МИНОБРНАУКИ РОССИИ

Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет «Московский институт электронной техники»

Лабораторная работа №7-8 по дисциплине «Системы управления базами данных» Мини-проект

Подготовил:

Студент группы ПИН-32 Трусов М.П.

Задание

Вариант 9. Экспонаты в музее.

Необходимо в соответствии с заданием разработать и реализовать БД и приложение для работы с ней.

База данных должна содержать не менее:

- 8 заполненных таблиц,
- 2 представлений,
- 2 триггеров,
- 5 процедур / функций.

Приложение должно демонстрировать:

- Отображение / добавление / обновлене / удаление данных в БД.
- Работу триггера / процерур и функций.

Создание структуры БД

```
-- 1. Таблица "Музеи"
CREATE TABLE Museums (
   MuseumID INT PRIMARY KEY,
   Name VARCHAR(100) NOT NULL,
   Address VARCHAR(200) NOT NULL,
   FoundationDate DATE,
   Director VARCHAR(100),
   Website VARCHAR(100)
);
-- 2. Таблица "Залы"
CREATE TABLE Halls (
   HallID INT PRIMARY KEY,
   MuseumID INT NOT NULL,
   Name VARCHAR(50) NOT NULL,
   Floor INT NOT NULL,
   Area DECIMAL(10,2),
   Theme VARCHAR(100),
    FOREIGN KEY (MuseumID) REFERENCES Museums(MuseumID)
);
-- 3. Таблица "Экспонаты"
CREATE TABLE Exhibits (
   ExhibitID INT PRIMARY KEY,
   HallID INT NOT NULL,
   Name VARCHAR(100) NOT NULL,
   Description TEXT,
   CreationDate DATE,
   Author VARCHAR(100),
   OriginCountry VARCHAR(50),
   AcquisitionDate DATE,
   Value DECIMAL(15,2),
   Status VARCHAR(50) CHECK (Status IN ('Ha выставке', 'В хранилище', 'На реставрации', 'В аренде')),
   FOREIGN KEY (HallID) REFERENCES Halls(HallID)
);
```

```
-- 4. Таблица "Категории экспонатов"
 CREATE TABLE ExhibitCategories (
     CategoryID INT PRIMARY KEY,
     Name VARCHAR(50) NOT NULL,
     Description TEXT
 );
 -- 5. Таблица "Связь экспонатов и категорий"
 CREATE TABLE ExhibitCategoryRelations (
     ExhibitID INT NOT NULL,
     CategoryID INT NOT NULL,
     PRIMARY KEY (ExhibitID, CategoryID),
     FOREIGN KEY (ExhibitID) REFERENCES Exhibits(ExhibitID),
     FOREIGN KEY (CategoryID) REFERENCES ExhibitCategories(CategoryID)
 );
 -- 6. Таблица "Сотрудники"
 CREATE TABLE Employees (
     EmployeeID INT PRIMARY KEY,
     MuseumID INT NOT NULL,
     FirstName VARCHAR(50) NOT NULL,
     LastName VARCHAR(50) NOT NULL,
     Position VARCHAR (50) NOT NULL,
     HireDate DATE NOT NULL,
     Email VARCHAR(100),
     Phone VARCHAR(20),
     FOREIGN KEY (MuseumID) REFERENCES Museums(MuseumID)
 );
-- 7. Таблица "Реставрации"

    CREATE TABLE Restorations (

    RestorationID INT PRIMARY KEY,
    ExhibitID INT NOT NULL,
    EmployeeID INT NOT NULL,
    StartDate DATE NOT NULL,
    EndDate DATE,
    Description TEXT,
    Cost DECIMAL(10,2),
    FOREIGN KEY (ExhibitID) REFERENCES Exhibits(ExhibitID),
    FOREIGN KEY (EmployeeID) REFERENCES Employees(EmployeeID)
 );
 -- 8. Таблица "Посетители"
CREATE TABLE Visitors (
    VisitorID INT PRIMARY KEY,
    FirstName VARCHAR(50) NOT NULL,
    LastName VARCHAR(50) NOT NULL,
    Email VARCHAR(100),
    Phone VARCHAR(20),
    VisitDate DATE NOT NULL.
    TicketType VARCHAR(30) CHECK (TicketType IN ('Взрослый', 'Детский', 'Льготный', 'Групповой'))
 );
```

```
| miscome | misc
```

