

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Казанский национальный исследовательский технический университет
им. А.Н. Туполева – КАИ»
Институт компьютерных технологий и защиты информации
Отделение СПО ИКТЗИ (Колледж информационных технологий)

ЛАБОРАТОРНАЯ РАБОТА №7
по дисциплине

Технология разработки и защиты баз данных

Тема: «ПРЕДСТАВЛЕНИЯ, ТРИГГЕРЫ, ХРАНИМЫЕ ПРОЦЕДУРЫ И
ТРАНЗАКЦИИ»

Работу выполнил
Студент гр.4235
Желваков А. С.

Принял
Преподаватель Валова П. А.

Казань 2024

Цель: Ознакомиться с представлениями, триггерами, хранимыми процедурами и транзакциями.

Задание на лабораторную работу:

Создать 5 представлений, 5 триггеров, 5 хранимых процедур

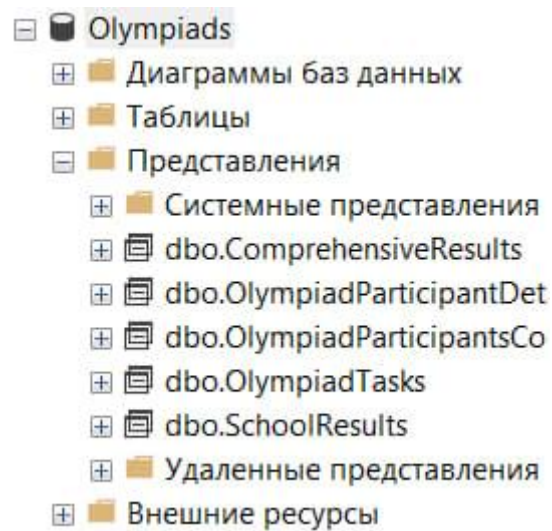


Рис. 1 - Созданные представления

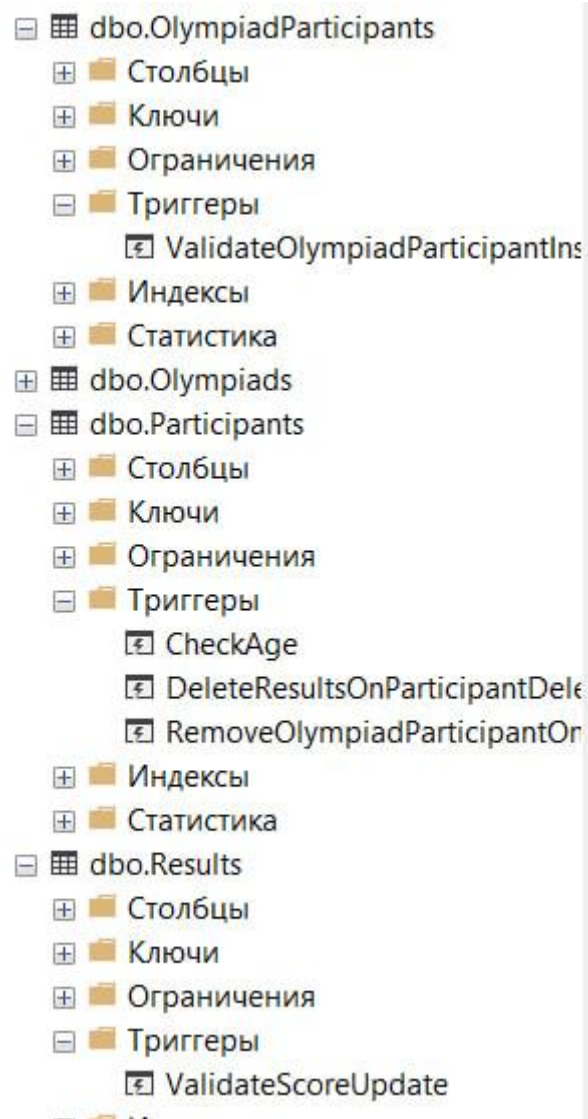


Рис. 2 - Созданные триггеры

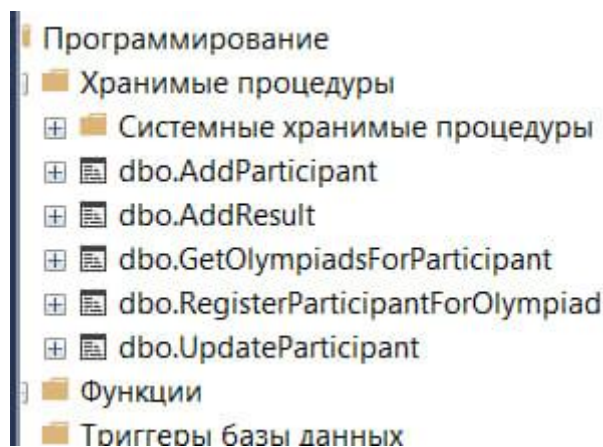


Рис. 3 - Созданные хранимые процедуры

```
select * from OlympiadParticipantsCount
```

