 'Калин', 'Йорданов', 'Склад', 1450.00, '2024-02-15', '088888999', 'k.yordanov@shop.bg'),
(109, 'Петя', 'Станчева', 'Касиер', 1550.00, '2024-04-01', '088999000', 'p.stanchev@shop.bg'),
(110, 'Стефан', 'Атанасов', 'Мениджър', 2600.00, '2019-12-01', '088000111', 's.atanasov@shop.bg');
```

-- 10 записа в Products

```
INSERT INTO Products (ProductID, ProductName, Price, Stock, CategoryID, IsNew) VALUES
(1001, 'Спининг пръчка "Щука"', 85.50, 40, 1, 0),
(1002, 'Комплект за хранка', 15.90, 120, 2, 1),
(1003, 'Термо яке "Ловец"', 150.00, 25, 3, 1),
(1004, 'Патрони кал. 12(25бр.)', 35.00, 80, 4, 0),
(1005, 'Мълтитул "Природа"', 45.75, 50, 5, 0),
(1006, 'Въдица телескоп 3.6м', 45.00, 60, 1, 0),
(1007, 'Бойли с аромат "Ягода"', 7.50, 200, 2, 1),
(1008, 'Панталон ловенн', 95.00, 30, 3, 0),
(1009, 'Въздушна пушка 5.5мм', 299.90, 15, 4, 1),
(1010, 'Водоустойчив фенер', 22.50, 75, 5, 0);
```

-- 10 записа в Sales

```
INSERT INTO Sales (SaleID, ProductID, PersonnelID, SaleDate, Quantity, TotalAmount) VALUES
(1, 1001, 102, '2025-11-28', 1, 85.50),
(2, 1003, 101, '2025-11-28', 1, 150.00),
(3, 1004, 102, '2025-11-29', 2, 70.00),
(4, 1006, 103, '2025-11-29', 3, 135.00),
(5, 1008, 103, '2025-11-30', 1, 95.00),
(6, 1002, 104, '2025-12-01', 5, 79.50),
(7, 1009, 101, '2025-12-01', 1, 299.00),
(8, 1001, 102, '2025-12-01', 2, 171.00),
(9, 1007, 104, '2025-12-02', 10, 75.00),
(10, 1010, 103, '2025-12-02', 1, 22.50);
```

-- обновяване и изтриване (UPDATE, DELETE, обновяване на заплатата в случая)

```
UPDATE Personnel
SET Salary = 1950.00
WHERE FirstName = 'Мария' AND LastName = 'Георгиева';
```

--изтрива от таблицата продукти всички продукти които имат наличност по- малко от 10 (DELETE)

```
DELETE FROM Products
Where Stock < 10;
```

-- основни SELECT заявкиdemo

--1. всичко в таблица продукти(SELECT)

```
Select *
From Products;
```

--2. Тези с цена над 50.00лв.(WHERE)

```
SELECT ProductName, Price
FROM Products
WHERE Price > 50.50;
```

--3. Тези които са нови и са на цена под 100.00лв.

```
SELECT ProductName, Price, IsNew
FROM Products
```

```
WHERE IsNew = 1 AND Price < 100.00;
```

```
--4. Служителите които са продавач или касиер и ги сортира по заплата (OR, ORDER BY).
```

```
select FirstName, LastName, Position, Salary  
FROM Personnel  
Where Position = 'Продавач' OR Position = 'Касиер'  
ORDER BY Salary DESC;
```

```
--5. (GROUP BY, SUM, AVG, MAX, MIN)
```

```
SELECT  
PersonnelID,  
COUNT(SaleID) AS TotalSalesCount,  
SUM(TotalAmount) AS TotalRevenue,  
AVG(TotalAmount) AS AverageSale,  
MAX(TotalAmount) AS MaxSale,  
MIN(TotalAmount) AS MinSale  
FROM Sales  
GROUP BY PersonnelID;
```

```
--6.(INNER JOIN, извлича името на продукта и името на категорията, към която принадлежи  
(съкращава пътят за намиране, чрез псевдоними)
```

```
-- INNER JOIN Categories AS c ON p.CategoryID = c.CategoryID -> ("Вземи данните от таблицата  
Categories (означена с c)
```

```
-- -- С помощта на AS се задават псевдоними на таблиците
```

```
--и ги свържи с таблицата Products (означена с p), като съответстват по колоната CategoryID.")
```

```
SELECT  
p.ProductName,  
c.CategoryName,  
p.Price  
FROM Products AS p  
INNER JOIN Categories as c ON p.CategoryID = c.CategoryID  
ORDER BY c.CategoryName;
```

```
--7.изглед за служителите с висока заплата (CREATE VIEW)
```

```
CREATE VIEW HighSalaryPersonnelView AS
```

```
SELECT  
FirstName,  
LastName,  
Position,  
Salary  
FROM Personnel  
WHERE SALARY > 2000.00;
```

```
-- извлича данни от създадения изглед.
```

```
SELECT *  
FROM HighSalaryPersonnelView;
```

```
-- izpolzvani sa (PRIMARY KEY, FOREIGN KEY, CREATE TABLE, DROP, ALTER, INSERT, UPDATE,  
SELECT, WHERE, OR, ORDER BY, GROUP BY, SUM, AVG ,MIN ,MAX,
```

```
-- CREATE VIEW, INNER JOIN)
```