Заполнение таблиц тестовыми данными

```
INSERT INTO Museums (MuseumID, Name, Address, FoundationDate, Director, Website)
VALUES
(1, 'Государственный Эрмитаж', 'Дворцовая пл., 2, Санкт-Петербург', '1764-02-05', 'Михаил Пиотровский', 'hermitagemuseum.org'), (2, 'Пушкинский музей', 'ул. Волхонка, 12, Москва', '1912-05-31', 'Марина Лошак', 'pushkinmuseum.art'),
(3, 'Третьяковская галерея', 'Лаврушинский пер., 10, Москва', '1856-05-22', 'Зельфира Трегулова', 'tretyakovgallery.ru');
-- Заполнение залов
INSERT INTO Halls (HallID, MuseumID, Name, Floor, Area, Theme)
VALUES
(1, 1, 'Главный зал', 1, 500.00, 'Искусство эпохи Возрождения'),
(2, 1, 'Зал древностей', 2, 350.50, 'Античное искусство'),
(3, 2, 'Галерея импрессионистов', 1, 400.75, 'Импрессионизм'),
(4, 2, 'Зал древнего Египта', 2, 300.00, 'Древний Египет'),
(5, 3, 'Зал русских икон', 1, 450.25, 'Русская иконопись'),
(6, 3, 'Зал передвижников', 2, 380.00, 'Товарищество передвижных выставок');
-- Заполнение категорий экспонатов
INSERT INTO ExhibitCategories (CategoryID, Name, Description)
VALUES
(1, 'Живопись', 'Картины и художественные полотна'),
(2, 'Скульптура', 'Объемные художественные произведения'),
(3, 'Артефакт', 'Исторические предметы и археологические находки'),
(4, 'Икона', 'Религиозные изображения'),
(5, 'Графика', 'Рисунки, гравюры, литографии');
   - Заполнение экспонатов
INSERT INTO Exhibits (ExhibitID, HallID, Name, Description, CreationDate, Author, OriginCountry, AcquisitionDate, Value, Status)
VALUES
VALUES
(1, 1, 'Мадонна Литта', 'Картина Леонардо да Винчи', '1490-01-01', 'Леонардо да Винчи', 'Италия', '1865-03-12', 1000000000.00, 'На выставке'), (2, 1, 'Возвращение блудного сына', 'Картина Рембрандта', '1668-01-01', 'Рембрандт', 'Нидерланды', '1766-07-18', 850000000.00, 'На выставке'), (3, 3, 'Голубые танцовщицы', 'Картина Эдгара Дега', '1897-01-01', 'Эдгар Дега', 'Франция', '1948-11-05', 450000000.00, 'На выставке'), (4, 4, 'Статуя Нефертити', 'Бюст царицы Нефертити', '1345-01-01', 'Тутмос', 'Древний Египет', '1920-05-15', 750000000.00, 'На реставрации'), (5, 5, 'Троица', 'Икона Андрея Рублева', '1425-01-01', 'Андрей Рублев', 'Русь', '1929-10-20', 900000000.00, 'На выставке'), (6, 6, 'Бурлаки на Волге', 'Картина Ильи Репина', '1873-01-01', 'Илья Репин', 'Россия', '1873-05-30', 600000000.00, 'В хранилище');
 -- Связи экспонатов и категорий
INSERT INTO ExhibitCategoryRelations (ExhibitID, CategoryID)
VALUES
(1, 1), (2, 1), (3, 1), (4, 2), (5, 4), (6, 1);
  - Заполнение сотрудников
INSERT INTO Employees (EmployeeID, MuseumID, FirstName, LastName, Position, HireDate, Email, Phone)
VALUES
 (1, 1, 'Иван', 'Петров', 'Хранитель', '2010-06-15', 'i.petrov@hermitage.ru', '+79111234567')
(1, 1, 'Иван', 'Петров', 'Хранитель', '2010-00-10', 'Lipetrovgnermitage.ru', '1911/23-50'), (2, 1, 'Елена', 'Смирнова', 'Реставратор', '2015-09-20', 'e.smirnova@hermitage.ru', '+79117654321'), (3, 2, 'Алексей', 'Кузнецов', 'Экскурсовод', '2018-04-10', 'a.kuznetsov@pushkinmuseum.ru', '+79035557788'), (4, 2, 'Ольга', 'Иванова', 'Хранитель', '2012-11-05', 'o.ivanova@pushkinmuseum.ru', '+79036668899'), (5, 3, 'Дмитрий', 'Соколов', 'Директор зала', '2005-08-12', 'd.sokolov@tretyakov.ru', '+79119998877'), (6, 3, 'Мария', 'Васильева', 'Реставратор', '2017-03-25', 'm.vasileva@tretyakov.ru', '+79118887766');
```

```
-- Заполнение реставраций
INSERT INTO Restorations (RestorationID, ExhibitID, EmployeeID, StartDate, EndDate, Description, Cost)
VALUES
(1, 4, 2, '2023-01-10', NULL, 'Pectabpaqua быста Нефертити', 500000.00),
(2, 5, 6, '2022-11-15', '2023-02-20', 'Консервация иконы "Троица"', 350000.00);

-- Заполнение посетителей
INSERT INTO Visitors (VisitorID, FirstName, LastName, Email, Phone, VisitDate, TicketType)
VALUES
(1, 'Анна', 'Сидорова', 'a.sidorova@mail.ru', '+79031234567', '2023-05-10', 'Варослый'),
(2, 'Петр', 'Васильев', 'p.vasilev@gmail.com', '+79042345678', '2023-05-11', 'Варослый'),
(3, 'Екатерина', 'Михайлова', 'e.mikhailova@yandex.ru', '+79053456789', '2023-05-12', 'Льготный'),
(4, 'Сергей', 'Николаев', NULL, '+79064567890', '2023-05-13', 'Групповой'),
(5, 'Марина', 'Федорова', 'm.fedorova@mail.ru', NULL, '2023-05-14', 'Детский');
```