110 %

Результаты Сообщения

	OlympiadName	ParticipantCount
1	Олимпиада по математике	7
2	Олимпиада по физике	6

Рис. 4 - Представление 1

```
select * from SchoolResults
```

	SchoolName	AvgScore	MaxScore	MinScore
1	Школа №1	95	95	95
2	Школа №10	87	87	87
3	Школа №11	91	91	91
4	Школа №12	89	89	89
5	Школа №13	93	93	93
6	Школа №2	88	88	88
7	Школа №3	92	92	92
8	Школа №4	78	78	78
9	Школа №5	85	85	85
10	Школа №6	90	90	90
11	Школа №7	94	94	94
12	Школа №8	82	82	82
13	Школа №9	96	96	96

Рис. 5 - Представление 2

```
select * from OlympiadTasks
```

110 %

Результаты Сообщения

	OlympiadName	SubjectName	TaskDescription
1	Олимпиада по математике	Математика	Решить задачу на геометрию
2	Олимпиада по математике	Математика	Решить систему уравнений
3	Олимпиада по математике	Математика	Найти производную функции
4	Олимпиада по математике	Математика	Решить интеграл
5	Олимпиада по математике	Математика	Доказать теорему
6	Олимпиада по математике	Математика	Найти корень уравнения
7	Олимпиада по математике	Математика	Решить задачу на комбинаторику
8	Олимпиада по математике	Математика	Решить задачу на алгебру
9	Олимпиада по математике	Математика	Составить математическую модель
10	Олимпиада по математике	Математика	Решить задачу на теорию вероятностей
11	Олимпиада по математике	Математика	Решить задачу на анализ
12	Олимпиада по математике	Математика	Найти гипотенузу треугольника
13	Олимпиада по математике	Математика	Решить задачу на дифференциальные уравнения
14	Олимпиада по математике	Физика	Рассчитать силу тяжести
15	Олимпиада по математике	Физика	Решить задачу на динамику
16	Олимпиада по математике	Физика	Определить плотность материала
17	Олимпиада по математике	Физика	Решить задачу на электродинамику

```
select * from OlympiadParticipantDetails
```

110 %

Результаты Сообщения

	OlympiadName	FirstName	LastName	DateOfBirth	Gender	SchoolName
1	Олимпиада по математике	Иван	Иванов	2000-05-15	М	Школа №1
2	Олимпиада по математике	Мария	Петрова	2001-08-22	Ж	Школа №2
3	Олимпиада по математике	Алексей	Смирнов	1999-11-10	М	Школа №3
4	Олимпиада по физике	Екатерина	Козлова	2000-03-08	Ж	Школа №4
5	Олимпиада по физике	Дмитрий	Соколов	2002-06-25	М	Школа №5
6	Олимпиада по физике	Анна	Иванова	1998-12-17	Ж	Школа №6
7	Олимпиада по математике	Павел	Морозов	2001-02-14	М	Школа №7
8	Олимпиада по физике	Наталья	Петрова	2002-10-30	Ж	Школа №8
9	Олимпиада по математике	Владимир	Смирнов	1997-09-05	М	Школа №9
10	Олимпиада по физике	Ольга	Козлова	1999-04-03	Ж	Школа №10
11	Олимпиада по математике	Александр	Соколов	2003-07-12	М	Школа №11
12	Олимпиада по физике	Ирина	Иванова	1996-11-20	Ж	Школа №12
13	Олимпиада по математике	Евгений	Морозов	2000-01-28	М	Школа №13