Создание представлений

```
Query Query History
1 -- 1. Представление "Дорогие экспонаты" (стоимость выше 500 млн)
2 v CREATE VIEW ValuableExhibits AS
3 SELECT e.ExhibitID, e.Name, e.Value, m.Name AS Museum, h.Name AS Hall
  FROM Exhibits e
4
5
    JOIN Halls h ON e.HallID = h.HallID
JOIN Museums m ON h.MuseumID = m.MuseumID
   WHERE e.Value > 500000000.00
7
8 ORDER BY e.Value DESC;
9
10
   -- 2. Представление "Текущие реставрации"
11 V CREATE VIEW CurrentRestorations AS
12
    SELECT r.RestorationID, e.Name AS Exhibit,
           CONCAT(emp.FirstName, ' ', emp.LastName) AS Restorer,
13
           r.StartDate, r.Description, r.Cost
14
15
    FROM Restorations r
    JOIN Exhibits e ON r.ExhibitID = e.ExhibitID
16
17
    JOIN Employees emp ON r.EmployeeID = emp.EmployeeID
  WHERE r.EndDate IS NULL;
18
```

Создание триггеров

```
-- 1. Триггер для логирования изменений статуса экспонатов
CREATE OR REPLACE FUNCTION log_exhibit_status_change()
RETURNS TRIGGER AS $$
BEGIN
    IF OLD. Status IS DISTINCT FROM NEW. Status THEN
        INSERT INTO ExhibitStatusLog (ExhibitID, OldStatus, NewStatus, ChangedBy)
        VALUES (NEW.ExhibitID, OLD.Status, NEW.Status, current_user);
    END IF;
    RETURN NEW;
END;
$$ LANGUAGE plpgsql;
CREATE TRIGGER trg_exhibit_status_change
AFTER UPDATE ON Exhibits
FOR EACH ROW EXECUTE FUNCTION log_exhibit_status_change();
-- 2. Триггер для проверки даты реставрации
CREATE OR REPLACE FUNCTION check_restoration_date()
RETURNS TRIGGER AS $$
DECLARE
   exhibit_date DATE;
BEGIN
   SELECT CreationDate INTO exhibit_date FROM Exhibits WHERE ExhibitID = NEW.ExhibitID;
   IF NEW.StartDate < exhibit_date THEN</pre>
       RAISE EXCEPTION 'Дата начала реставрации не может быть раньше даты создания экспоната';
   END IF;
   RETURN NEW;
END;
$$ LANGUAGE plpgsql;
CREATE TRIGGER trg_check_restoration_date
BEFORE INSERT ON Restorations
FOR EACH ROW EXECUTE FUNCTION check_restoration_date();
```

Создание процедур и функций

```
-- 1. Процедура для перемещения экспоната между залами
CREATE OR REPLACE PROCEDURE move_exhibit(exhibit_id INT, new_hall_id INT)
LANGUAGE plpgsql AS $$
DECLARE
   current_status VARCHAR(50);
BEGIN
   SELECT Status INTO current_status FROM Exhibits WHERE ExhibitID = exhibit_id;
   IF current_status != 'На выставке' THEN
      RAISE EXCEPTION 'Нельзя перемещать экспонат, который не находится на выставке';
   FLSE
      UPDATE Exhibits SET HallID = new_hall_id WHERE ExhibitID = exhibit_id;
   END IF;
END:
$$;
-- 2. Процедура для добавления нового экспоната
CREATE OR REPLACE PROCEDURE add_new_exhibit(
   hall id INT.
   name VARCHAR(100),
   description TEXT
   creation_date DATE,
   author VARCHAR(100),
   country VARCHAR(50),
   value DECIMAL(15,2)
LANGUAGE plpgsql AS $$
BEGIN
   INSERT INTO Exhibits (HallID, Name, Description, CreationDate, Author, OriginCountry, AcquisitionDate, Value, Status)
   VALUES (hall_id, name, description, creation_date, author, country, CURRENT_DATE, value, 'Ha выставке');
END:
$$;
          -- 3. Функция для расчета средней стоимости экспонатов в музее
          CREATE OR REPLACE FUNCTION get_avg_exhibit_value(museum_id INT)
          RETURNS DECIMAL(15,2) AS $$
          BEGIN
               RETURN (
                   SELECT AVG(e.Value)
                    FROM Exhibits e
                    JOIN Halls h ON e.HallID = h.HallID
                    WHERE h.MuseumID = museum_id
               );
           END;
          $$ LANGUAGE plpgsql;
          -- 4. Функция для получения экспонатов по категории
          CREATE OR REPLACE FUNCTION get_exhibits_by_category(category_name VARCHAR(50))
          RETURNS TABLE (
               exhibit_id INT,
               name VARCHAR(100),
               author VARCHAR(100),
               creation_date DATE,
               museum VARCHAR(100)
          ) AS $$
          BEGIN
               RETURN QUERY
               SELECT e.ExhibitID, e.Name, e.Author, e.CreationDate, m.Name
               FROM Exhibits e
               JOIN ExhibitCategoryRelations ecr ON e.ExhibitID = ecr.ExhibitID
               JOIN ExhibitCategories ec ON ecr.CategoryID = ec.CategoryID
               JOIN Halls h ON e.HallID = h.HallID
               JOIN Museums m ON h.MuseumID = m.MuseumID
               WHERE ec.Name = category_name
               ORDER BY e.Name:
           $$ LANGUAGE plpgsql;
```

```
-- 5. Функция для подсчета количества экспонатов в зале

CREATE OR REPLACE FUNCTION count_exhibits_in_hall(hall_id INT)

RETURNS INT AS $$

BEGIN

RETURN (

SELECT COUNT(*)

FROM Exhibits

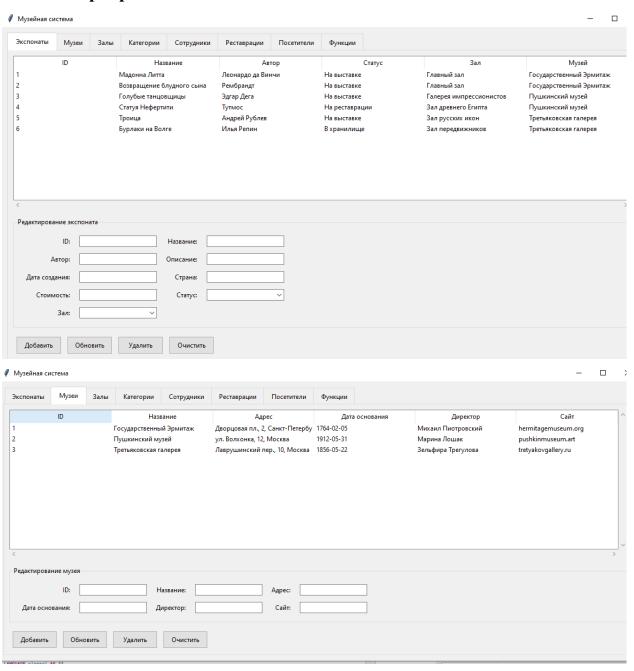
WHERE HallID = hall_id AND Status = 'Ha выставке'

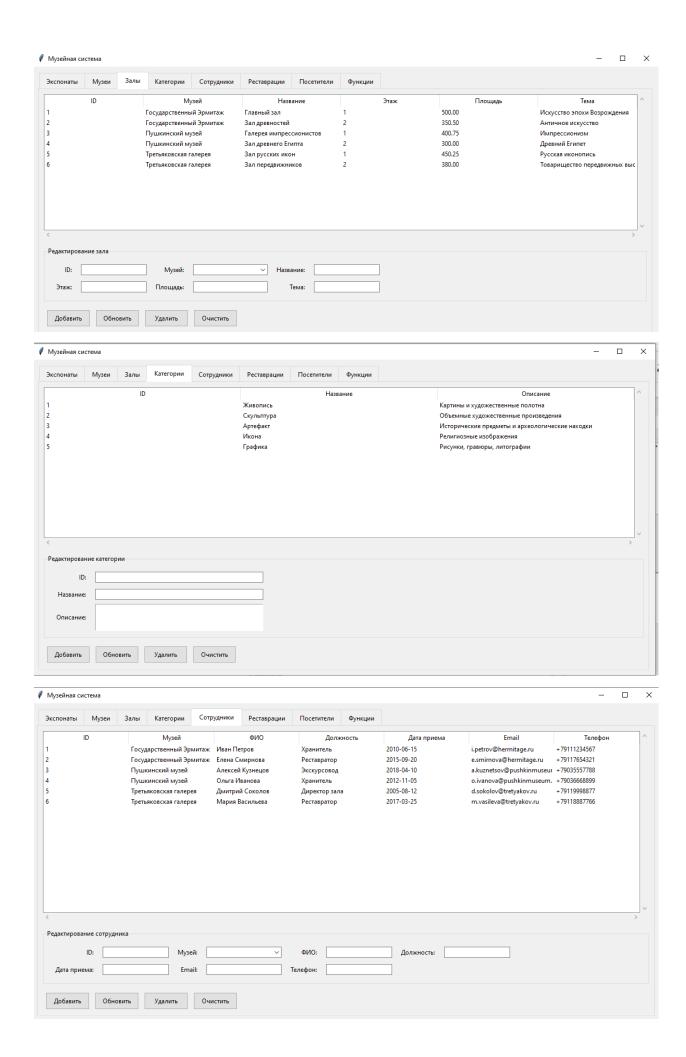
);

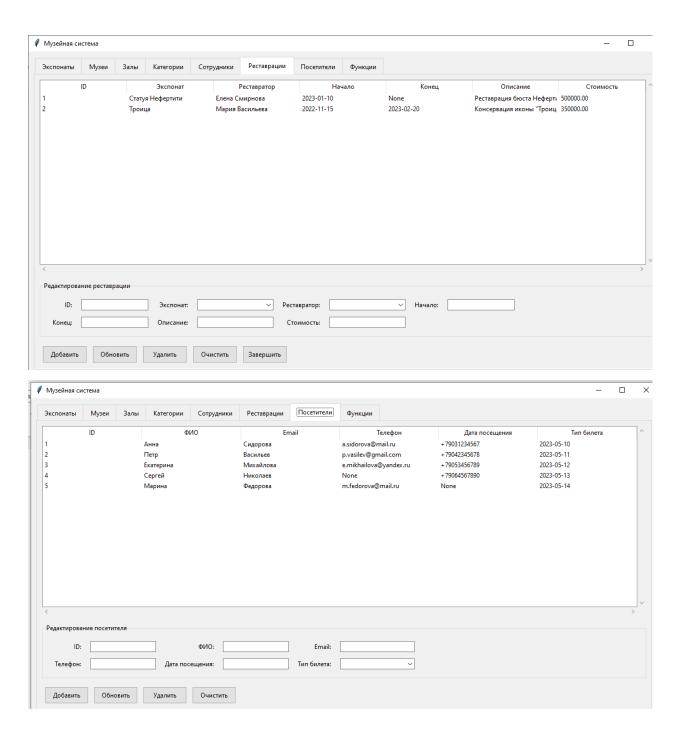
END;

$$ LANGUAGE plpgsql;
```