Рис. 6 - Представление 3

```
select * from OlympiadTasks
```

110 %

Результаты Сообщения

	OlympiadName	SubjectName	TaskDescription
1	Олимпиада по математике	Математика	Решить задачу на геометрию
2	Олимпиада по математике	Математика	Решить систему уравнений
3	Олимпиада по математике	Математика	Найти производную функции
4	Олимпиада по математике	Математика	Решить интеграл
5	Олимпиада по математике	Математика	Доказать теорему
6	Олимпиада по математике	Математика	Найти корень уравнения
7	Олимпиада по математике	Математика	Решить задачу на комбинаторику
8	Олимпиада по математике	Математика	Решить задачу на алгебру
9	Олимпиада по математике	Математика	Составить математическую модель
10	Олимпиада по математике	Математика	Решить задачу на теорию вероятностей
11	Олимпиада по математике	Математика	Решить задачу на анализ
12	Олимпиада по математике	Математика	Найти гипотенузу треугольника
13	Олимпиада по математике	Математика	Решить задачу на дифференциальные уравнения
14	Олимпиада по математике	Физика	Рассчитать силу тяжести
15	Олимпиада по математике	Физика	Решить задачу на динамику
16	Олимпиада по математике	Физика	Определить плотность материала
17	Олимпиада по математике	Физика	Решить задачу на электродинамику

Рис. 7 - Представление 4

select * from ComprehensiveResults				
110 %				
<div> <div>Результаты</div> <div>Сообщения</div> </div>				
	FirstName	LastName	OlympiadName	Score
1	Иван	Иванов	Олимпиада по математике	95
2	Мария	Петрова	Олимпиада по математике	88
3	Алексей	Смирнов	Олимпиада по математике	92
4	Екатерина	Козлова	Олимпиада по физике	78
5	Дмитрий	Соколов	Олимпиада по физике	85
6	Анна	Иванова	Олимпиада по физике	90
7	Павел	Морозов	Олимпиада по математике	94
8	Наталья	Петрова	Олимпиада по физике	82
9	Владимир	Смирнов	Олимпиада по математике	96
10	Ольга	Козлова	Олимпиада по физике	87
11	Александр	Соколов	Олимпиада по математике	91
12	Ирина	Иванова	Олимпиада по физике	89
13	Евгений	Морозов	Олимпиада по математике	93

Рис. 8 - Представление 5

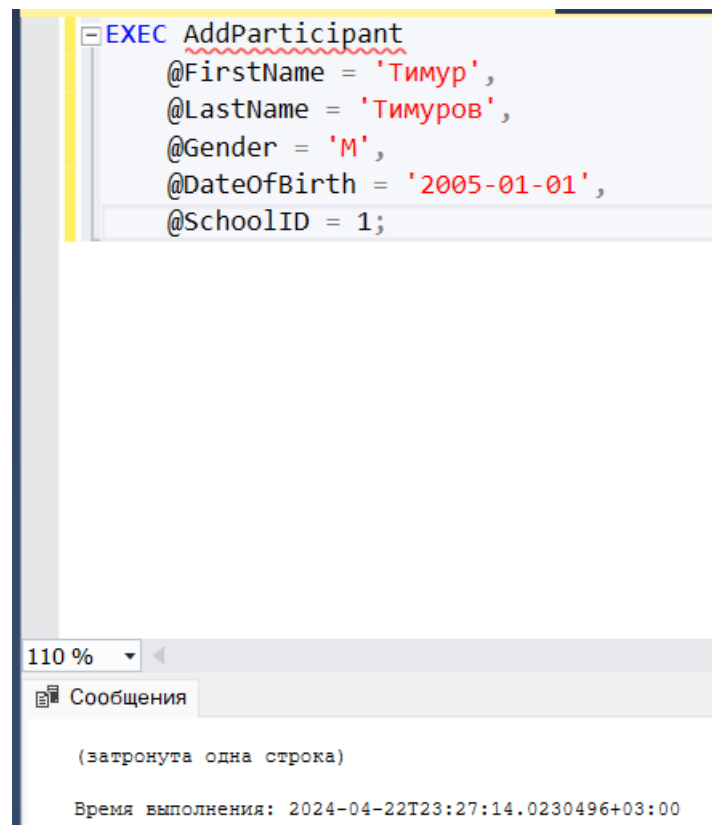


Рис. 9 - Процедура 1

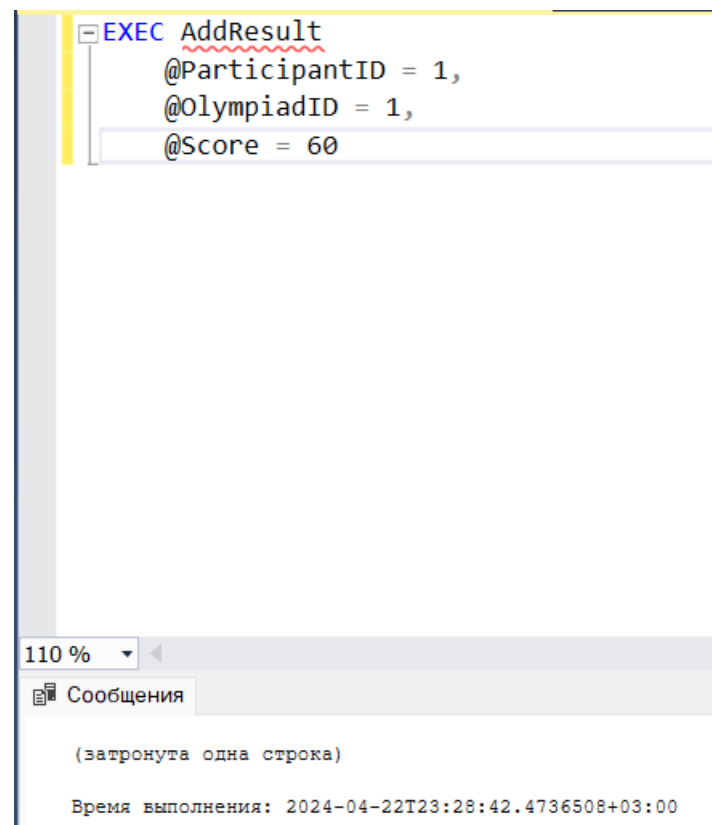


Рис. 10 - Процедура 2

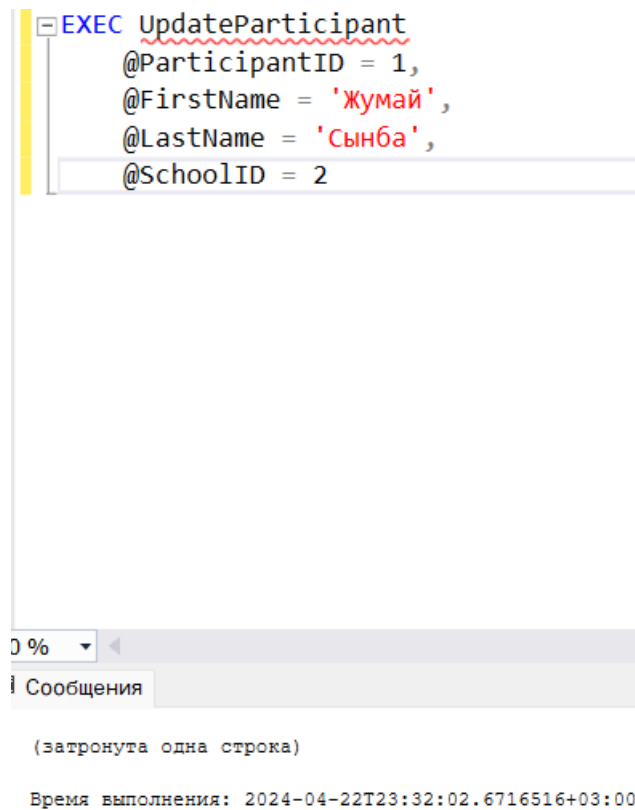


Рис. 11 - Процедура 3

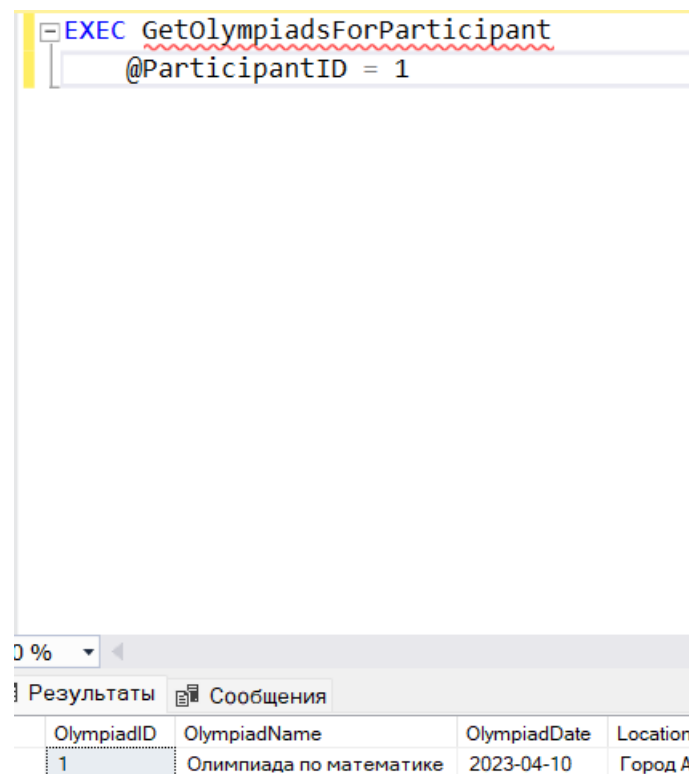


Рис. 12 - Процедура 4

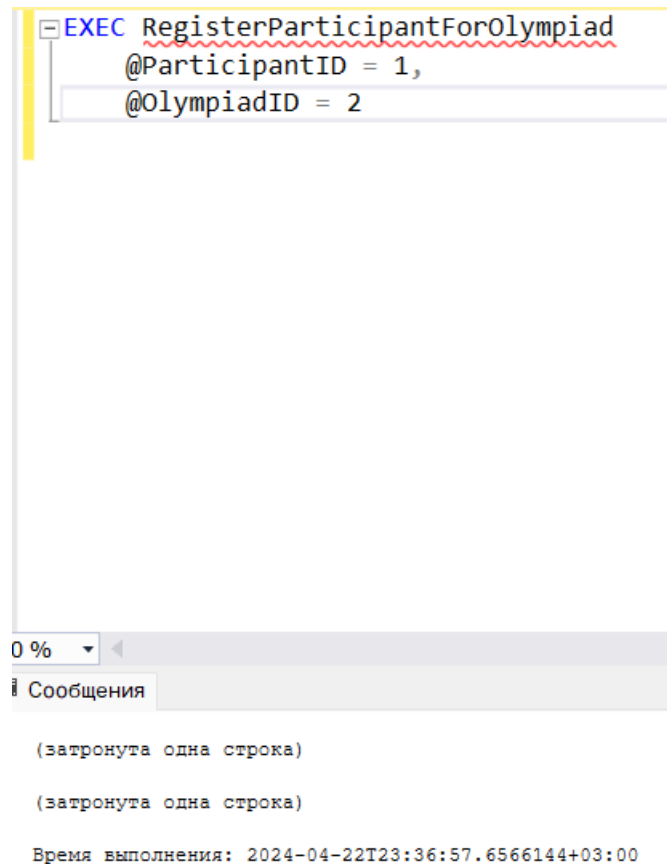


Рис. 13 - Процедура 5

Вывод:

В ходе выполнения лабораторной работы я познакомился с представлениями, триггерами и хранимыми процедурами, а также научился их создавать и выполнять.

ЛИСТИНГ

Представления

```
CREATE VIEW OlympiadParticipantsCount AS  
SELECT o.OlympiadName, COUNT(op.ParticipantID) as  
ParticipantCount  
FROM Olympiads o  
JOIN OlympiadParticipants op ON o.OlympiadID = op.OlympiadID  
GROUP BY o.OlympiadName;
```

```
CREATE VIEW SchoolResults AS  
SELECT s.SchoolName, AVG(r.Score) AS AvgScore, MAX(r.Score) AS  
MaxScore, MIN(r.Score) AS MinScore  
FROM Schools s  
JOIN Participants p ON s.SchoolID = p.SchoolID  
JOIN Results r ON p.ParticipantID = r.ParticipantID  
GROUP BY s.SchoolName;
```

```
CREATE VIEW OlympiadParticipantDetails AS  
SELECT o.OlympiadName, p.FirstName, p.LastName, p.DateOfBirth,  
p.Gender, s.SchoolName  
FROM OlympiadParticipants op  
JOIN Participants p ON op.ParticipantID = p.ParticipantID  
JOIN Olympiads o ON op.OlympiadID = o.OlympiadID  
JOIN Schools s ON p.SchoolID = s.SchoolID;
```

```
CREATE VIEW OlympiadTasks AS  
SELECT o.OlympiadName, sub.SubjectName, t.TaskDescription  
FROM Tasks t  
JOIN Olympiads o ON t.OlympiadID = o.OlympiadID  
JOIN Subjects sub ON t.SubjectID = sub.SubjectID;
```

```
CREATE VIEW ComprehensiveResults AS  
SELECT p.FirstName, p.LastName, o.OlympiadName, r.Score  
FROM Results r  
JOIN Participants p ON r.ParticipantID = p.ParticipantID  
JOIN Olympiads o ON r.OlympiadID = o.OlympiadID;
```

Триггеры

CREATE TRIGGER CheckAge

ON Participants

AFTER INSERT

AS

BEGIN

IF EXISTS (SELECT 1 FROM inserted WHERE DATEDIFF(year, inserted.DateOfBirth, GETDATE()) < 10)

BEGIN

RAISERROR ('Participant is too young.', 16, 1);

ROLLBACK TRANSACTION;

END

END;

CREATE TRIGGER DeleteResultsOnParticipantDeletion

ON Participants

AFTER DELETE

AS

BEGIN

DELETE FROM Results

WHERE ParticipantID IN (SELECT ParticipantID FROM deleted);

END;

CREATE TRIGGER RemoveOlympiadParticipantOnParticipantDelete

ON Participants

AFTER DELETE

AS

BEGIN

```
DELETE FROM OlympiadParticipants  
WHERE ParticipantID IN (SELECT ParticipantID FROM deleted);  
END;
```

```
CREATE TRIGGER ValidateOlympiadParticipantInsert  
ON OlympiadParticipants  
INSTEAD OF INSERT  
AS  
BEGIN  
    IF EXISTS (  
        SELECT 1  
        FROM inserted  
        WHERE NOT EXISTS (SELECT 1 FROM Participants WHERE  
ParticipantID = inserted.ParticipantID)  
        OR NOT EXISTS (SELECT 1 FROM Olympiads WHERE  
OlympiadID = inserted.OlympiadID)  
    )  
    BEGIN  
        RAISERROR ('Participant or Olympiad does not exist.', 16, 1);  
    END  
    ELSE  
    BEGIN  
        INSERT INTO OlympiadParticipants (OlympiadParticipantID,  
ParticipantID, OlympiadID)  
        SELECT OlympiadParticipantID, ParticipantID, OlympiadID  
        FROM inserted;  
    END  
END;
```

```
CREATE TRIGGER ValidateScoreUpdate  
ON Results  
INSTEAD OF INSERT  
AS  
BEGIN  
    IF EXISTS (  
        SELECT 1  
        FROM inserted  
        WHERE Score < 0 OR Score > 100  
    )  
        BEGIN  
            RAISERROR ('Score must be between 0 and 100.', 16, 1);  
            ROLLBACK TRANSACTION;  
        END  
END;
```

Пррцедуры

```
CREATE PROCEDURE AddParticipant  
    @FirstName NVARCHAR(100),  
    @LastName NVARCHAR(100),  
    @Gender CHAR(1),  
    @DateOfBirth DATE,  
    @SchoolID INT  
AS  
BEGIN  
    INSERT INTO Participants (ParticipantID, FirstName, LastName,  
    Gender, DateOfBirth, SchoolID)
```



```
VALUES ((SELECT COUNT(*) + 1 FROM Participants), @FirstName,  
@LastName, @Gender, @DateOfBirth, @SchoolID);  
END;
```

```
CREATE PROCEDURE AddResult
```

```
    @ParticipantID INT,
```

```
    @OlympiadID INT,
```

```
    @Score INT
```

```
AS
```

```
BEGIN
```

```
    INSERT INTO Results (ParticipantID, OlympiadID, Score)
```

```
    VALUES (@ParticipantID, @OlympiadID, @Score);
```

```
END;
```

```
CREATE PROCEDURE UpdateParticipant
```

```
    @ParticipantID INT,
```

```
    @FirstName NVARCHAR(100),
```

```
    @LastName NVARCHAR(100),
```

```
    @SchoolID INT
```

```
AS
```

```
BEGIN
```

```
    UPDATE Participants
```

```
    SET FirstName = @FirstName,
```

```
        LastName = @LastName,
```

```
        SchoolID = @SchoolID
```

```
    WHERE ParticipantID = @ParticipantID;
```

```
END;
```

CREATE PROCEDURE GetOlympiadsForParticipant

@ParticipantID INT

AS

BEGIN

SELECT o.OlympiadID, o.OlympiadName, o.OlympiadDate, o.Location

FROM Olympiads o

JOIN OlympiadParticipants op ON o.OlympiadID = op.OlympiadID

WHERE op.ParticipantID = @ParticipantID;

END;

CREATE PROCEDURE RegisterParticipantForOlympiad

@ParticipantID INT,

@OlympiadID INT

AS

BEGIN

**INSERT INTO OlympiadParticipants (OlympiadParticipantID,
ParticipantID, OlympiadID)**

**VALUES ((SELECT COUNT(*) + 1 FROM OlympiadParticipants),
@ParticipantID, @OlympiadID);**

END;