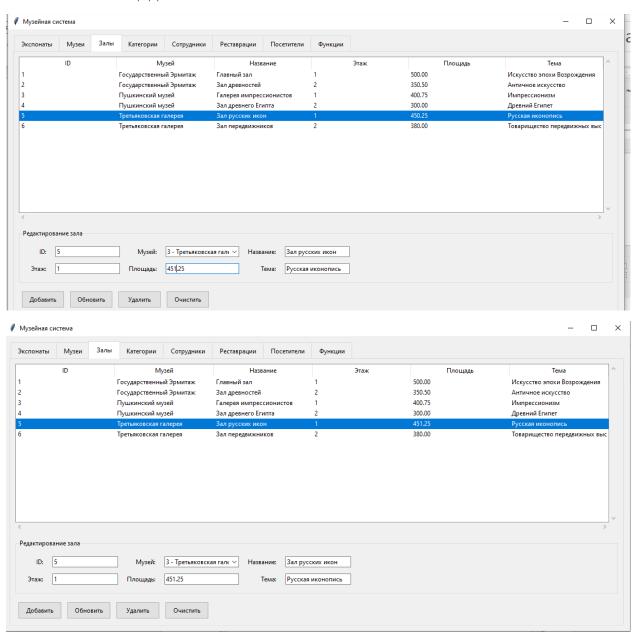
Готовая программа:







Обновим площадь



Функции:

