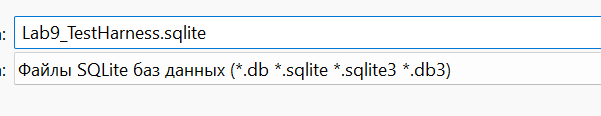
**Лабораторная работа №9**

**Тема:** Разработка тестовых модулей проекта для тестирования отдельных модулей

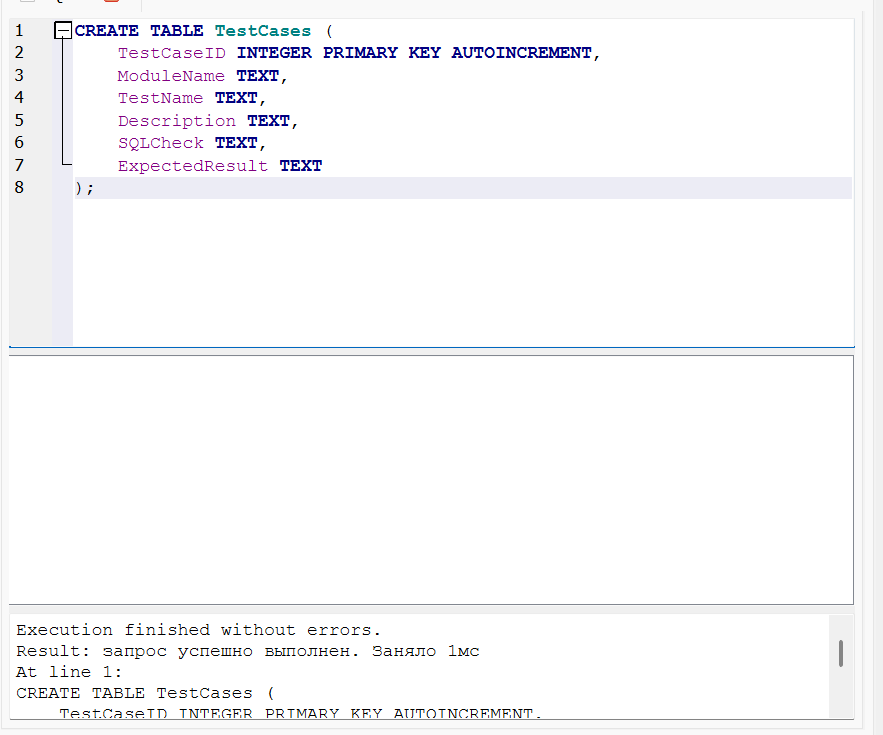
**Цель работы**

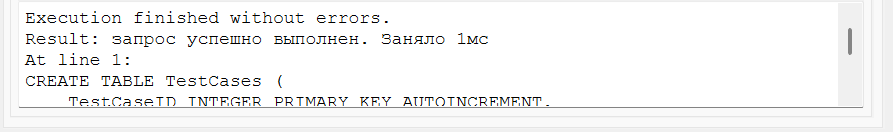
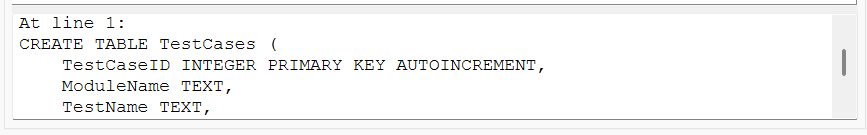
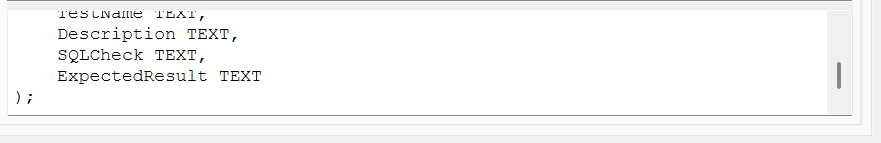
Научить студентов проектировать и реализовывать тестовые модули (test harness) для отдельных компонентов информационной системы: создавать таблицы логирования и тестов, писать SQL-проверки, автоматические VBA-тесты, а также визуализировать архитектуру и тестовые сценарии с помощью диаграмм в MS Visio.

**Шаг 1**: Создаем новую базу данных

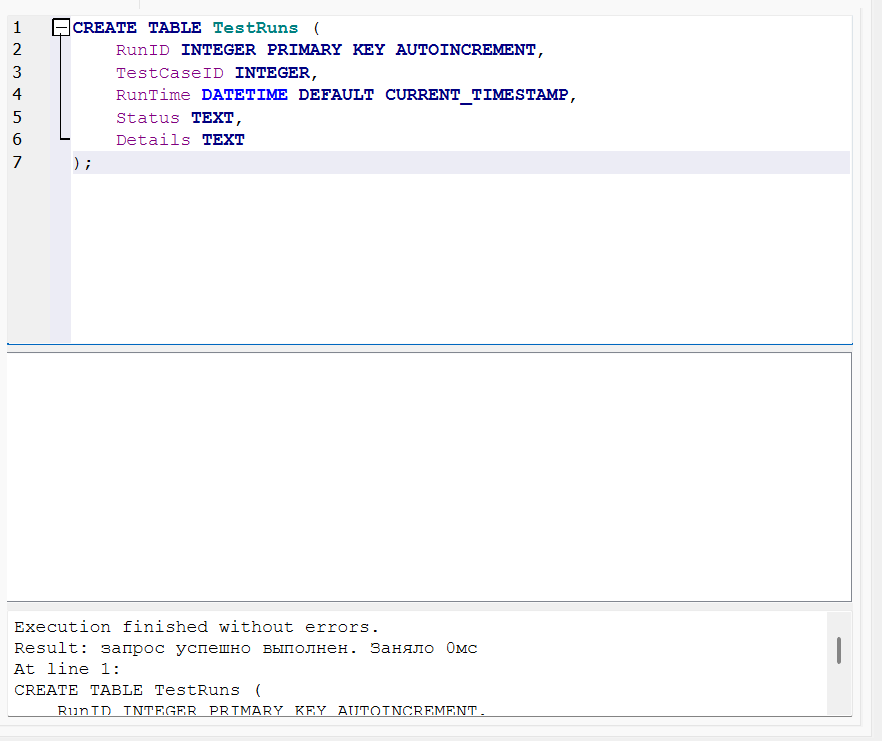
 

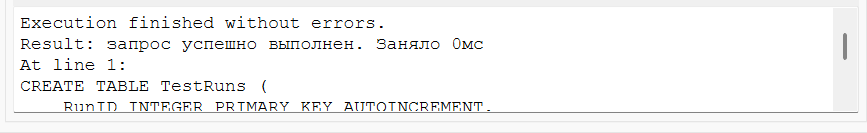
**Шаг 2**: Создаем таблицу TestCases

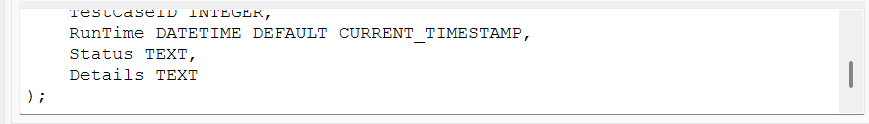


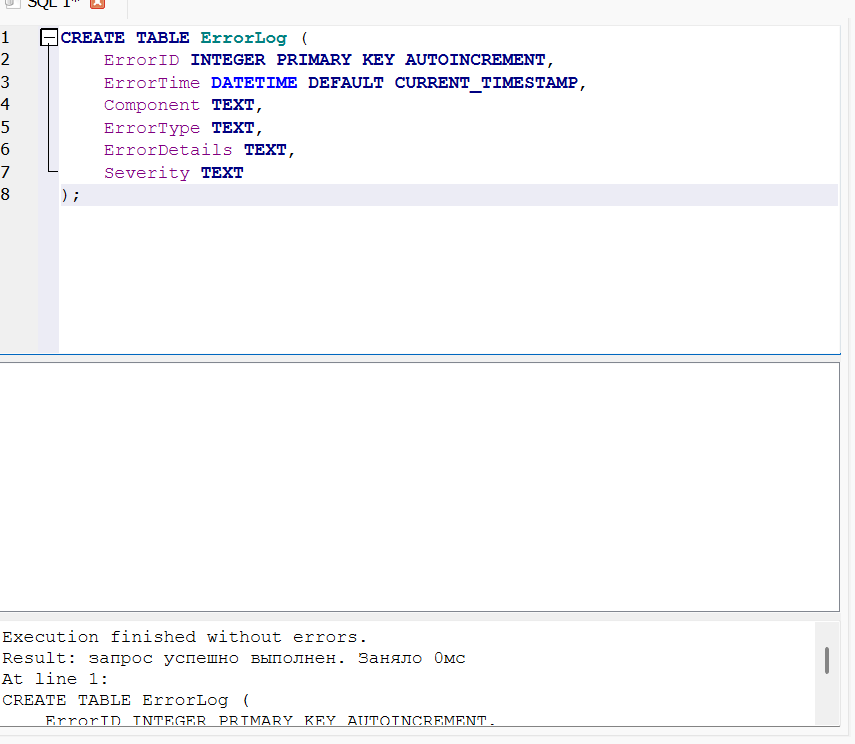
**Шаг 3**: Создаем таблицу TestRuns

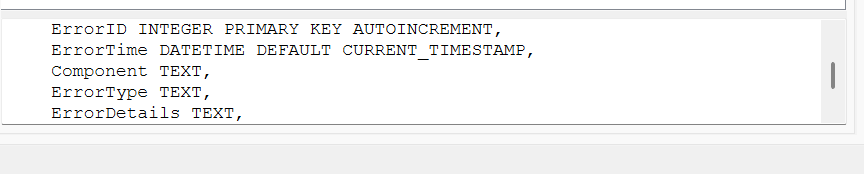


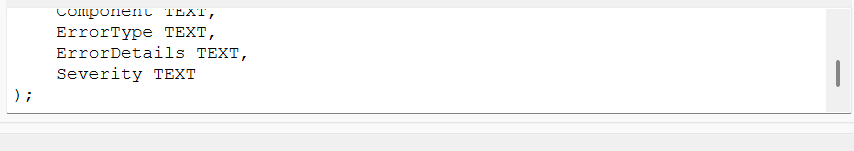




**Шаг 4**: Создаем таблицу ErrorLog

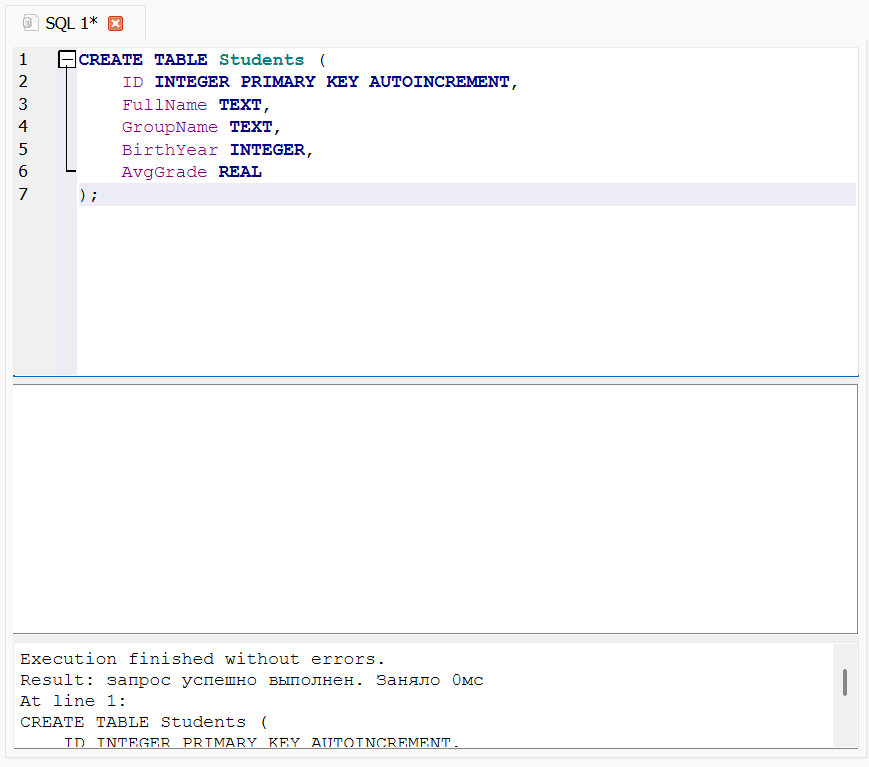


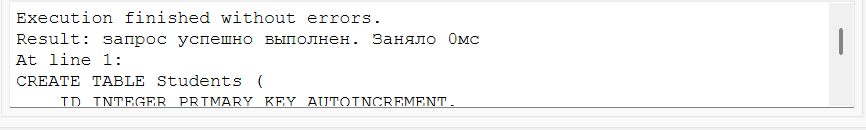


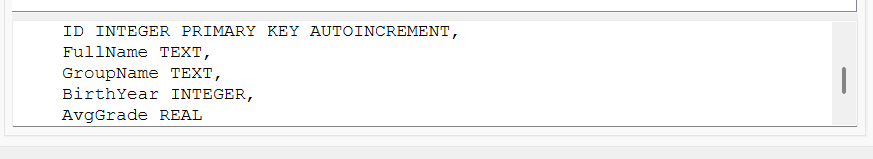


**ЗАДАНИЕ 1**: ТЕСТ-МОДУЛЬ ТАБЛИЦЫ STUDENTS

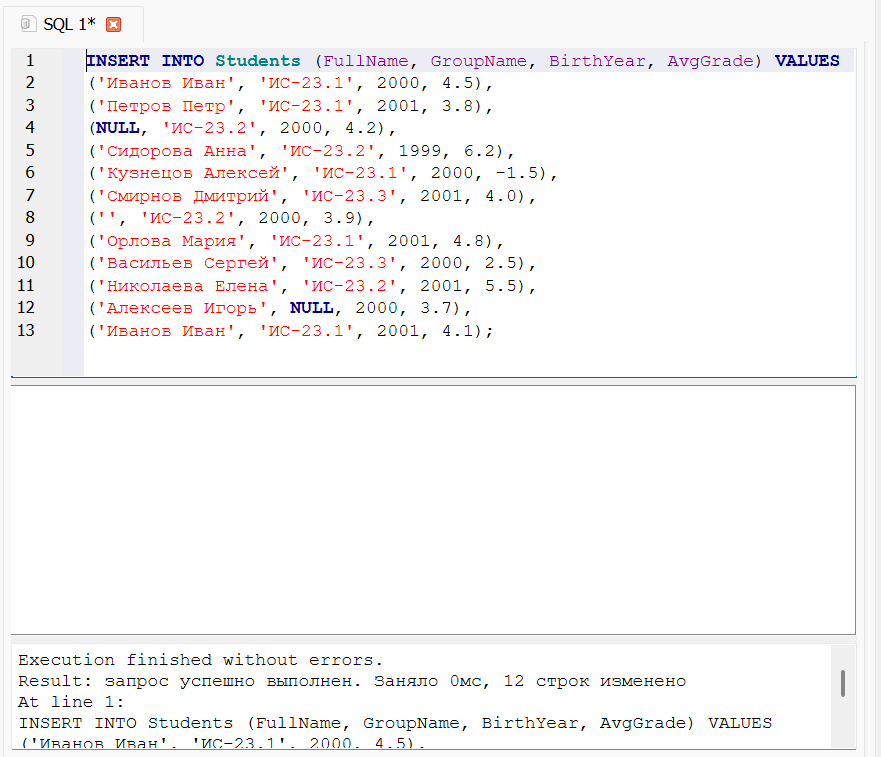
**Шаг 5**: Создаем таблицу Students

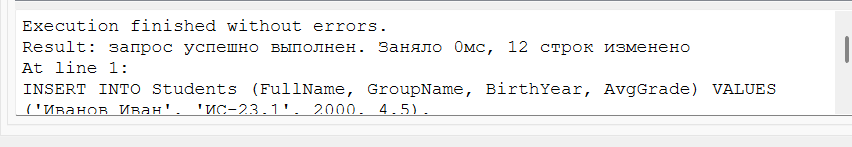


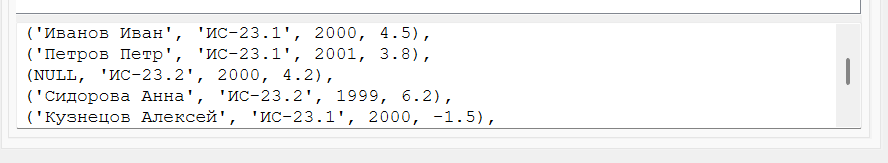


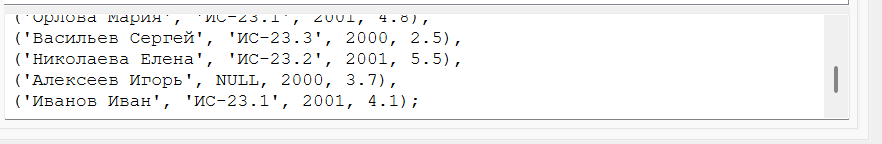


**Шаг 6**: Добавляем тестовые данные в Students

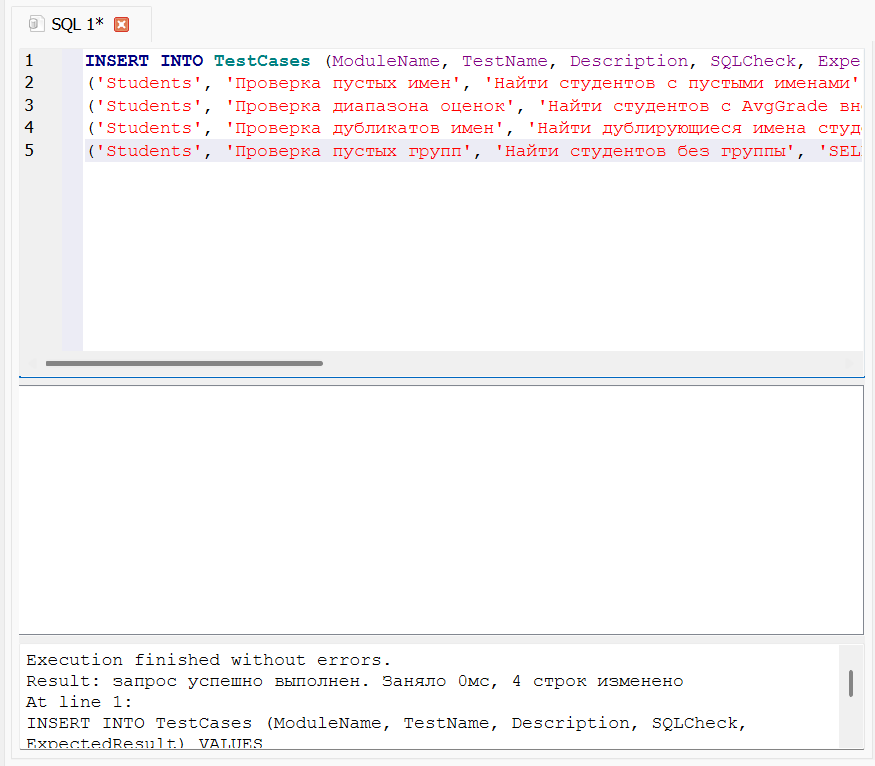


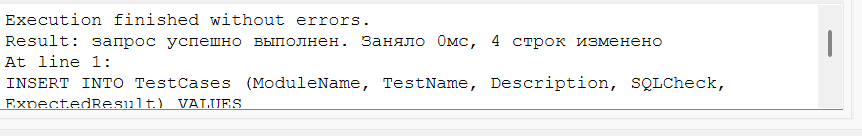


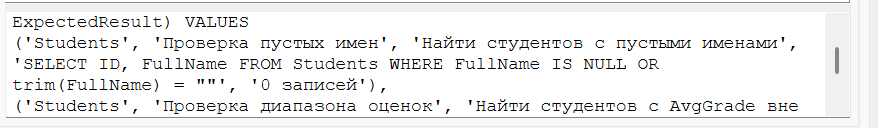


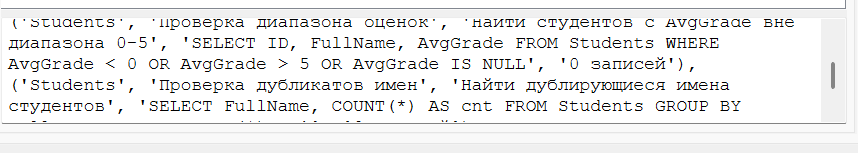


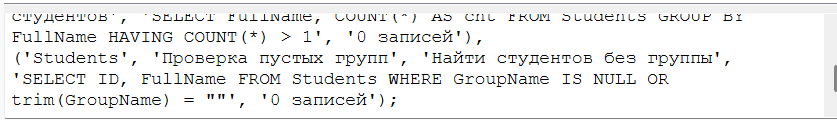
**Шаг 7**: Добавляем тест-кейсы в TestCases





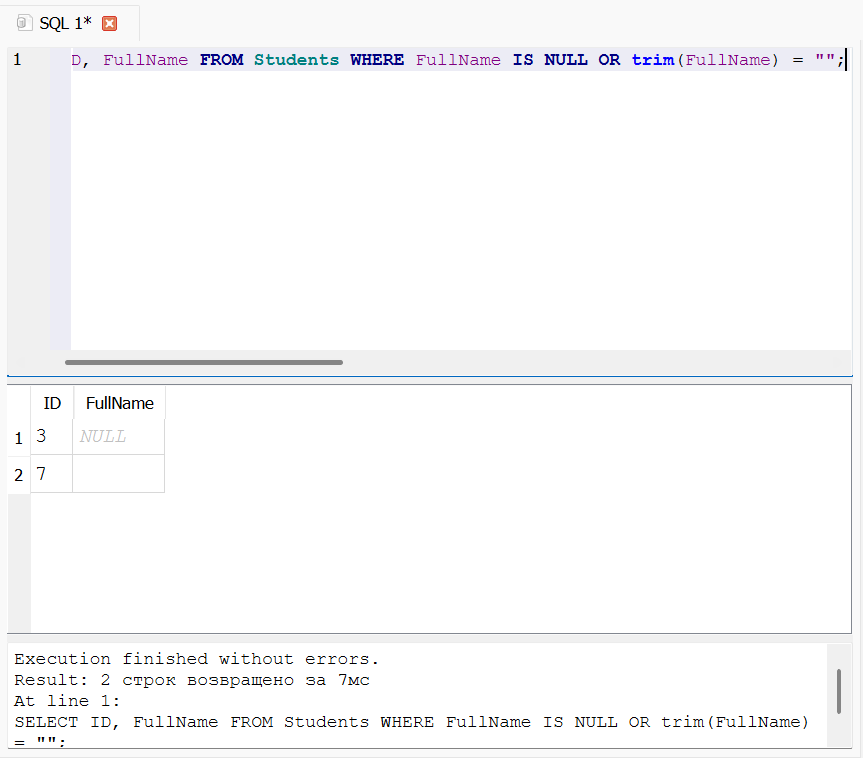


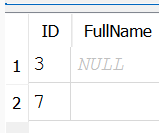


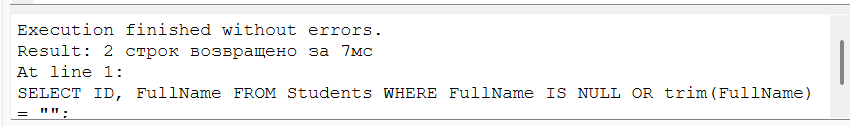


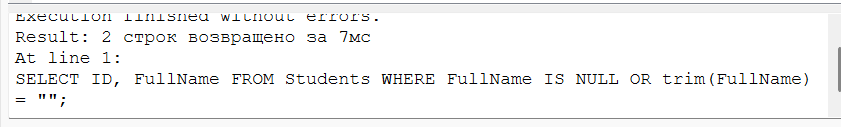
**Шаг 8**: Проверяем тест-кейсы вручную

первый тест вручную

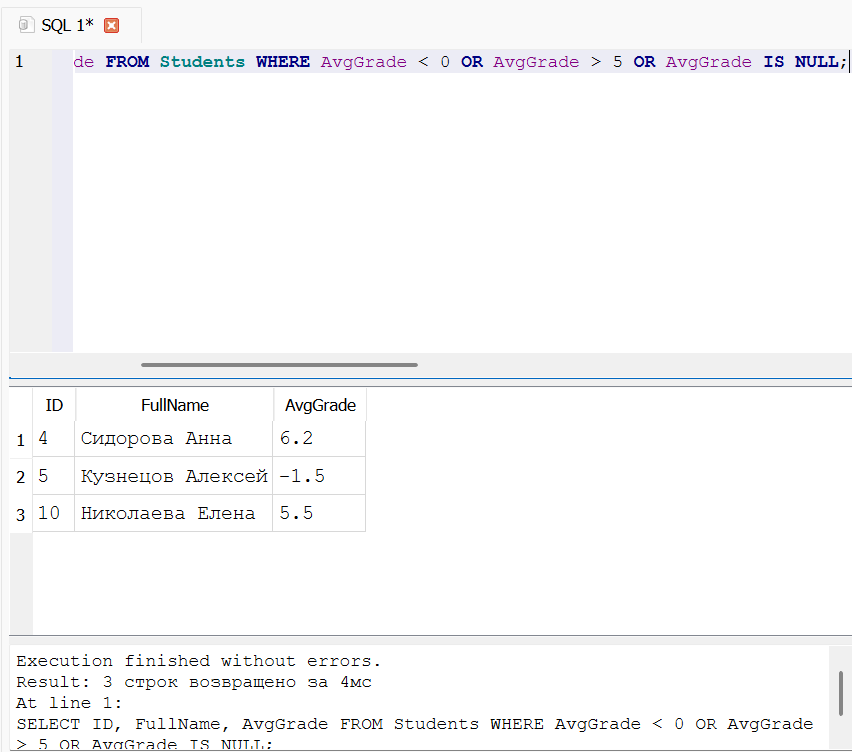


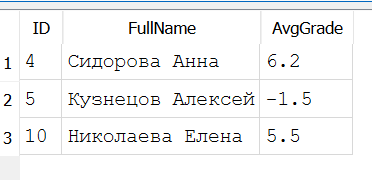


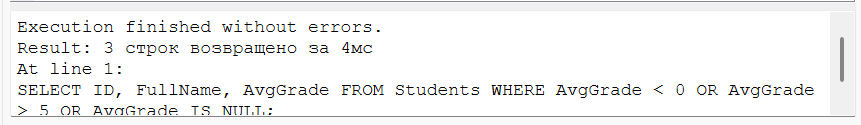




второй тест

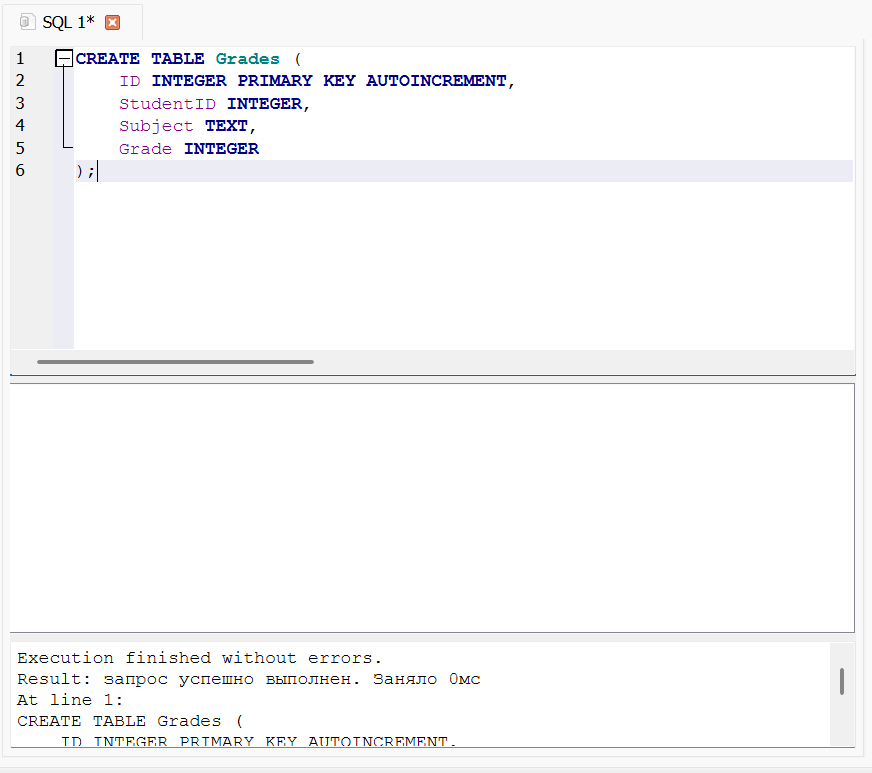


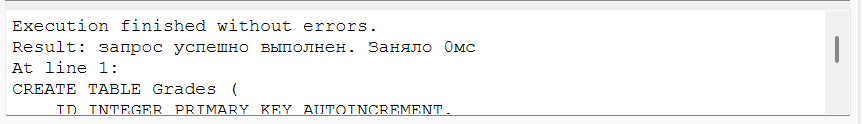


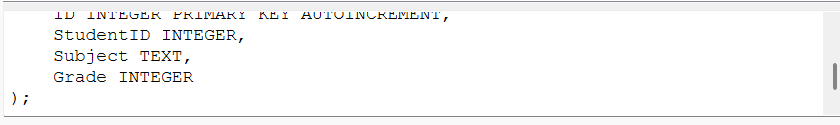


**ЗАДАНИЕ 2**: ТЕСТ-МОДУЛЬ СВЯЗЕЙ STUDENTS ↔ GRADES

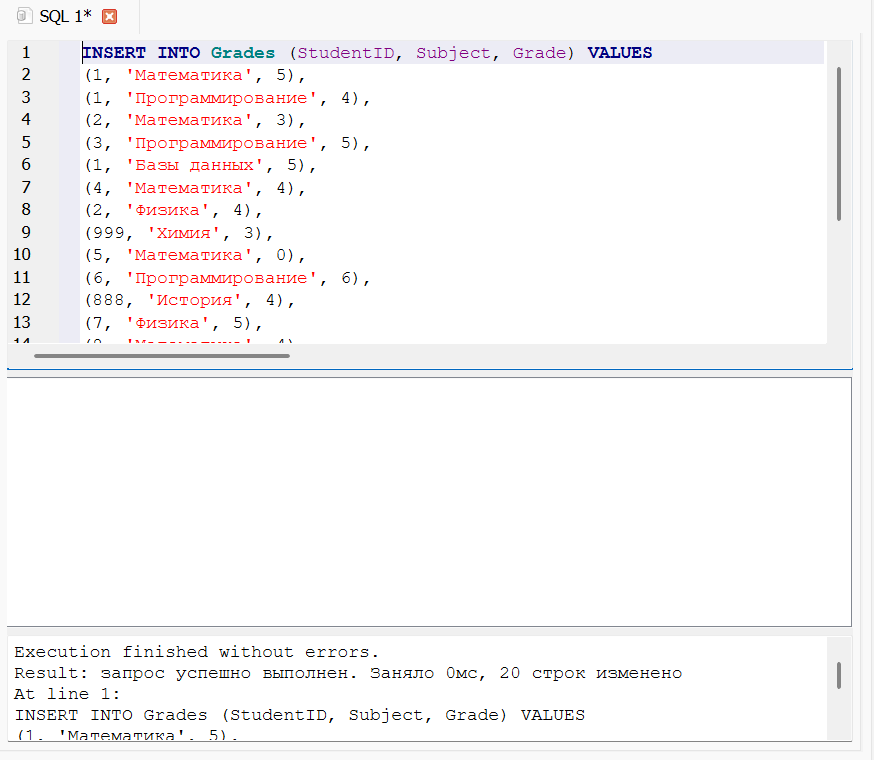
**Шаг 9**: Создаем таблицу Grades

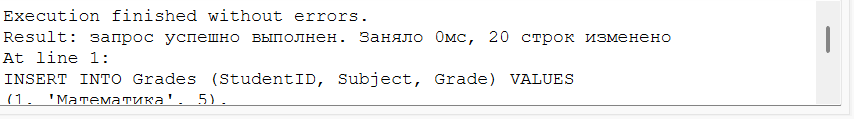


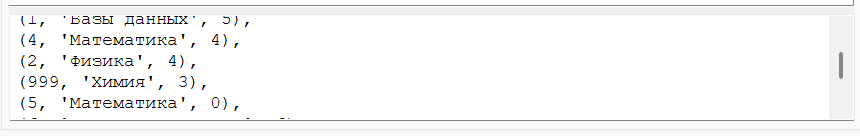




**Шаг 10**: Добавляем тестовые данные в Grades

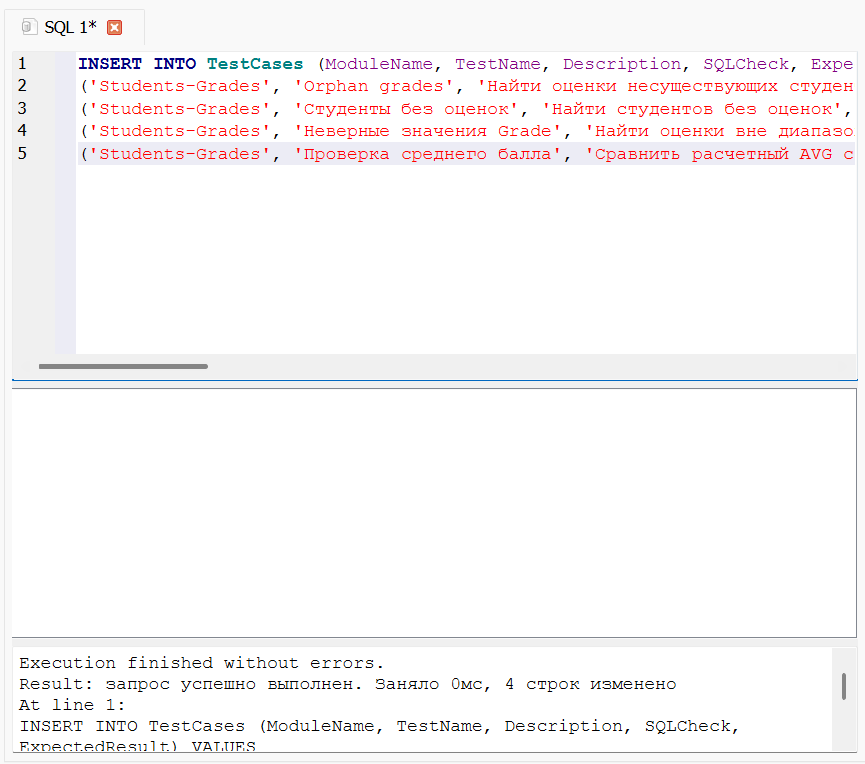
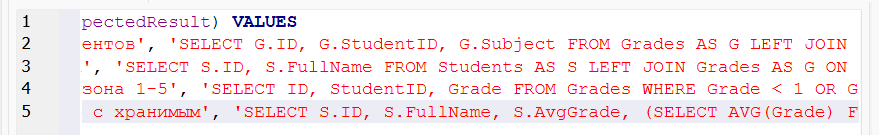
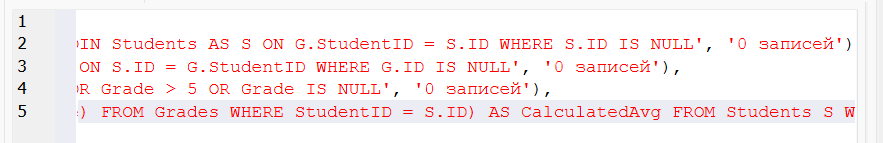
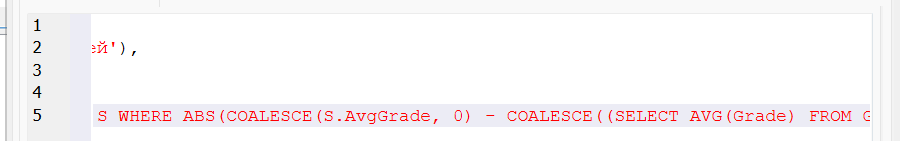


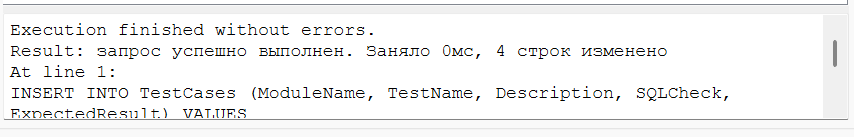
 

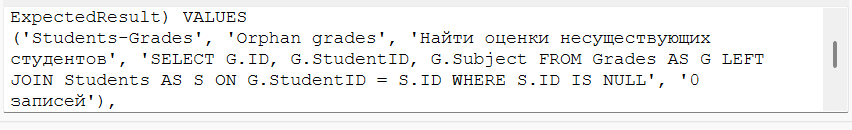




**Шаг 11**: Добавляем тест-кейсы для интеграции

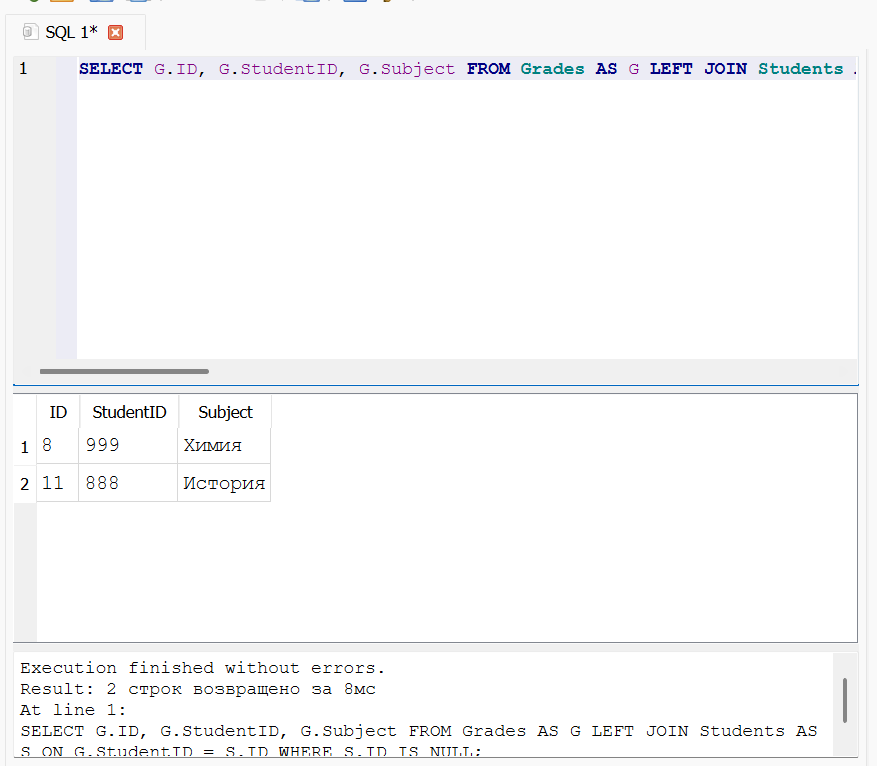


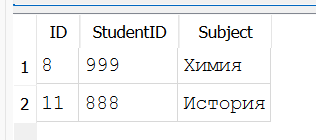
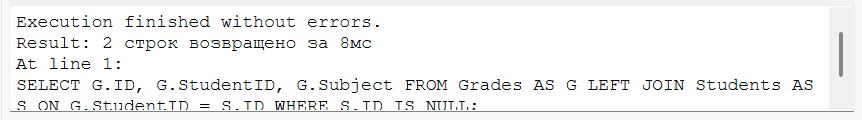
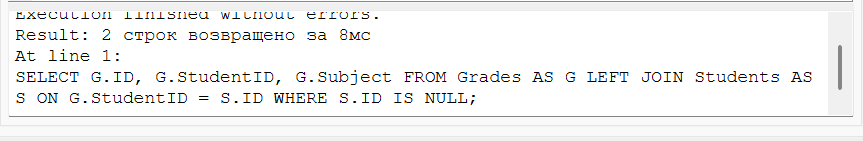


**Шаг 12**: Проверяем интеграционные тесты вручную

Проверяем orphan grades

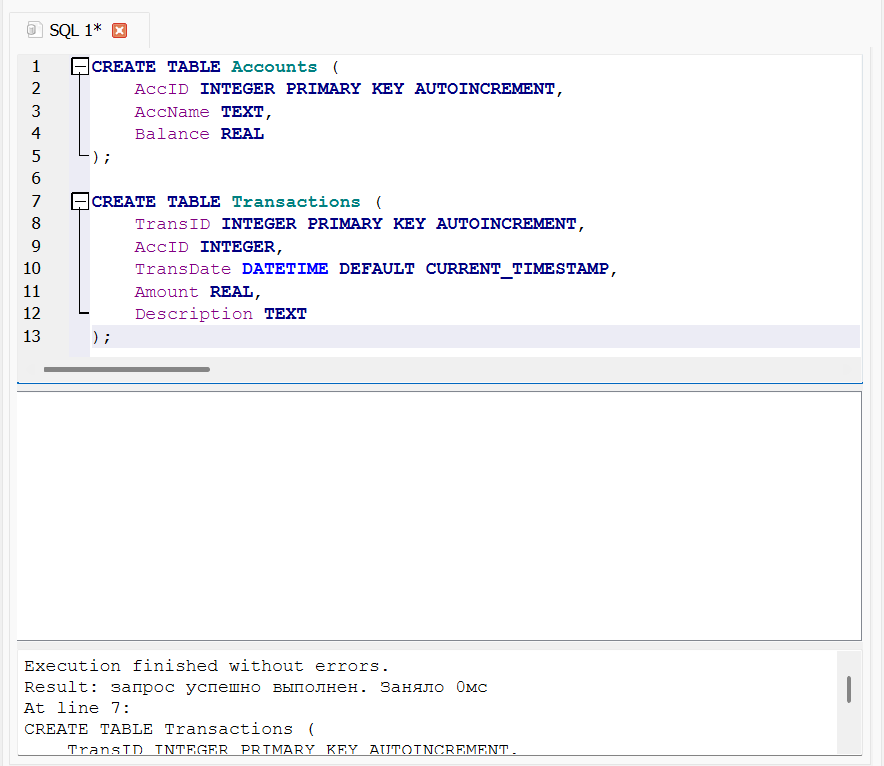
SELECT G.ID, G.StudentID, G.Subject FROM Grades AS G LEFT JOIN Students AS S ON G.StudentID = S.ID WHERE S.ID IS NULL;

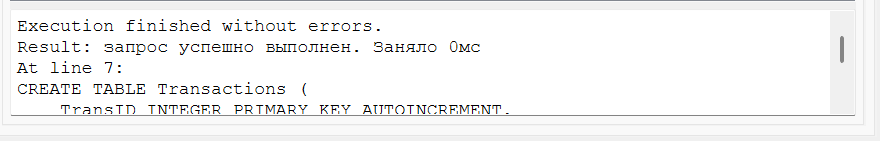


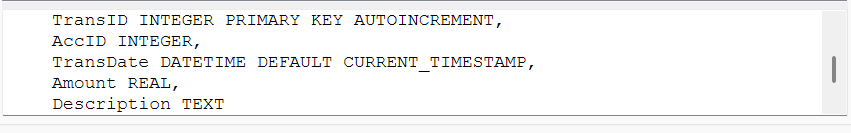
  

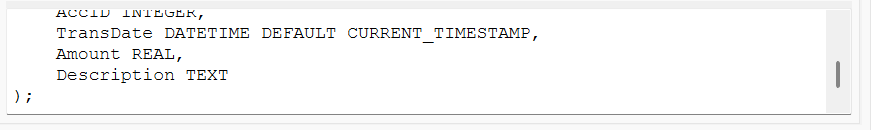
**ЗАДАНИЕ 3**: ТЕСТ-МОДУЛЬ ТРАНЗАКЦИЙ

**Шаг 13**: Создаем таблицы Accounts и Transactions

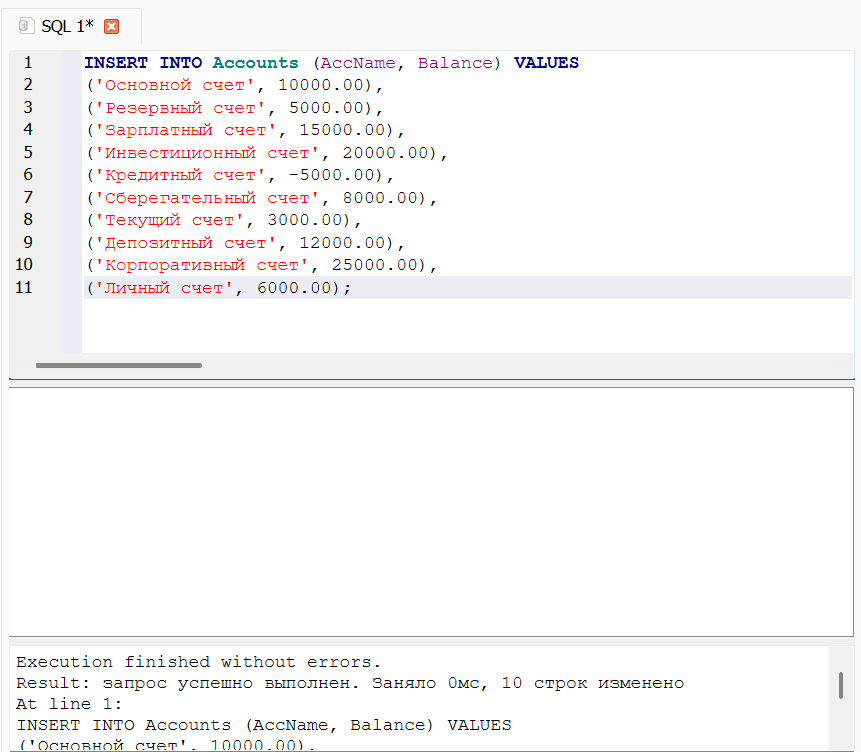


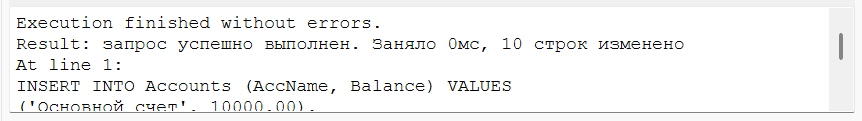


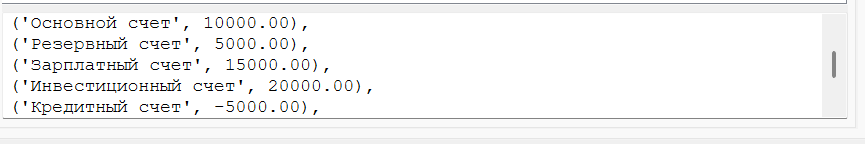




**Шаг 14**: Добавляем тестовые данные









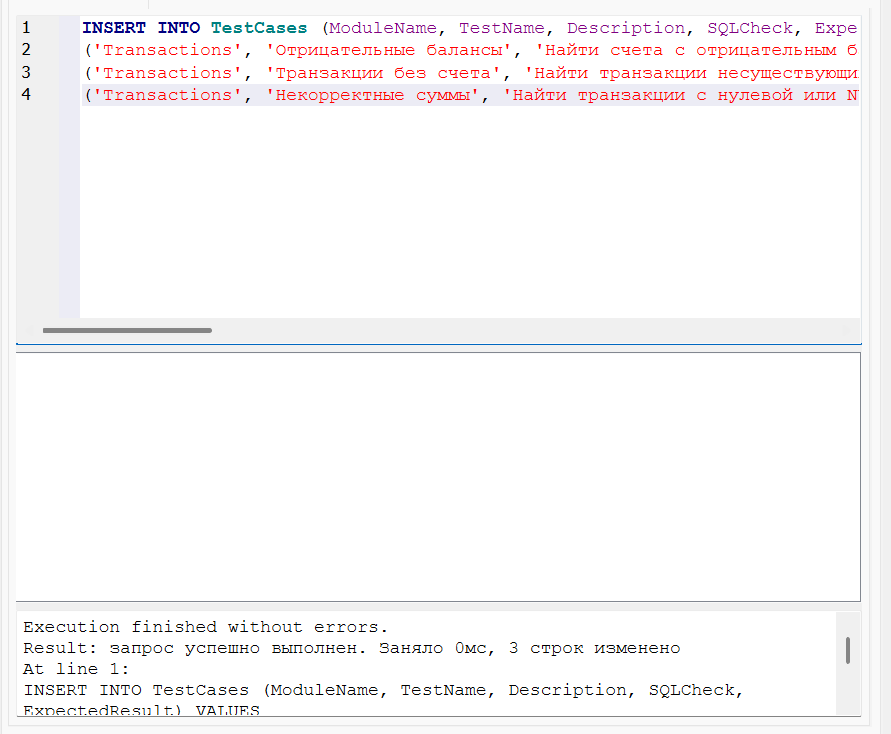
**Шаг 15**: Добавляем тест-кейсы для транзакций

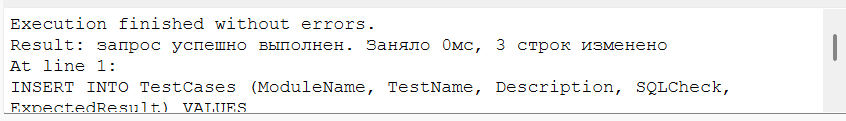
INSERT INTO TestCases (ModuleName, TestName, Description, SQLCheck, ExpectedResult) VALUES

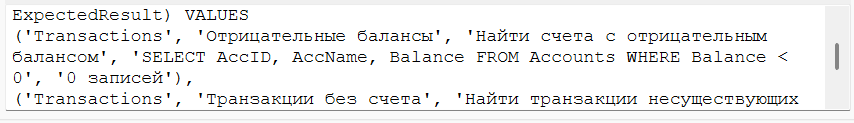
('Transactions', 'Отрицательные балансы', 'Найти счета с отрицательным балансом', 'SELECT AccID, AccName, Balance FROM Accounts WHERE Balance < 0', '0 записей'),

('Transactions', 'Транзакции без счета', 'Найти транзакции несуществующих счетов', 'SELECT T.TransID, T.AccID FROM Transactions AS T LEFT JOIN Accounts AS A ON T.AccID = A.AccID WHERE A.AccID IS NULL', '0 записей'),

('Transactions', 'Некорректные суммы', 'Найти транзакции с нулевой или NULL суммой', 'SELECT TransID, AccID, Amount FROM Transactions WHERE Amount = 0 OR Amount IS NULL', '0 записей');

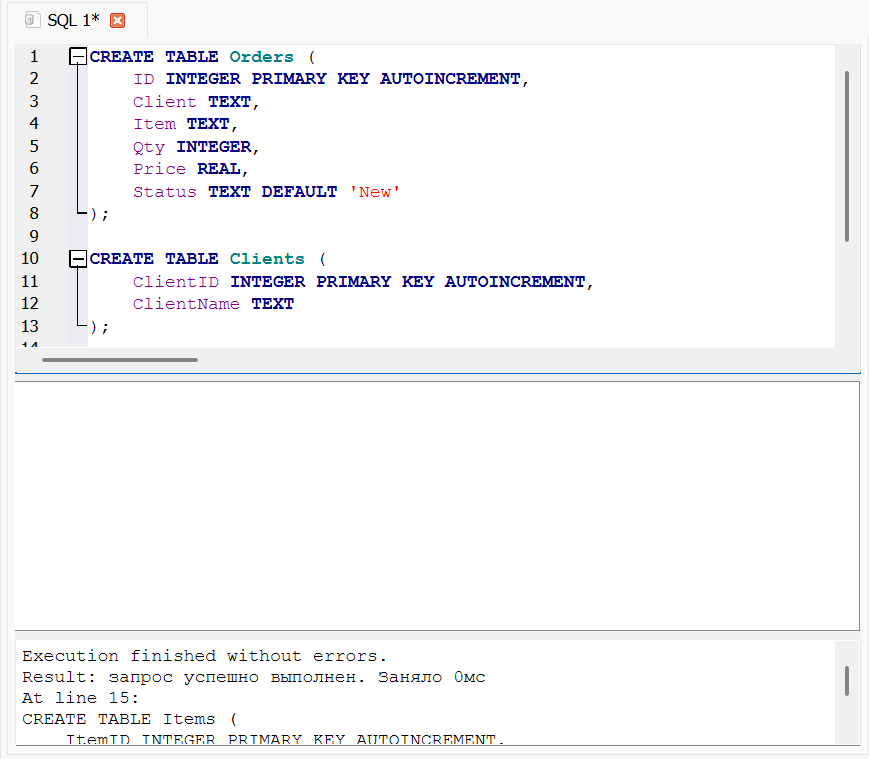
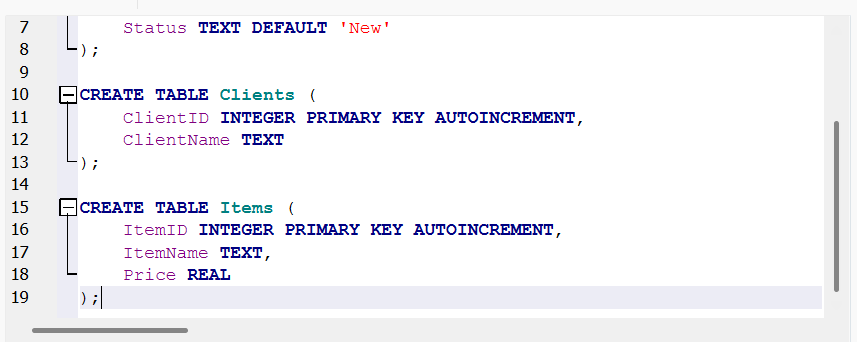


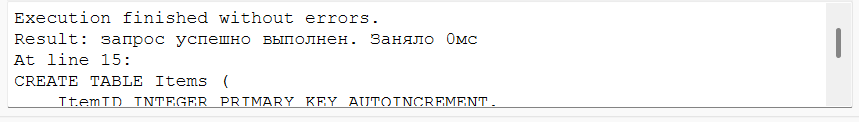
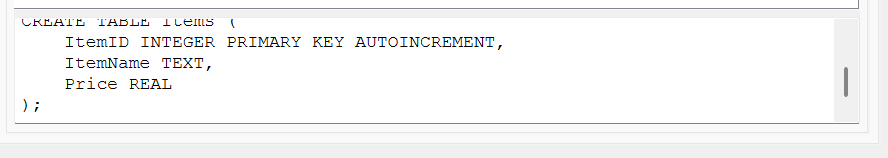




**ЗАДАНИЕ 4**: ТЕСТИРОВАНИЕ UI-МОДУЛЯ

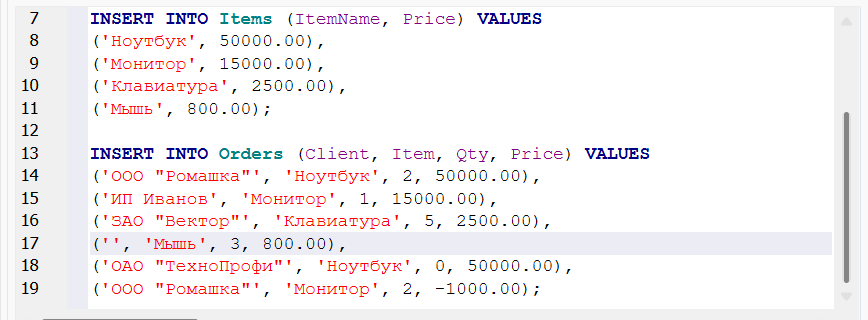
**Шаг 16**: Создаем таблицы для заказов

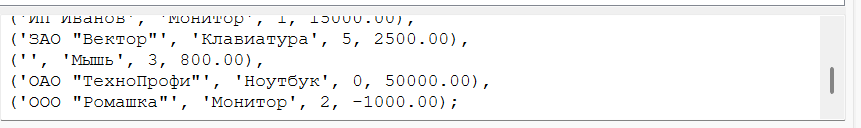
 

**Шаг 17**: Добавляем тестовые данные







**Шаг 18**: Добавляем тест-кейсы для заказов

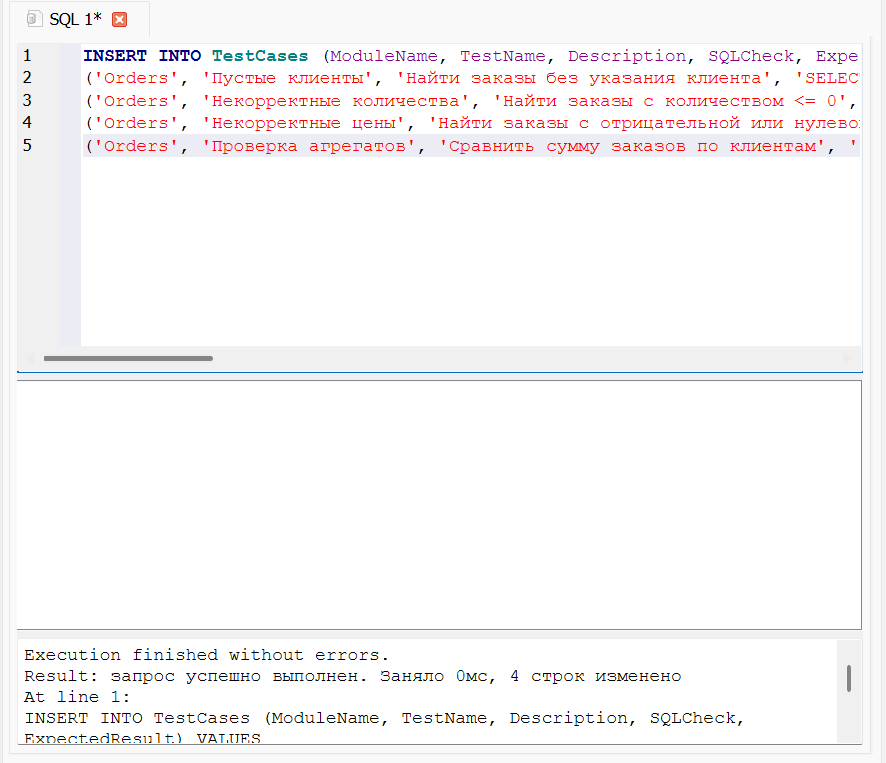
INSERT INTO TestCases (ModuleName, TestName, Description, SQLCheck, ExpectedResult) VALUES

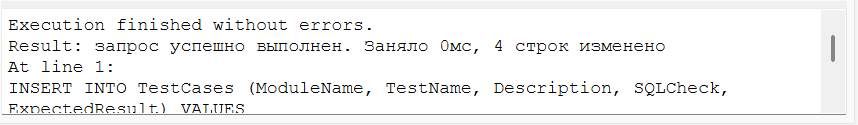
('Orders', 'Пустые клиенты', 'Найти заказы без указания клиента', 'SELECT ID, Client, Item FROM Orders WHERE Client IS NULL OR trim(Client) = ""', '0 записей'),

('Orders', 'Некорректные количества', 'Найти заказы с количеством <= 0', 'SELECT ID, Client, Item, Qty FROM Orders WHERE Qty <= 0 OR Qty IS NULL', '0 записей'),

('Orders', 'Некорректные цены', 'Найти заказы с отрицательной или нулевой ценой', 'SELECT ID, Client, Item, Price FROM Orders WHERE Price <= 0 OR Price IS NULL', '0 записей'),

('Orders', 'Проверка агрегатов', 'Сравнить сумму заказов по клиентам', 'SELECT Client, SUM(Qty \* Price) AS TotalAmount FROM Orders GROUP BY Client HAVING TotalAmount <= 0', '0 записей');





**ШАГ 19**: РУЧНОЕ ТЕСТИРОВАНИЕ ВСЕХ ТЕСТ-КЕЙСОВ

Проверяем все тест-кейсы по очереди

тесты для Students

-- Тест 1: Пустые имена

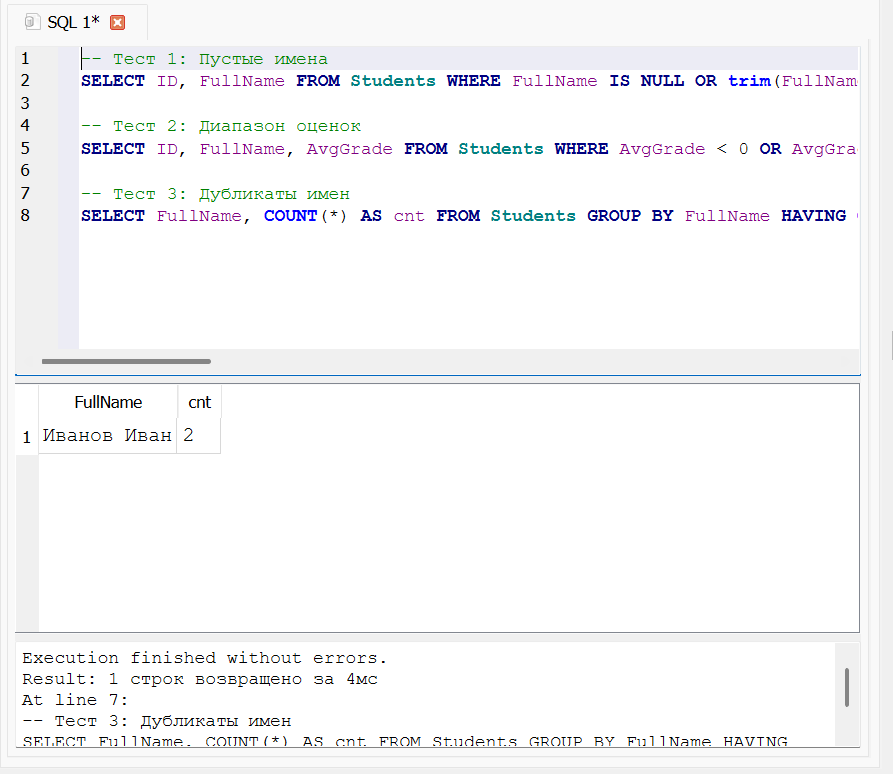
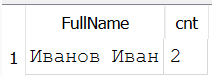
SELECT ID, FullName FROM Students WHERE FullName IS NULL OR trim(FullName) = "";

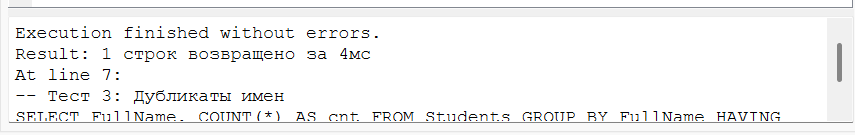
-- Тест 2: Диапазон оценок

SELECT ID, FullName, AvgGrade FROM Students WHERE AvgGrade < 0 OR AvgGrade > 5 OR AvgGrade IS NULL;

-- Тест 3: Дубликаты имен

SELECT FullName, COUNT(\*) AS cnt FROM Students GROUP BY FullName HAVING COUNT(\*) > 1;



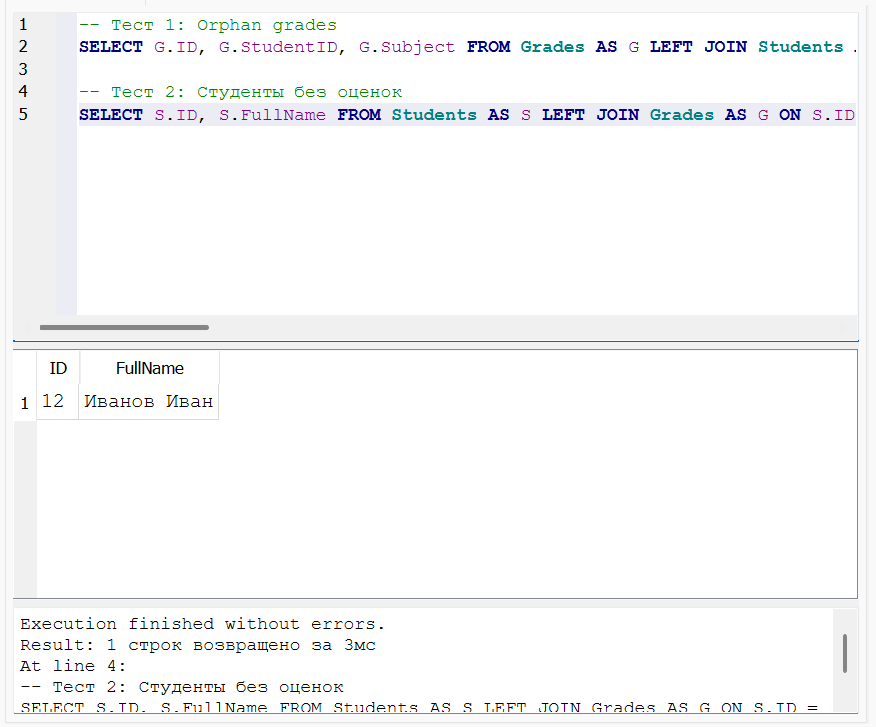
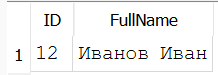
**тесты для Students-Grades**

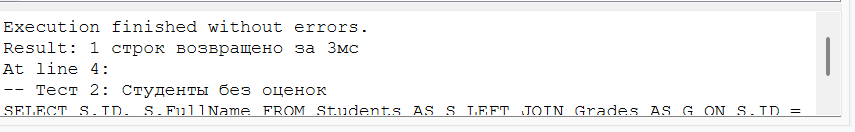
-- Тест 1: Orphan grades

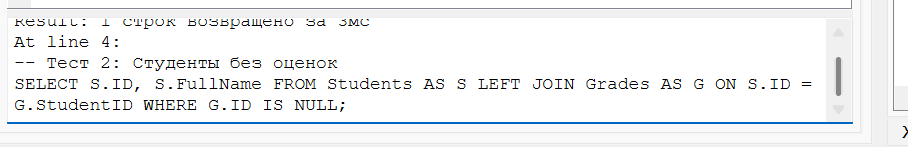
SELECT G.ID, G.StudentID, G.Subject FROM Grades AS G LEFT JOIN Students AS S ON G.StudentID = S.ID WHERE S.ID IS NULL;

-- Тест 2: Студенты без оценок

SELECT S.ID, S.FullName FROM Students AS S LEFT JOIN Grades AS G ON S.ID = G.StudentID WHERE G.ID IS NULL;

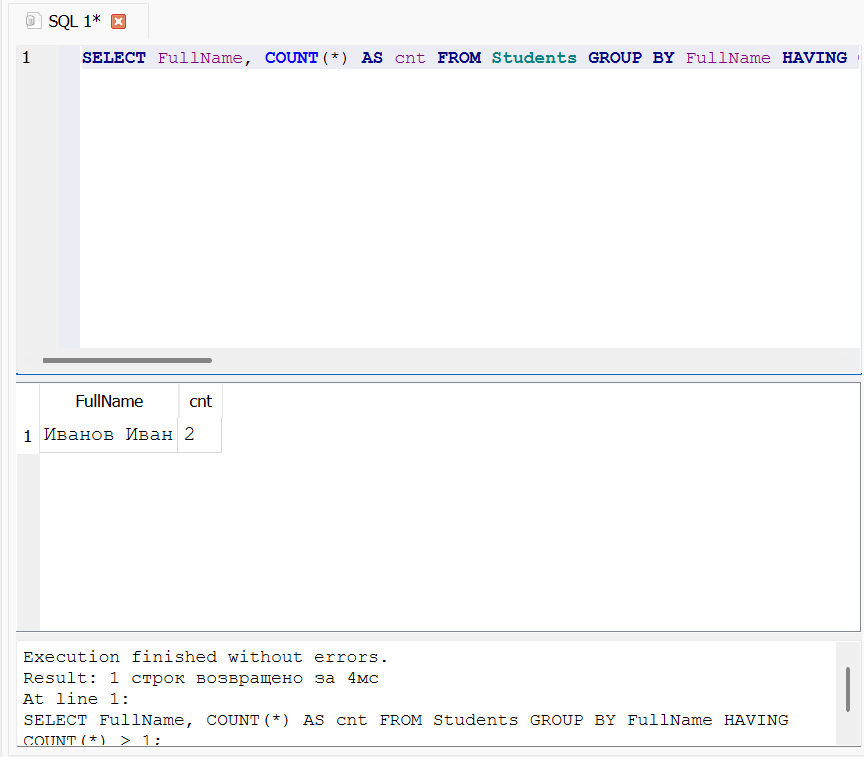
 

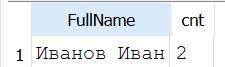


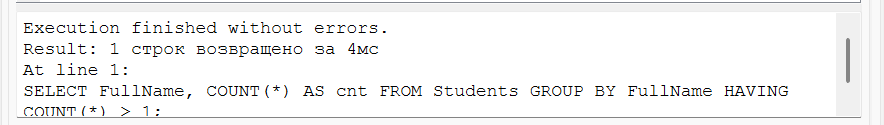


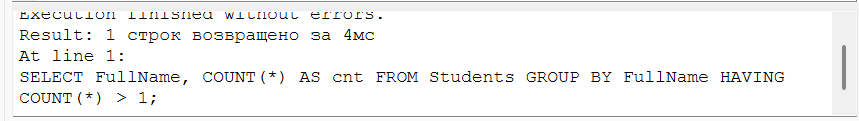
**Тест 3**: Дубликаты имен (Students)

SELECT FullName, COUNT(\*) AS cnt FROM Students GROUP BY FullName HAVING COUNT(\*) > 1;



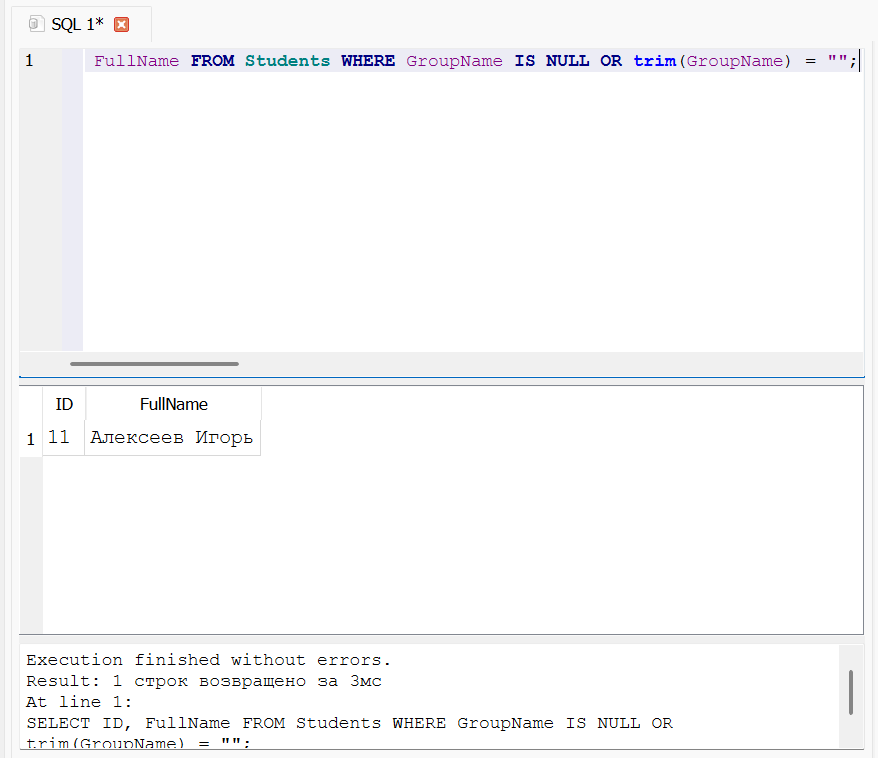


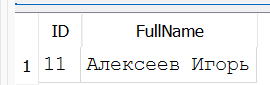


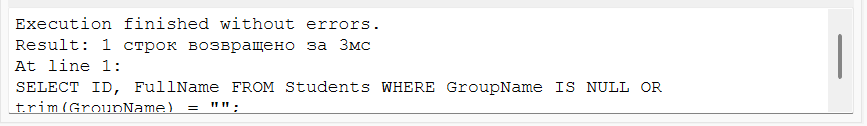


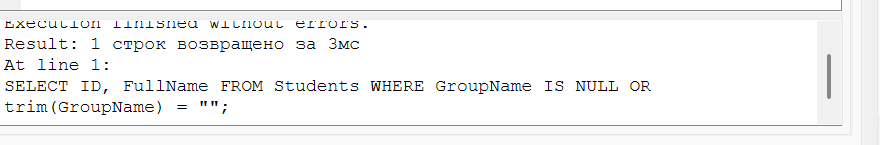
**Тест 4**: Пустые группы (Students)

SELECT ID, FullName FROM Students WHERE GroupName IS NULL OR trim(GroupName) = "";





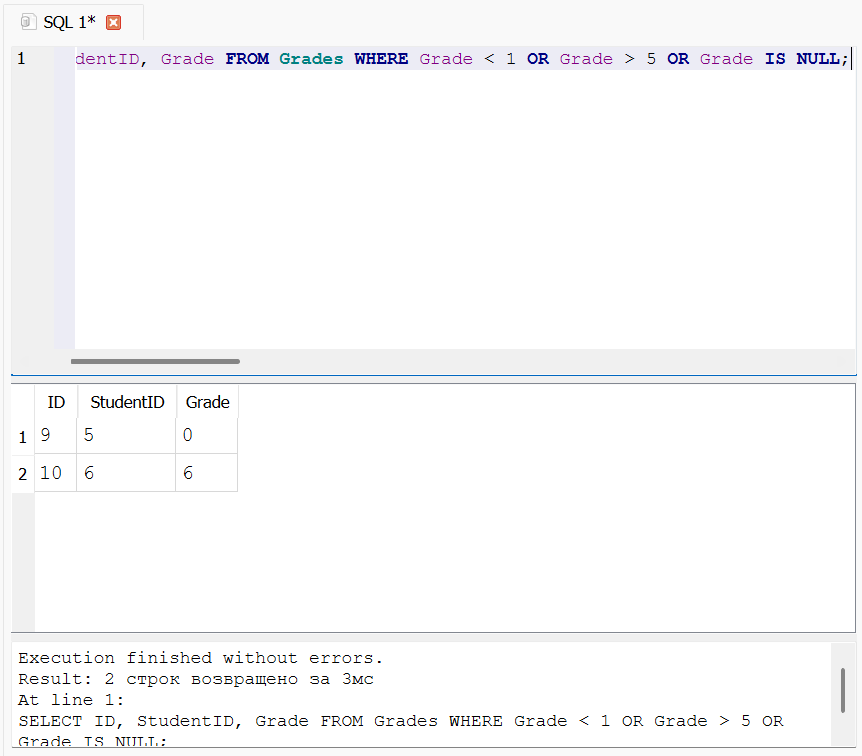


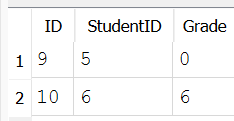


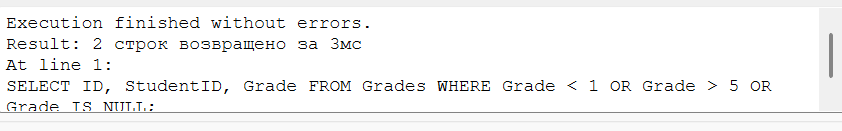
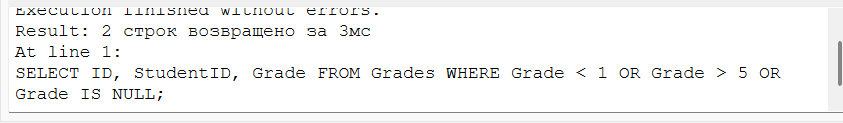
**ТЕСТИРОВАНИЕ STUDENTS-GRADES**

**Тест 3**: Неверные значения Grade

SELECT ID, StudentID, Grade FROM Grades WHERE Grade < 1 OR Grade > 5 OR Grade IS NULL;





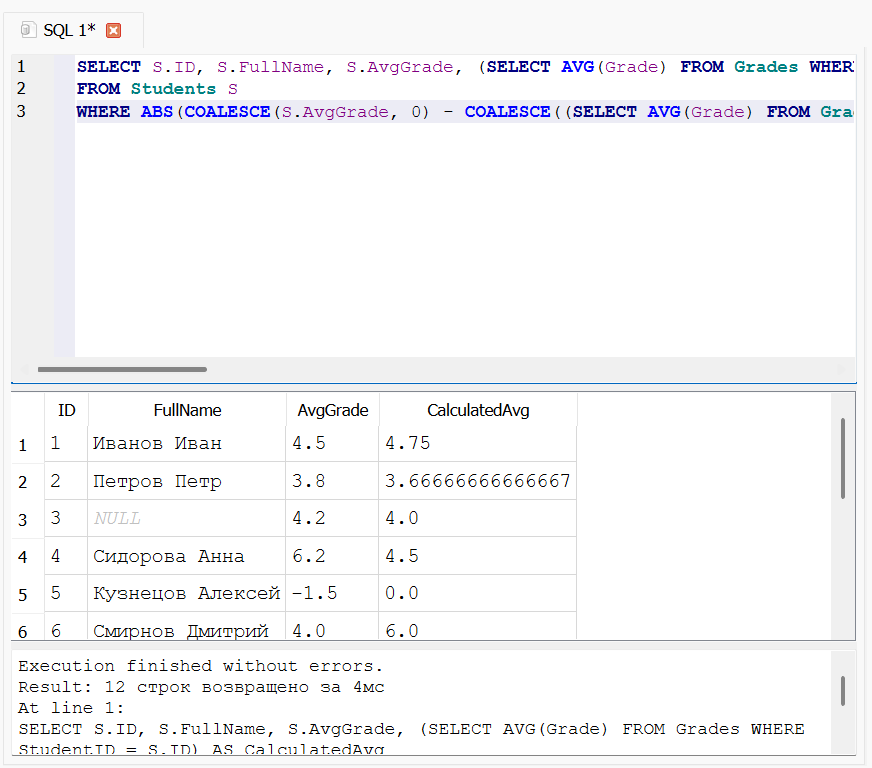
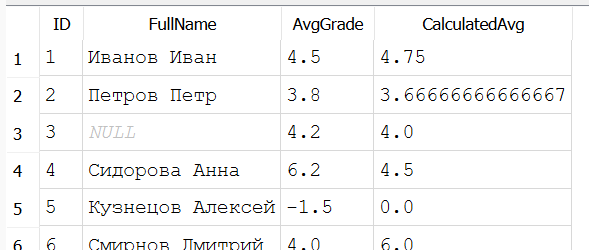
 

**Тест 4**: Проверка среднего балла

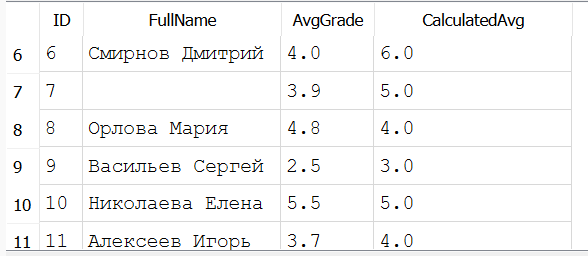
SELECT S.ID, S.FullName, S.AvgGrade, (SELECT AVG(Grade) FROM Grades WHERE StudentID = S.ID) AS CalculatedAvg

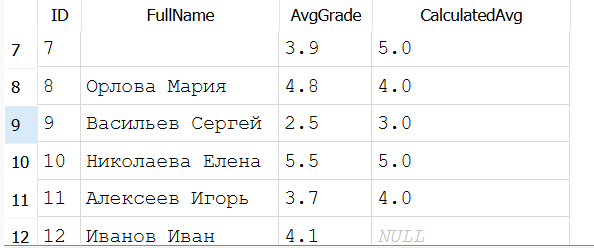
FROM Students S

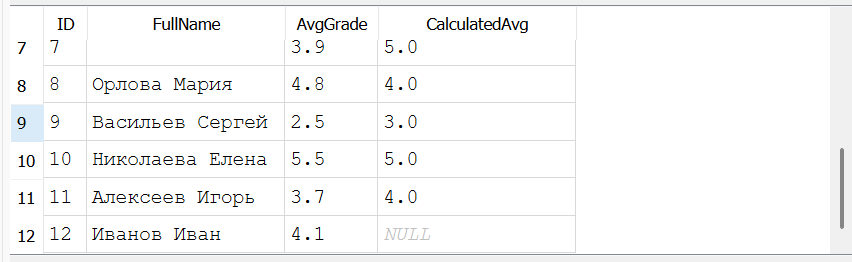
WHERE ABS(COALESCE(S.AvgGrade, 0) - COALESCE((SELECT AVG(Grade) FROM Grades WHERE StudentID = S.ID), 0)) > 0.1;

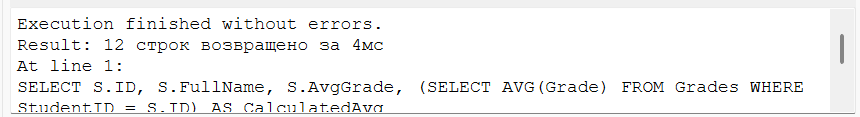
 

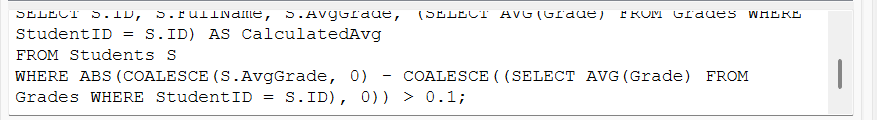








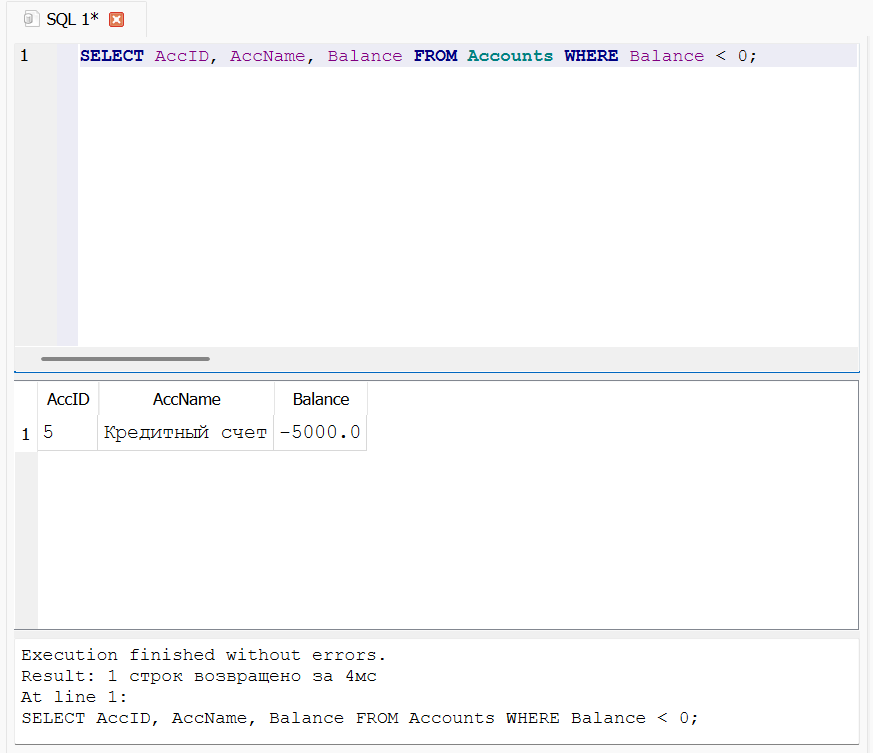
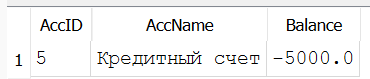
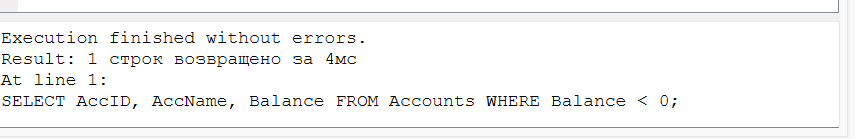




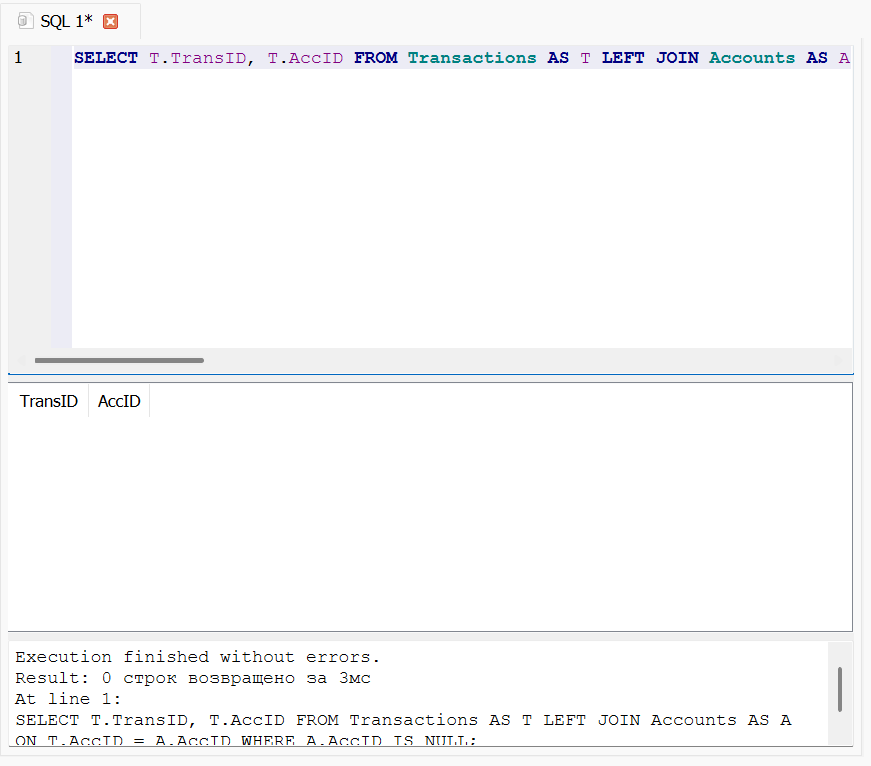
**ТЕСТИРОВАНИЕ TRANSACTIONS**

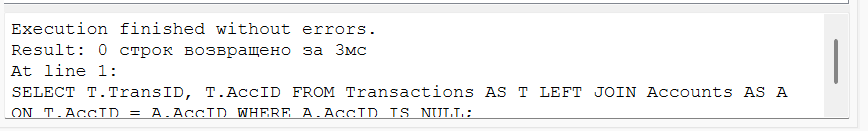
**Тест 1**: Отрицательные балансы

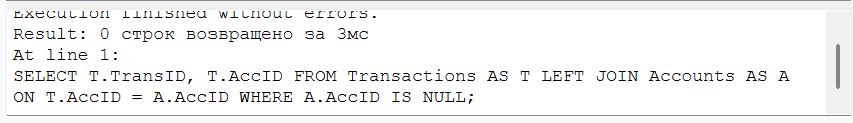
SELECT AccID, AccName, Balance FROM Accounts WHERE Balance < 0;

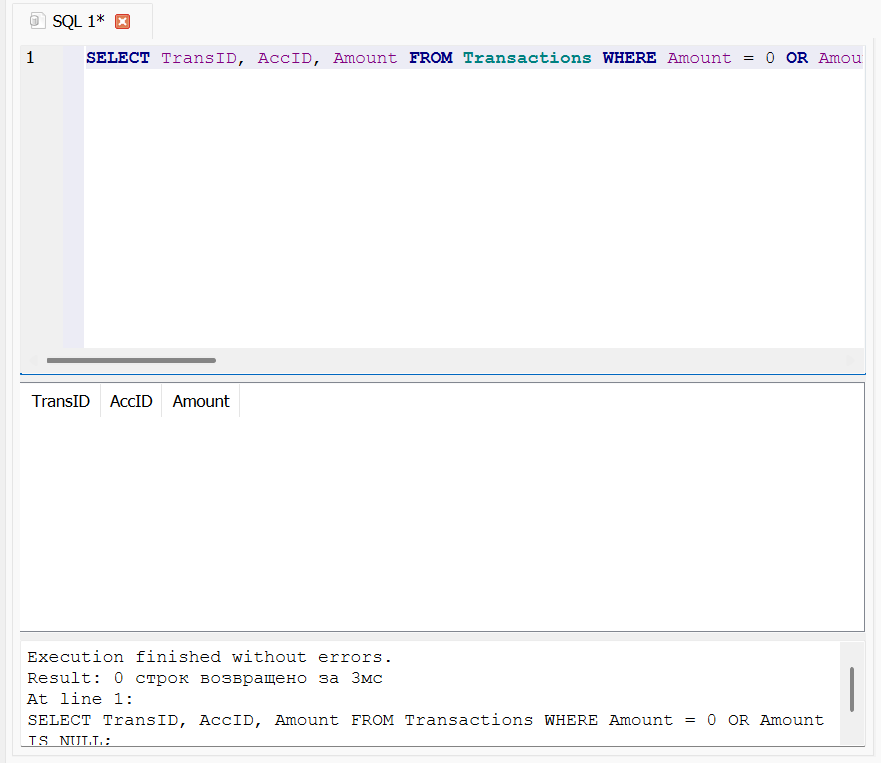
**Тест 2**: Транзакции без счета

SELECT T.TransID, T.AccID FROM Transactions AS T LEFT JOIN Accounts AS A ON T.AccID = A.AccID WHERE A.AccID IS NULL; 

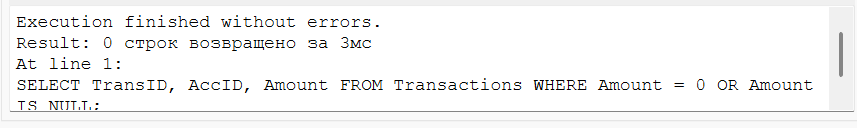


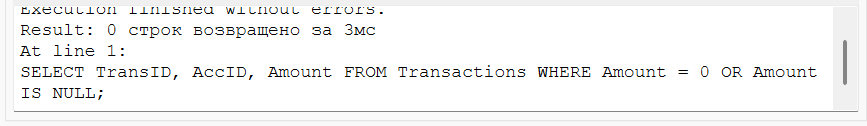


**Тест 3**: Некорректные суммы транзакций

SELECT TransID, AccID, Amount FROM Transactions WHERE Amount = 0 OR Amount IS NULL; 







ДОБАВИМ ТЕСТОВЫЕ ТРАНЗАКЦИИ ДЛЯ ПРОВЕРКИ

INSERT INTO Transactions (AccID, Amount, Description) VALUES

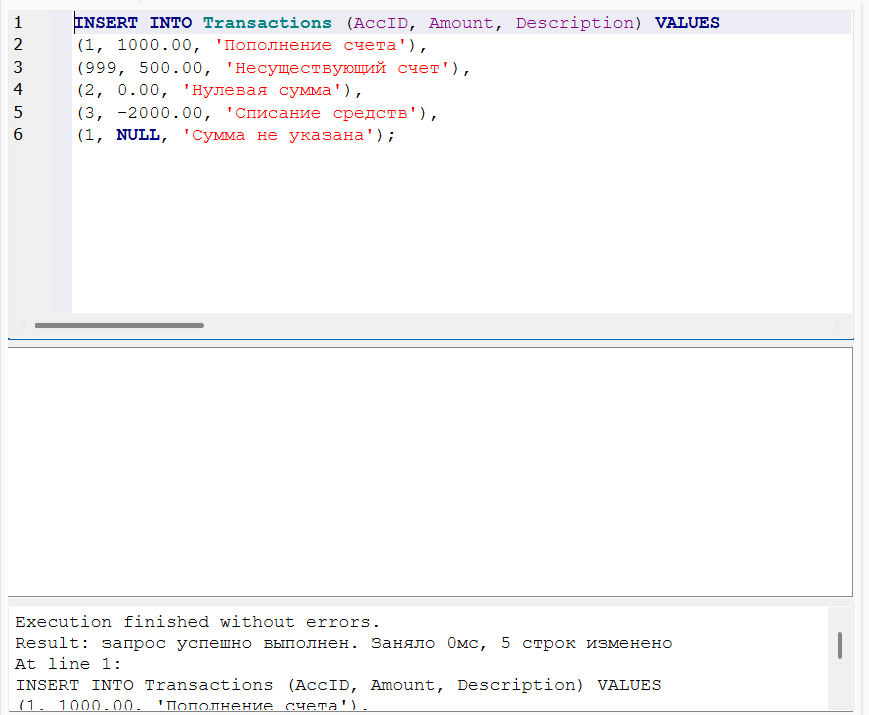
(1, 1000.00, 'Пополнение счета'),

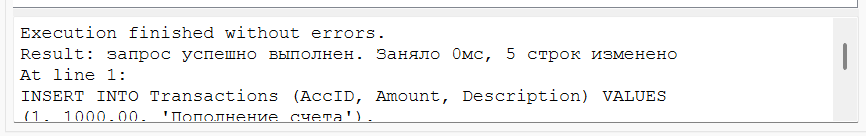
(999, 500.00, 'Несуществующий счет'),

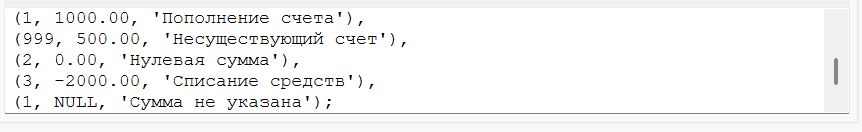
(2, 0.00, 'Нулевая сумма'),

(3, -2000.00, 'Списание средств'),

(1, NULL, 'Сумма не указана');

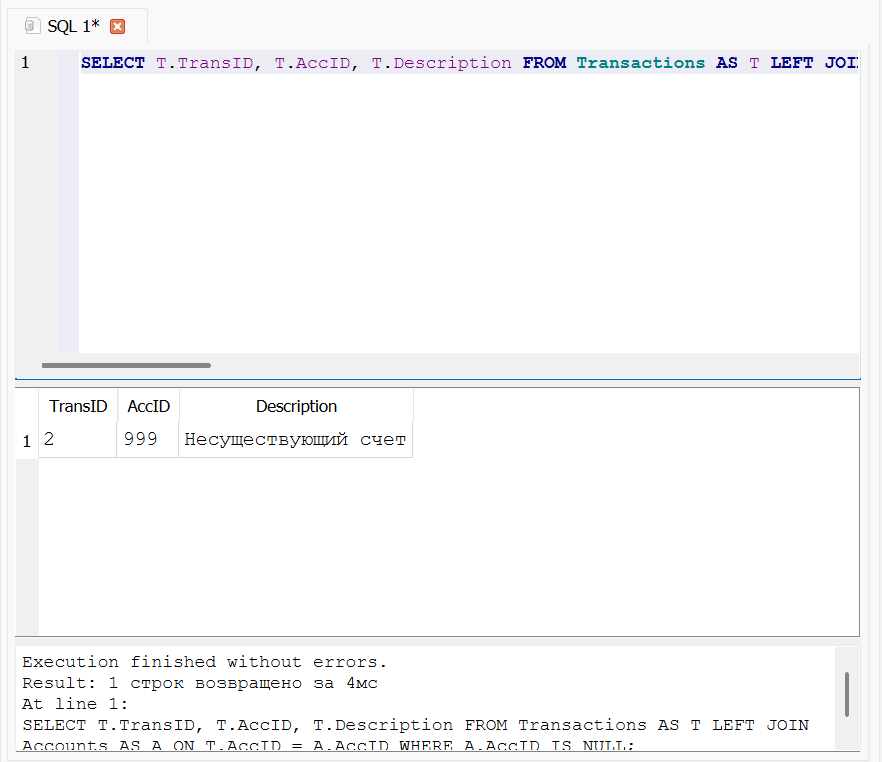


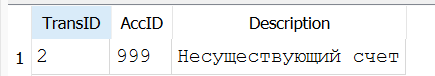


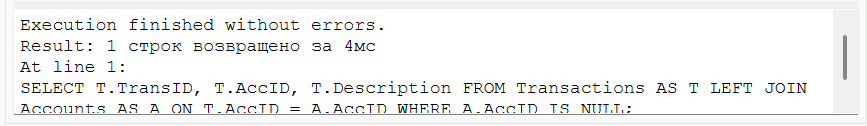


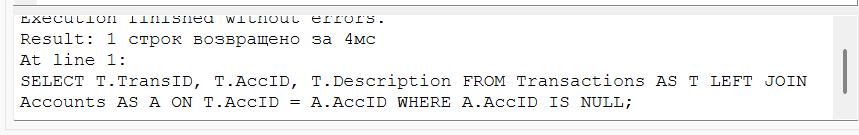
**Тест 2**: Транзакции без счета (повторно)

SELECT T.TransID, T.AccID, T.Description FROM Transactions AS T LEFT JOIN Accounts AS A ON T.AccID = A.AccID WHERE A.AccID IS NULL;

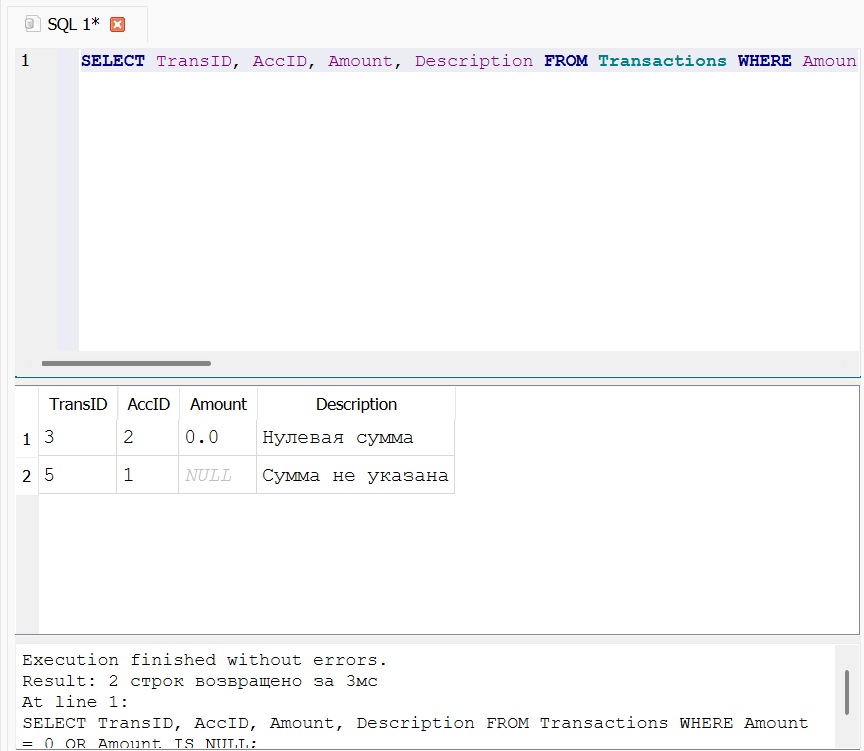


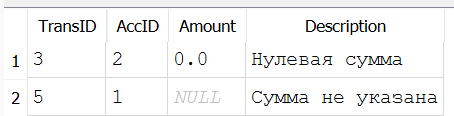


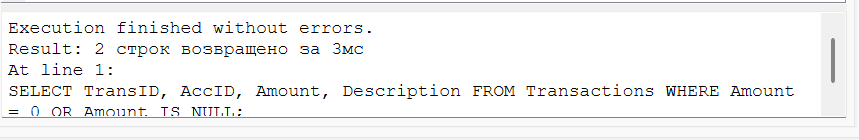


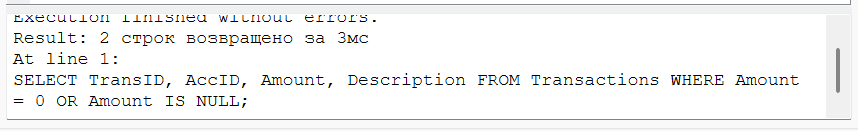


**Тест 3**: Некорректные суммы транзакций (повторно)

SELECT TransID, AccID, Amount, Description FROM Transactions WHERE Amount = 0 OR Amount IS NULL; 

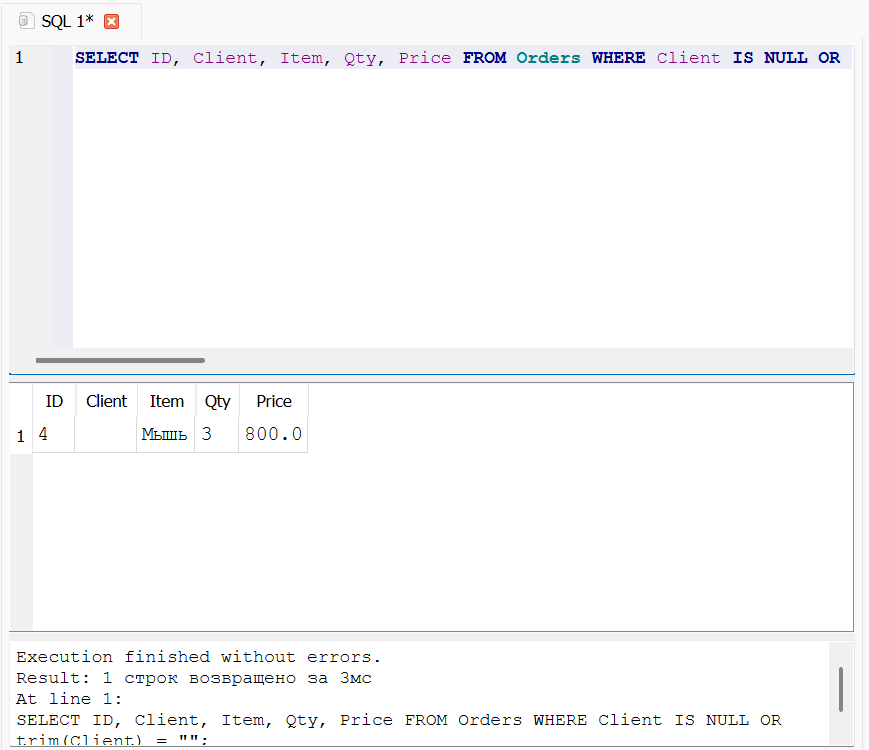


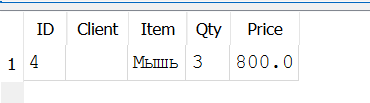


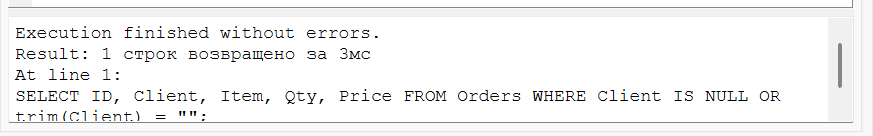


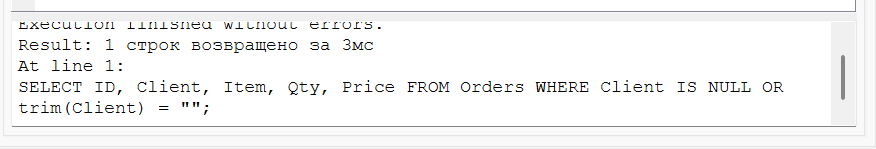
**ТЕСТИРОВАНИЕ ORDERS**

**Тест 1**: Пустые клиенты



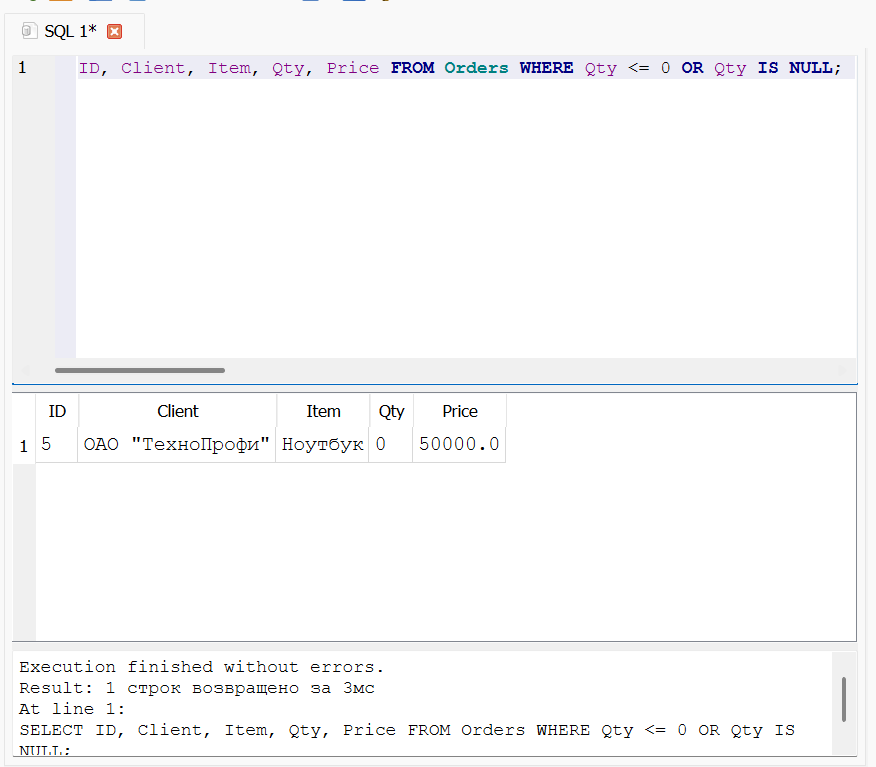


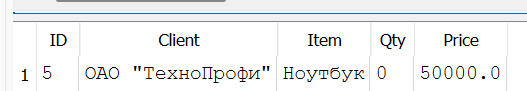


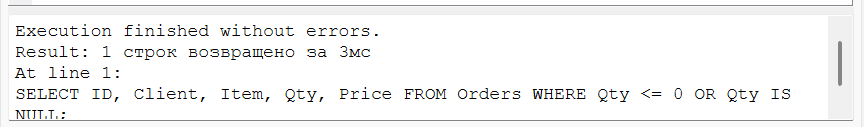


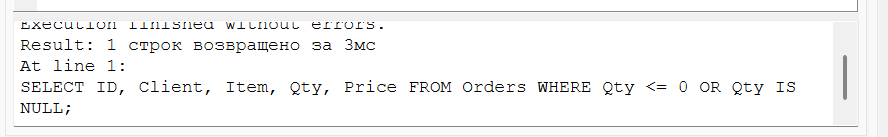
**Тест 2**: Некорректные количества

SELECT ID, Client, Item, Qty, Price FROM Orders WHERE Qty <= 0 OR Qty IS NULL;

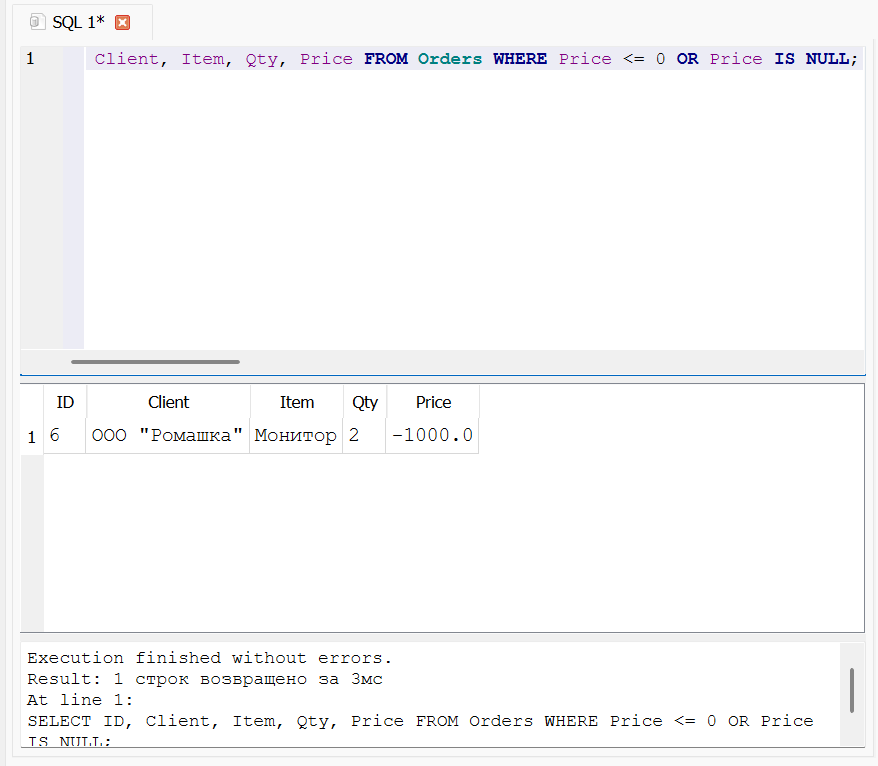
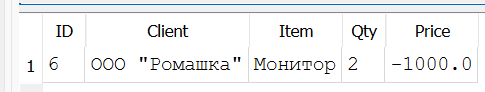


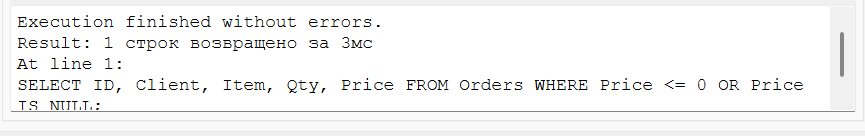


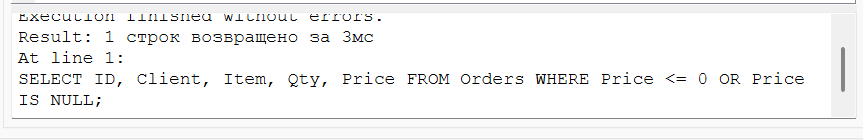




**Тест 3**: Некорректные цены

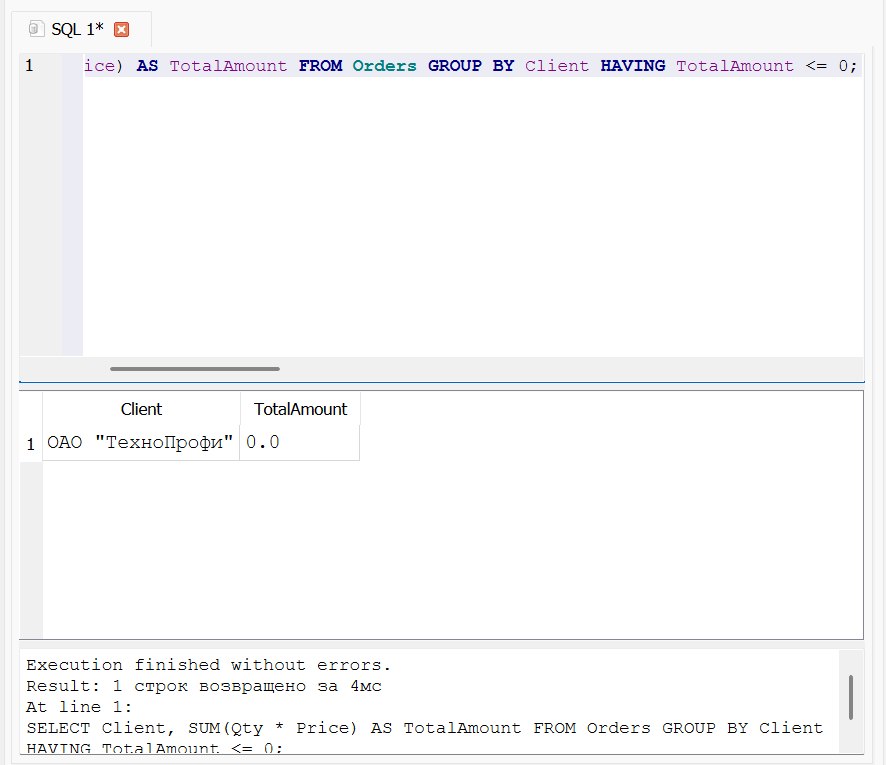
SELECT ID, Client, Item, Qty, Price FROM Orders WHERE Price <= 0 OR Price IS NULL;  

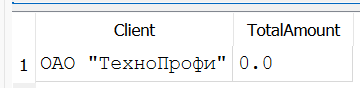


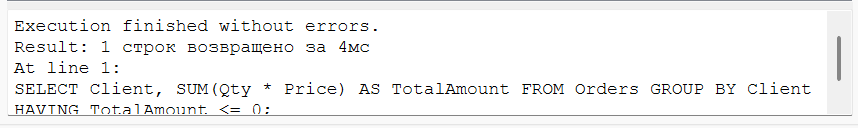


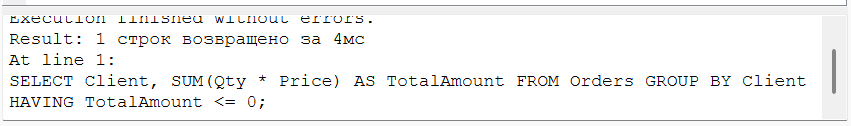
**Тест 4**: Проверка агрегатов

SELECT Client, SUM(Qty \* Price) AS TotalAmount FROM Orders GROUP BY Client HAVING TotalAmount <= 0;









**СВОДНЫЙ ОТЧЕТ ПО ВСЕМ ОШИБКАМ**

-- Сводный отчет по всем найденным ошибкам

SELECT 'STUDENTS - Пустые имена' as КатегорияОшибки, COUNT(\*) as КоличествоОшибок FROM Students WHERE FullName IS NULL OR trim(FullName) = ""

UNION ALL

SELECT 'STUDENTS - Некорректные оценки', COUNT(\*) FROM Students WHERE AvgGrade < 0 OR AvgGrade > 5 OR AvgGrade IS NULL

UNION ALL

SELECT 'STUDENTS - Дубликаты имен', (SELECT COUNT(\*) FROM (SELECT FullName FROM Students GROUP BY FullName HAVING COUNT(\*) > 1))

UNION ALL

SELECT 'STUDENTS - Пустые группы', COUNT(\*) FROM Students WHERE GroupName IS NULL OR trim(GroupName) = ""

UNION ALL

SELECT 'GRADES - Orphan записи', COUNT(\*) FROM Grades AS G LEFT JOIN Students AS S ON G.StudentID = S.ID WHERE S.ID IS NULL

UNION ALL

SELECT 'GRADES - Некорректные оценки', COUNT(\*) FROM Grades WHERE Grade < 1 OR Grade > 5 OR Grade IS NULL

UNION ALL

SELECT 'GRADES - Студенты без оценок', COUNT(\*) FROM Students AS S LEFT JOIN Grades AS G ON S.ID = G.StudentID WHERE G.ID IS NULL

UNION ALL

SELECT 'ACCOUNTS - Отрицательные балансы', COUNT(\*) FROM Accounts WHERE Balance < 0

UNION ALL

SELECT 'TRANSACTIONS - Несуществующие счета', COUNT(\*) FROM Transactions AS T LEFT JOIN Accounts AS A ON T.AccID = A.AccID WHERE A.AccID IS NULL

UNION ALL

SELECT 'TRANSACTIONS - Некорректные суммы', COUNT(\*) FROM Transactions WHERE Amount = 0 OR Amount IS NULL

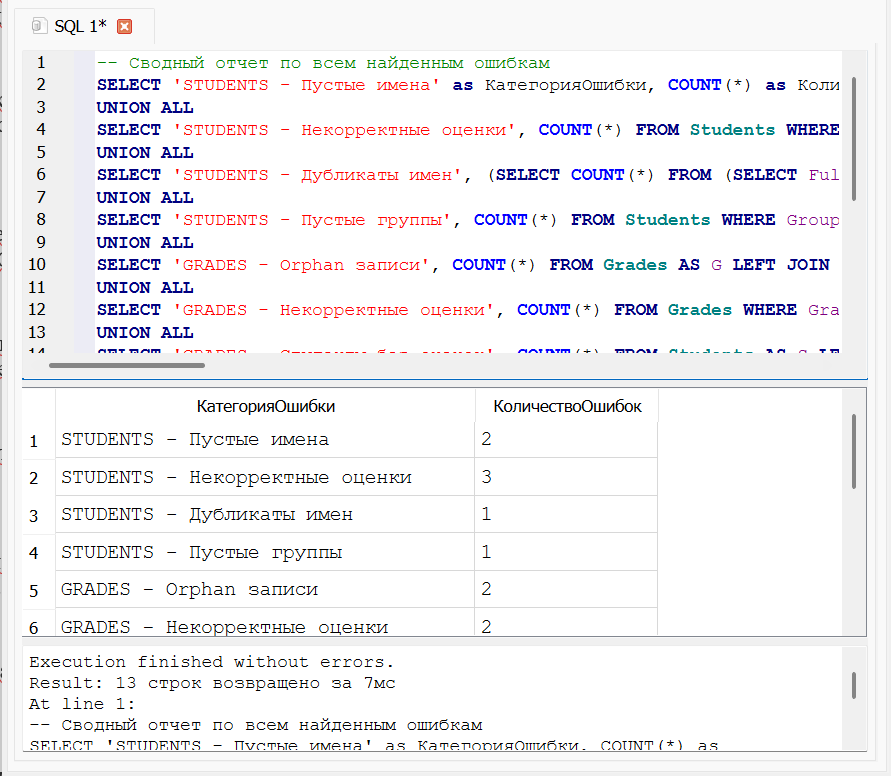
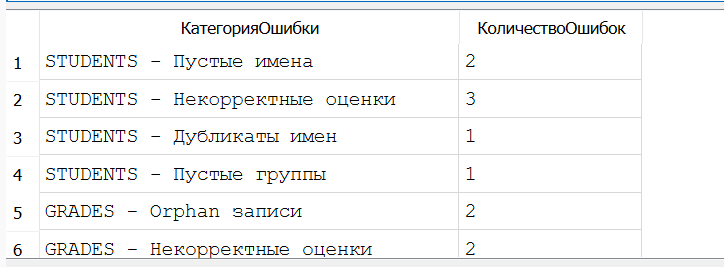
UNION ALL

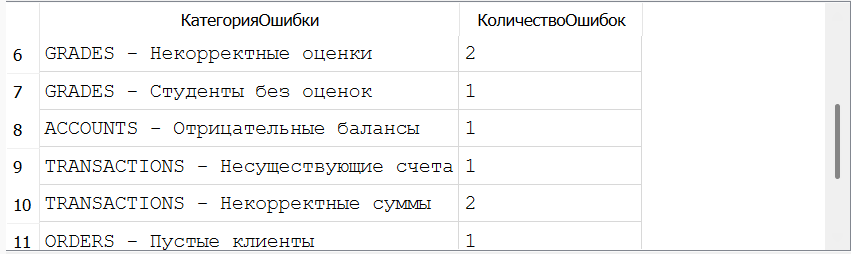
SELECT 'ORDERS - Пустые клиенты', COUNT(\*) FROM Orders WHERE Client IS NULL OR trim(Client) = ""

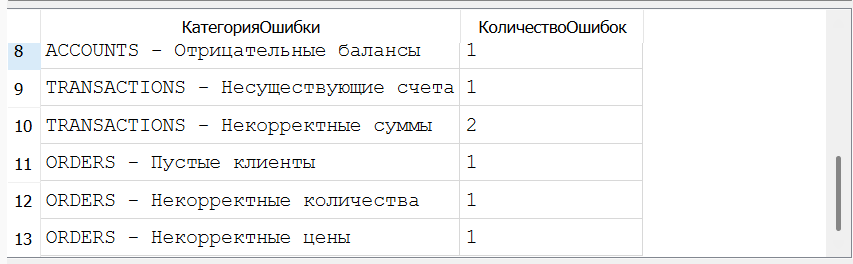
UNION ALL

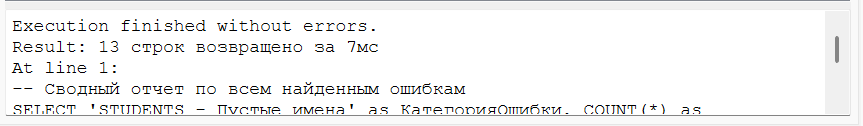
SELECT 'ORDERS - Некорректные количества', COUNT(\*) FROM Orders WHERE Qty <= 0 OR Qty IS NULL

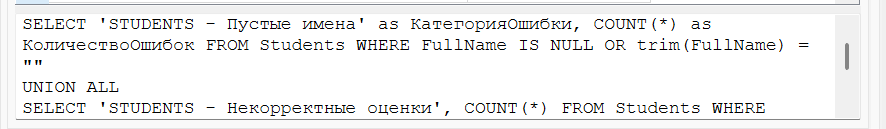
UNION ALL

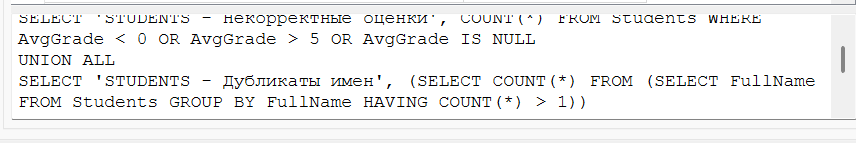
SELECT 'ORDERS - Некорректные цены', COUNT(\*) FROM Orders WHERE Price <= 0 OR Price IS NULL;  

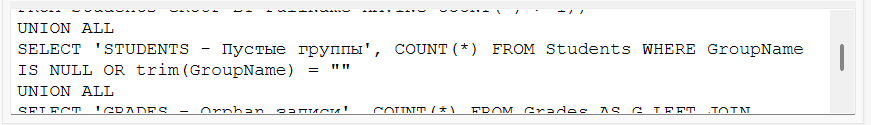


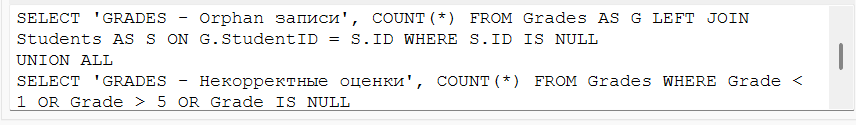


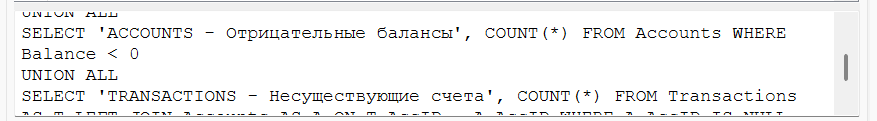












**ДЕТАЛИЗИРОВАННЫЙ ОТЧЕТ**

-- Детальный отчет по ошибкам

SELECT

'Students' as Module,

'Пустое имя: ' || COALESCE(FullName, 'NULL') as ErrorDescription,

'ID: ' || ID as Details

FROM Students

WHERE FullName IS NULL OR trim(FullName) = ""

UNION ALL

SELECT

'Students',

'Некорректная оценка: ' || COALESCE(AvgGrade, 'NULL'),

'ID: ' || ID || ', Имя: ' || COALESCE(FullName, 'NULL')

FROM Students

WHERE AvgGrade < 0 OR AvgGrade > 5 OR AvgGrade IS NULL

UNION ALL

SELECT

'Grades',

'Orphan запись - несуществующий StudentID: ' || G.StudentID,

'ID оценки: ' || G.ID || ', Предмет: ' || G.Subject

FROM Grades AS G

LEFT JOIN Students AS S ON G.StudentID = S.ID

WHERE S.ID IS NULL

UNION ALL

SELECT

'Grades',

'Некорректная оценка: ' || Grade,

'ID оценки: ' || ID || ', StudentID: ' || StudentID

FROM Grades

WHERE Grade < 1 OR Grade > 5 OR Grade IS NULL

UNION ALL

SELECT

'Accounts',

'Отрицательный баланс: ' || Balance,

'ID счета: ' || AccID || ', Название: ' || AccName

FROM Accounts

WHERE Balance < 0

UNION ALL

SELECT

'Transactions',

'Транзакция несуществующего счета: ' || T.AccID,

'ID транзакции: ' || T.TransID || ', Описание: ' || T.Description

FROM Transactions AS T

LEFT JOIN Accounts AS A ON T.AccID = A.AccID

WHERE A.AccID IS NULL

UNION ALL

SELECT

'Orders',

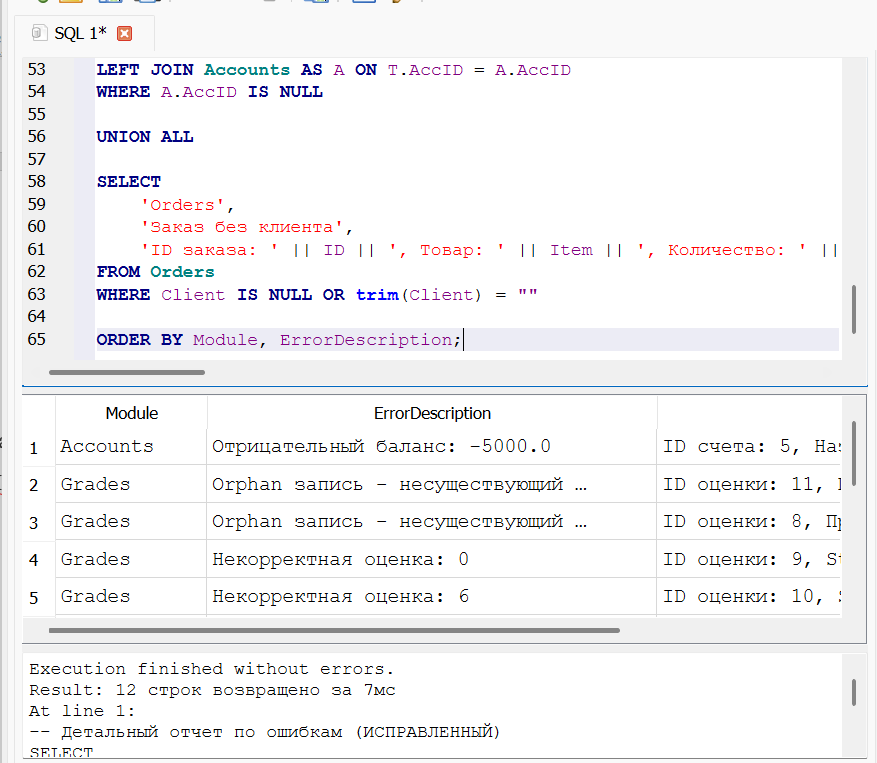
'Заказ без клиента',

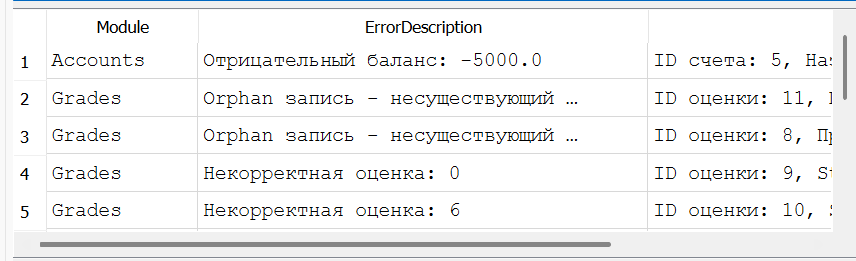
'ID заказа: ' || ID || ', Товар: ' || Item || ', Количество: ' || Qty

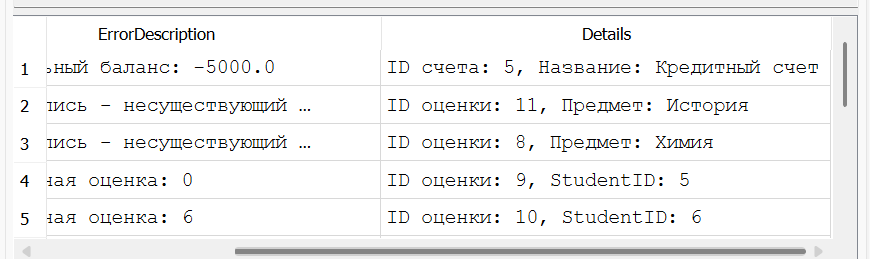
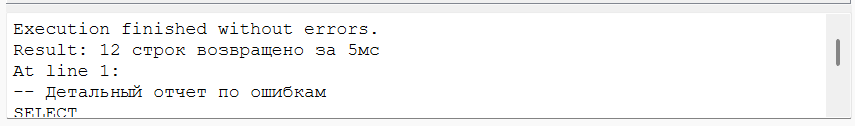
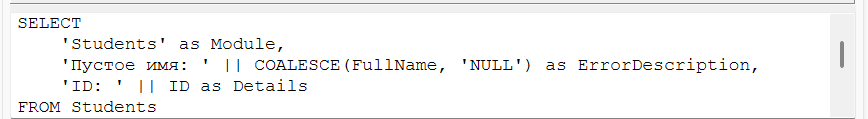
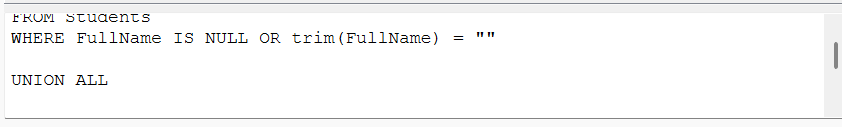
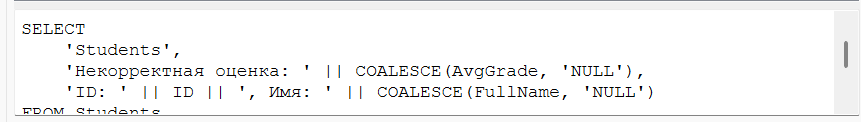
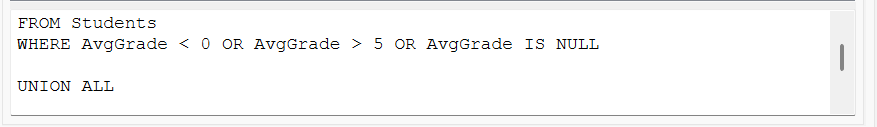
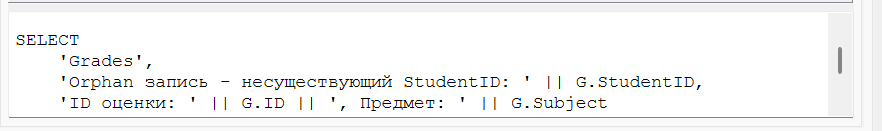
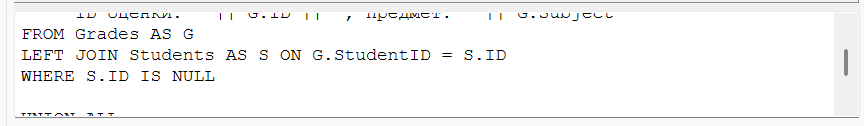
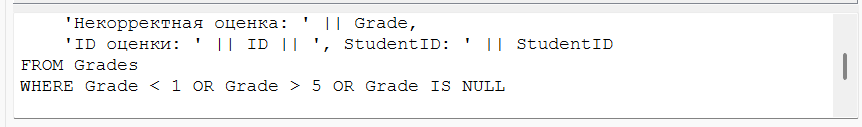
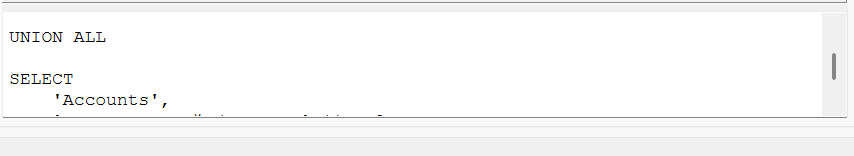
FROM Orders

WHERE Client IS NULL OR trim(Client) = ""

ORDER BY Module, ErrorDescription;





**СОХРАНЕНИЕ РЕЗУЛЬТАТОВ ТЕСТИРОВАНИЯ**

результаты тестирования в TestRuns (вручную)

INSERT INTO TestRuns (TestCaseID, RunTime, Status, Details) VALUES

(1, datetime('now'), 'Failed', 'Найдены студенты с пустыми именами: ID 3, 7'),

(2, datetime('now'), 'Failed', 'Найдены студенты с некорректными оценками: ID 4, 5, 10'),

(3, datetime('now'), 'Failed', 'Найден дубликат имени: Иванов Иван'),

(4, datetime('now'), 'Failed', 'Найден студент без группы: ID 11'),

(5, datetime('now'), 'Failed', 'Найдены orphan оценки: ID 8, 11'),

(6, datetime('now'), 'Passed', 'Студенты без оценок не найдены'),

(7, datetime('now'), 'Failed', 'Найдены некорректные оценки: ID 9, 10'),

(8, datetime('now'), 'Failed', 'Обнаружены расхождения в расчете среднего балла'),

(9, datetime('now'), 'Failed', 'Найден счет с отрицательным балансом: ID 5'),

(10, datetime('now'), 'Failed', 'Найдена транзакция несуществующего счета: ID 2'),

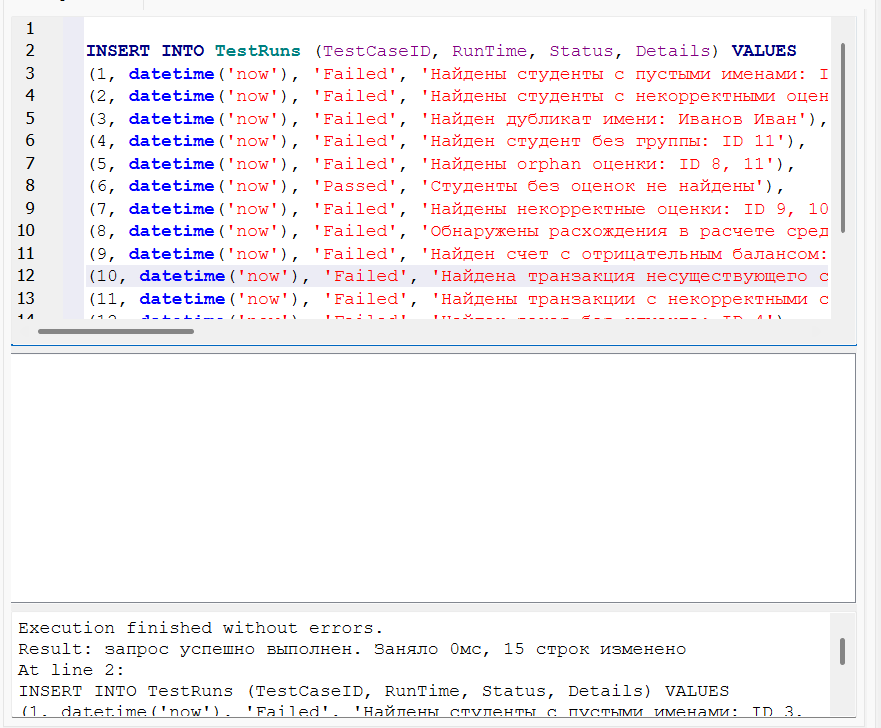
(11, datetime('now'), 'Failed', 'Найдены транзакции с некорректными суммами: ID 3, 5'),

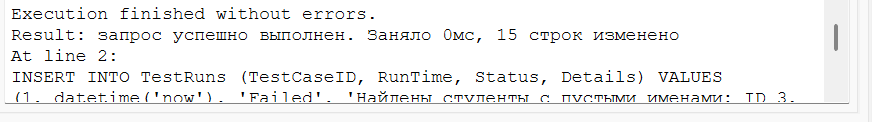
(12, datetime('now'), 'Failed', 'Найден заказ без клиента: ID 4'),

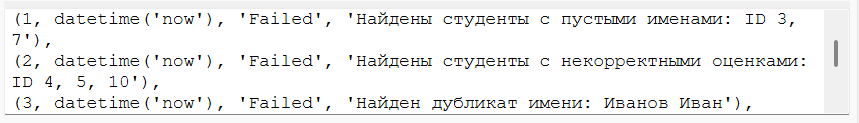
(13, datetime('now'), 'Failed', 'Найден заказ с некорректным количеством: ID 5'),

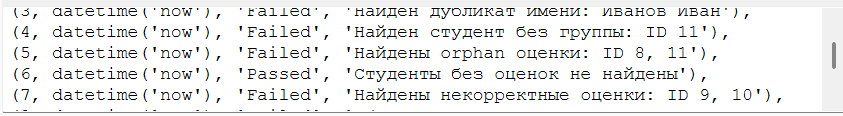
(14, datetime('now'), 'Failed', 'Найден заказ с некорректной ценой: ID 6'),

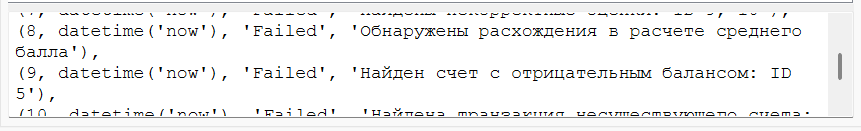
(15, datetime('now'), 'Passed', 'Агрегаты заказов корректны');

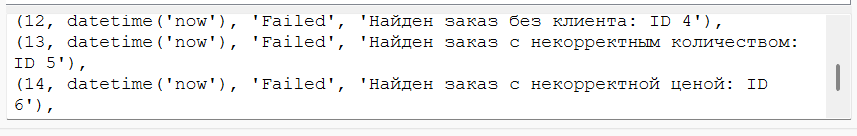












**Проверяем записанные результаты**

Полная истрия

SELECT

TR.RunID,

TC.ModuleName,

TC.TestName,

TR.Status,

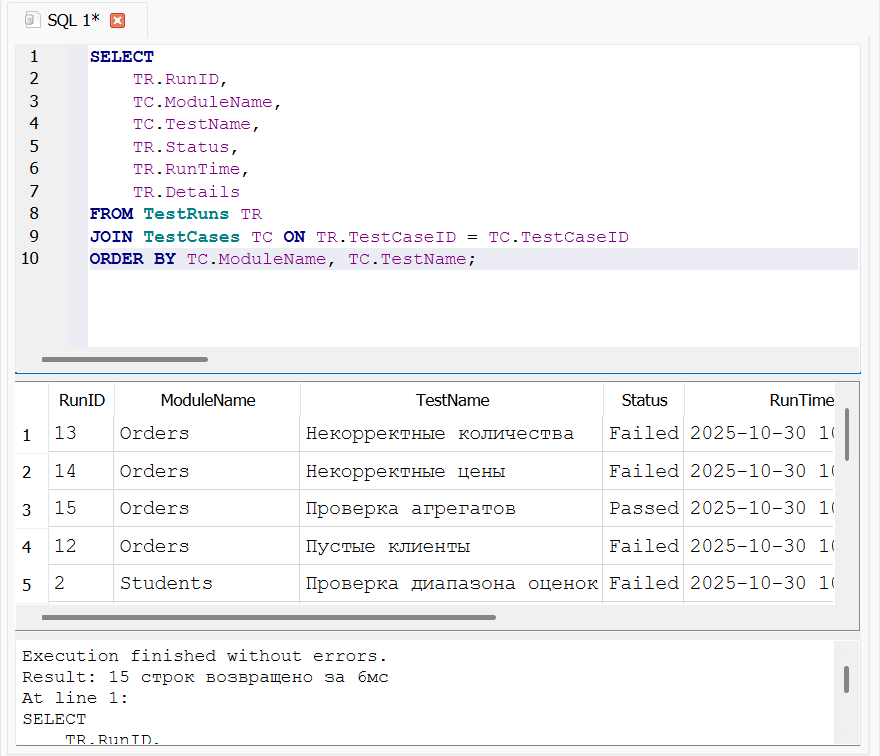
TR.RunTime,

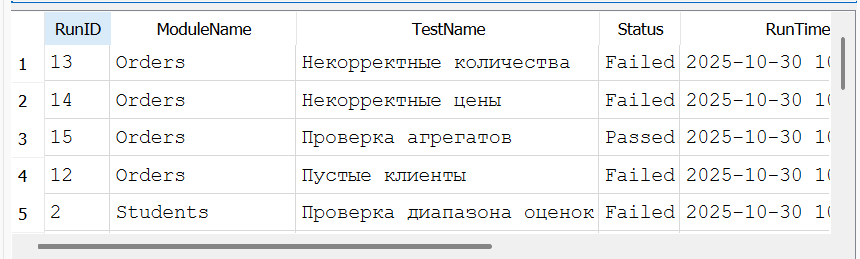
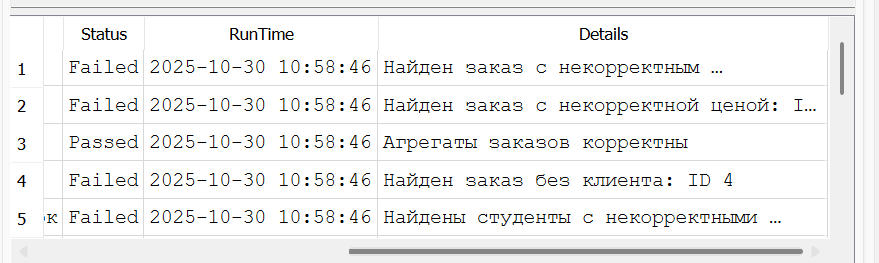
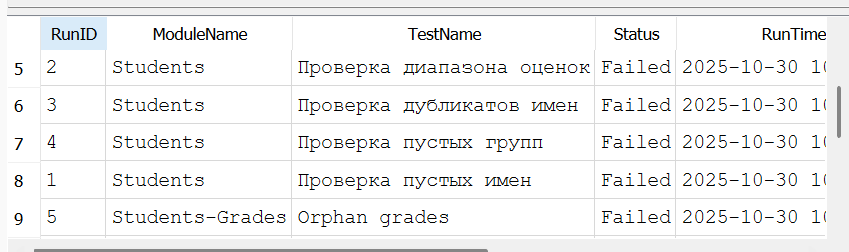
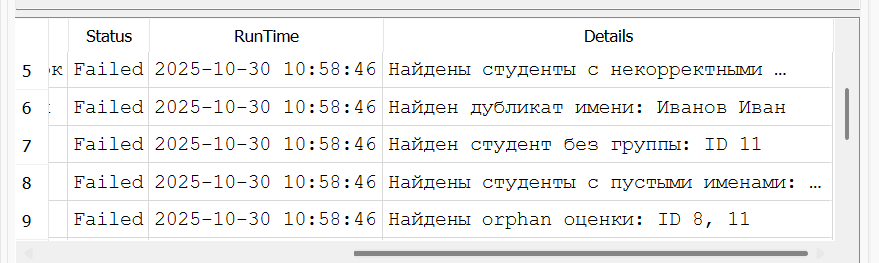
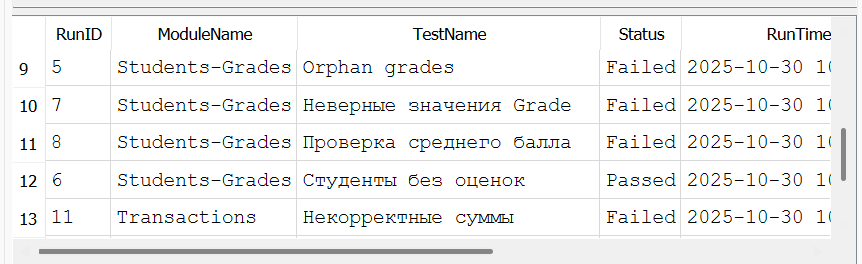
TR.Details

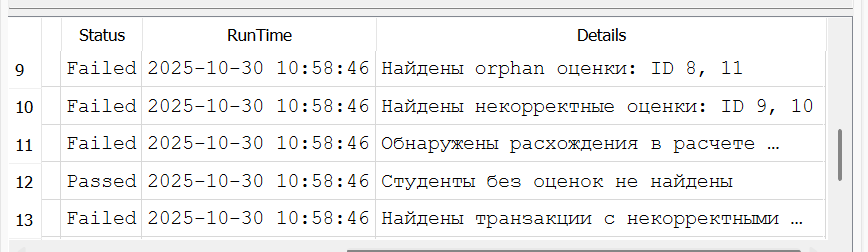
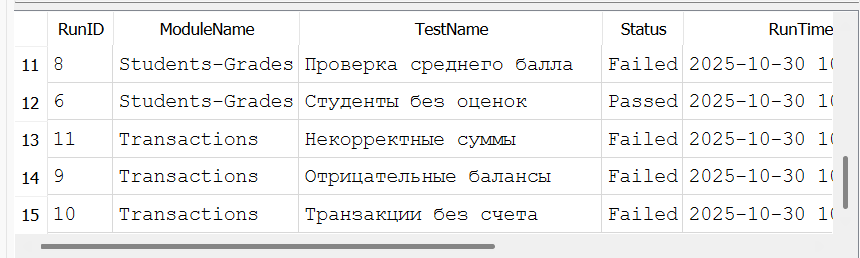
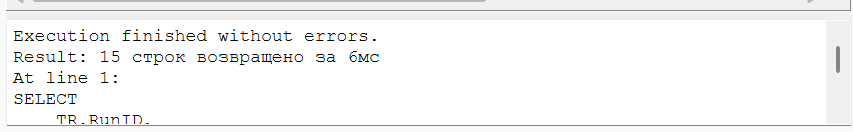
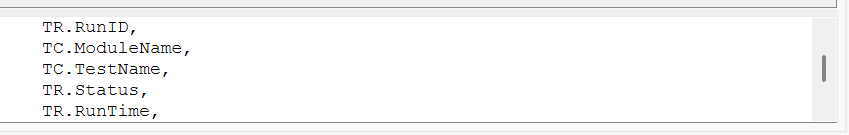
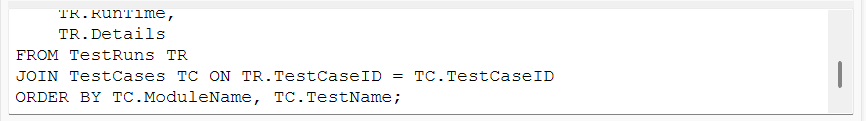
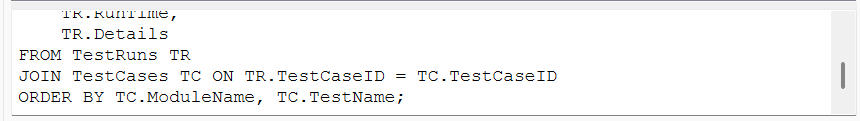
FROM TestRuns TR

JOIN TestCases TC ON TR.TestCaseID = TC.TestCaseID

ORDER BY TC.ModuleName, TC.TestName;



**ШАГ 20**: СОЗДАНИЕ ПРОЦЕДУР ДЛЯ АВТОМАТИЧЕСКОГО ТЕСТИРОВАНИЯ

CREATE VIEW vw\_RunAllTests AS

SELECT

tc.ModuleName,

tc.TestName,

tc.Description,

tc.SQLCheck,

tc.ExpectedResult,

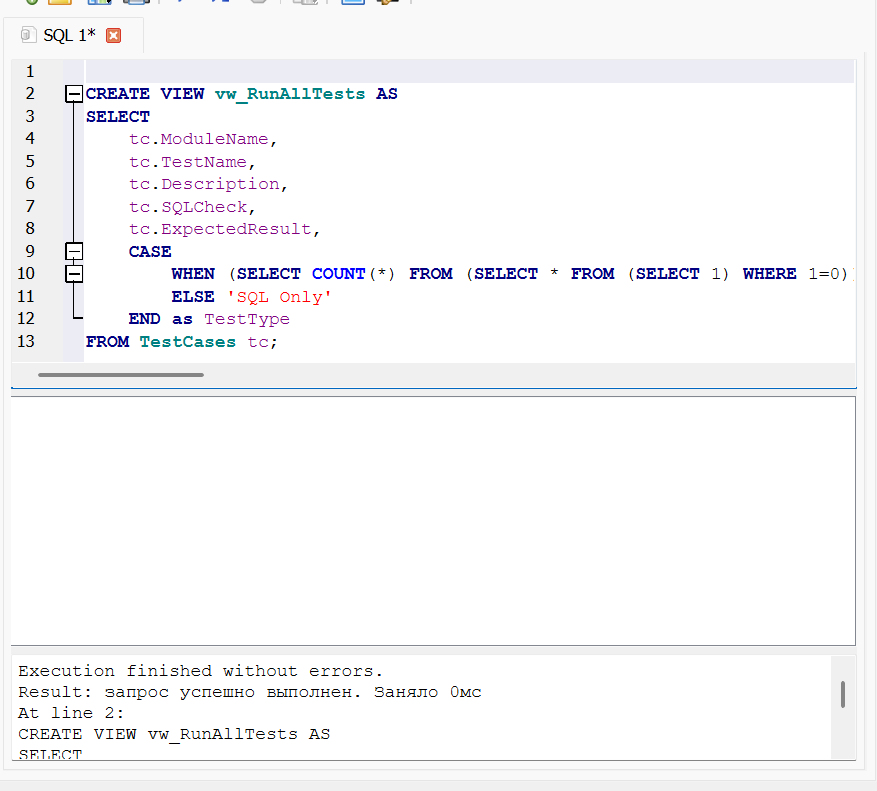
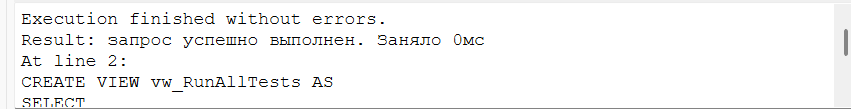
CASE

WHEN (SELECT COUNT(\*) FROM (SELECT \* FROM (SELECT 1) WHERE 1=0)) = 0 THEN 'Manual Check Required'

ELSE 'SQL Only'

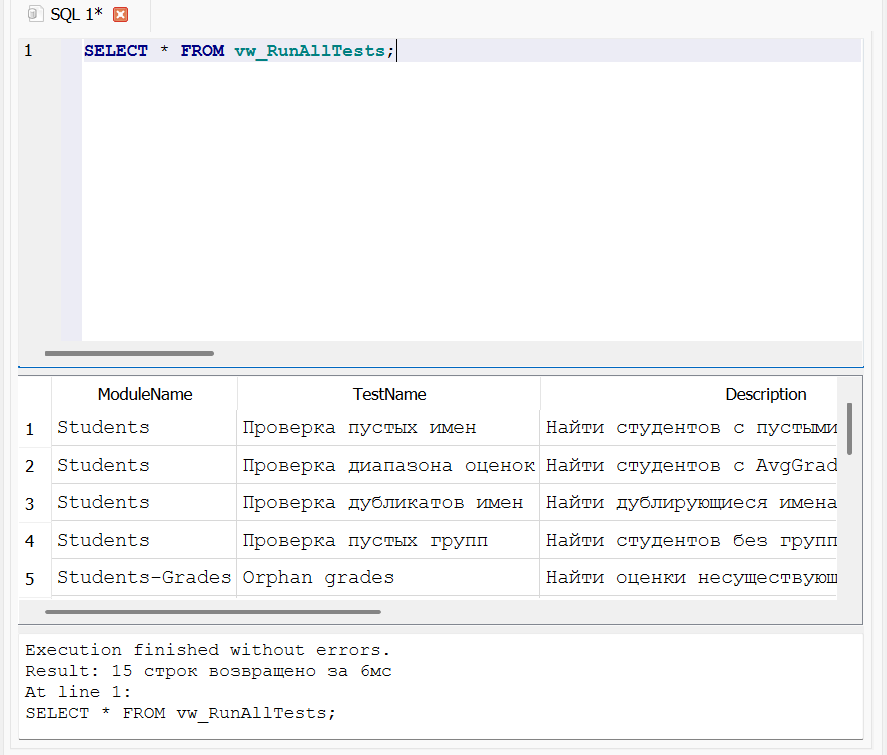
END as TestType

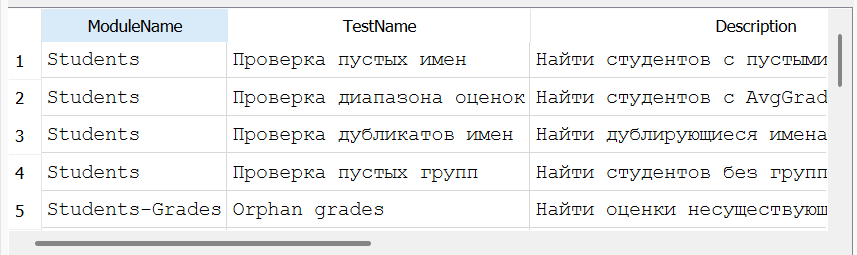
FROM TestCases tc;

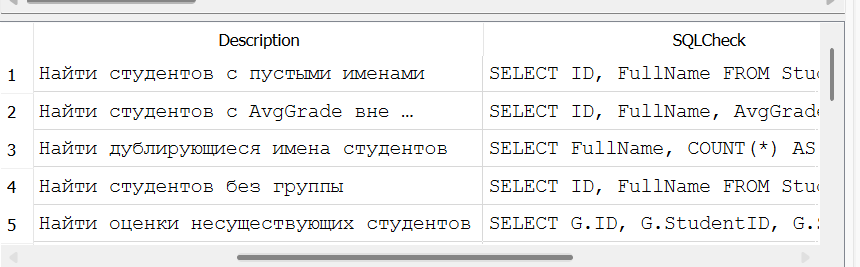
 

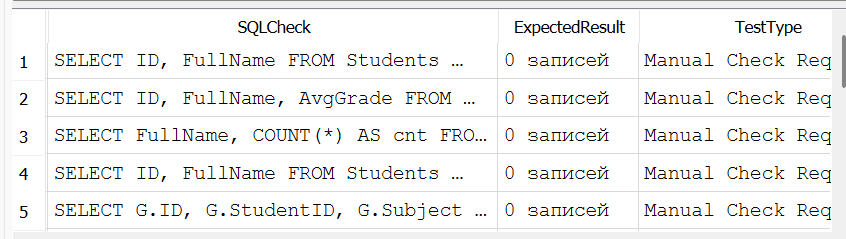
**Проверка представлений**

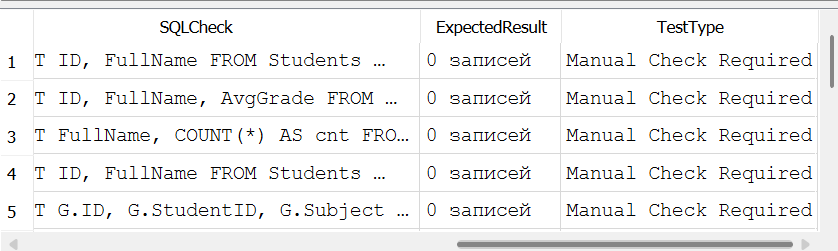
SELECT \* FROM vw\_RunAllTests;

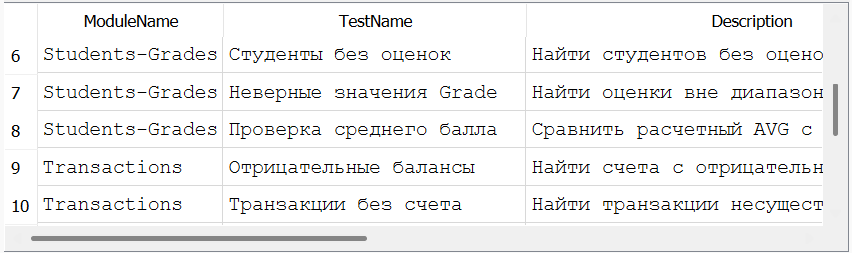
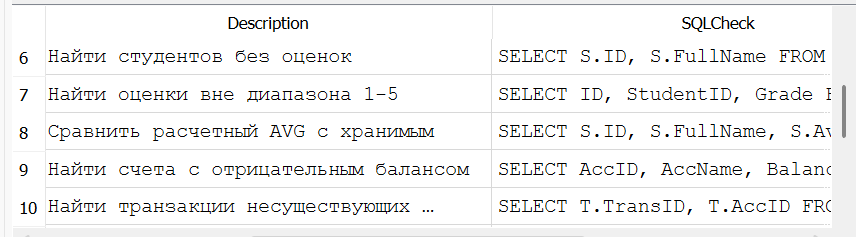


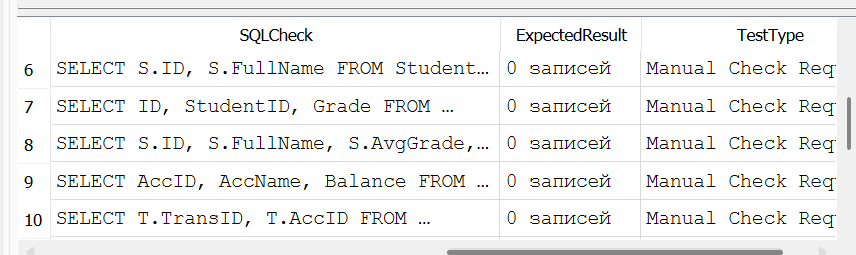


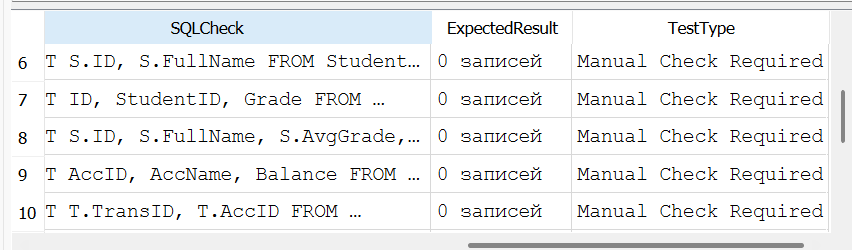


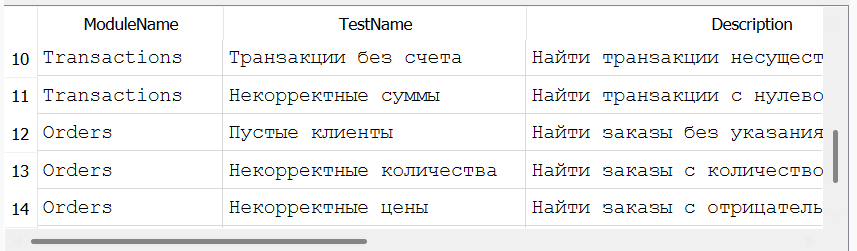


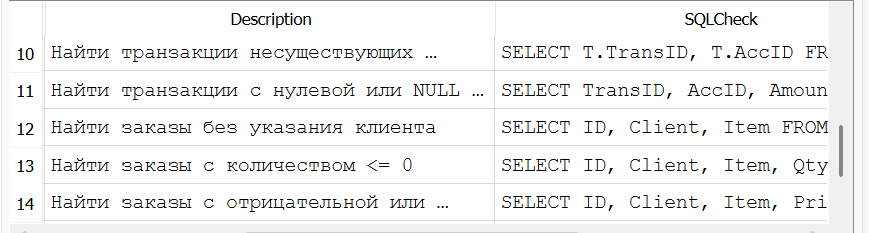


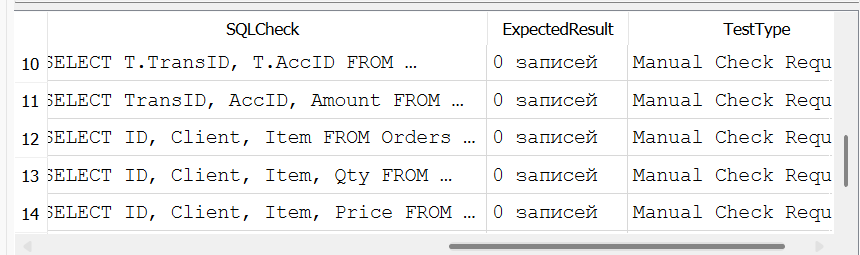
 

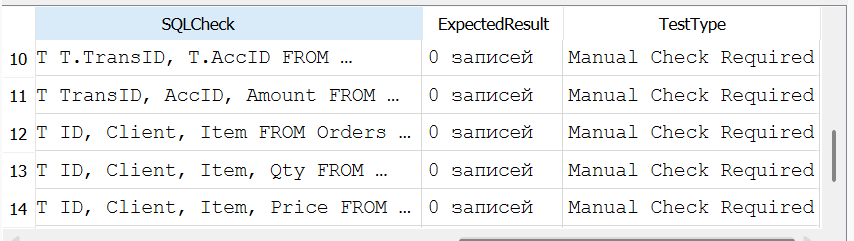


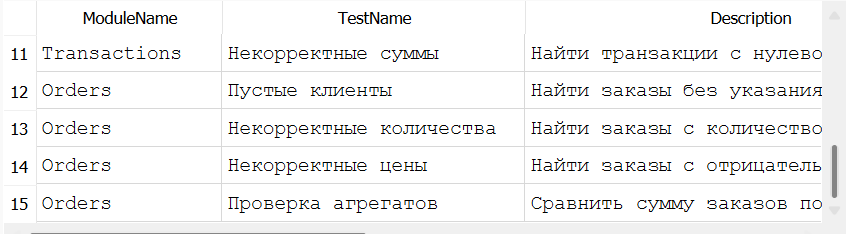


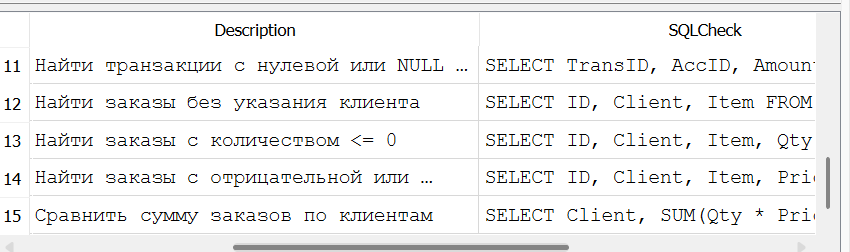


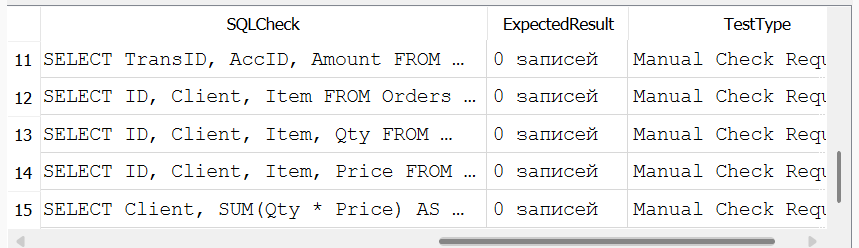
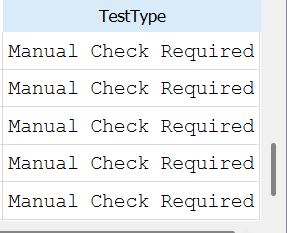










**ШАГ 21**: СОЗДАНИЕ ОТЧЕТОВ

SELECT

ModuleName,

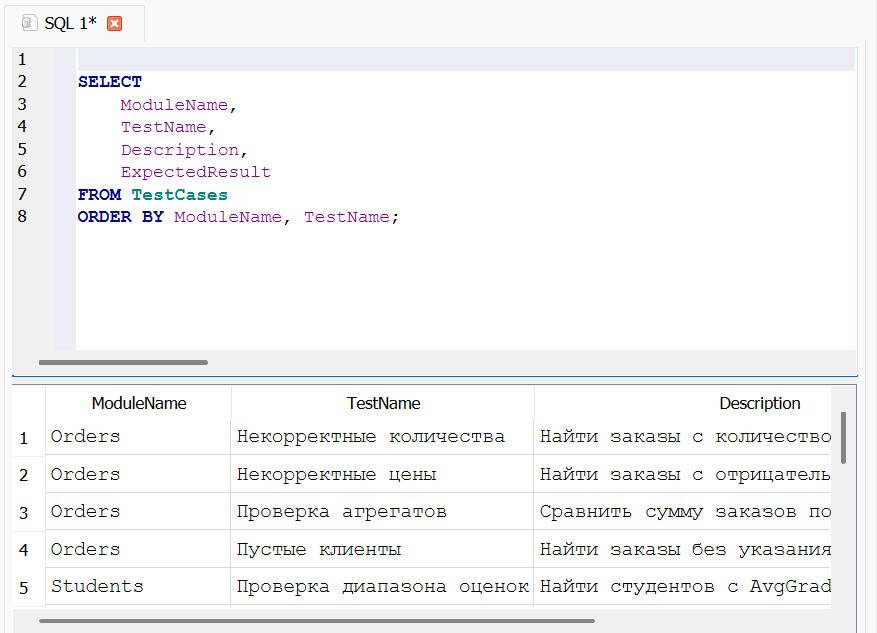
TestName,

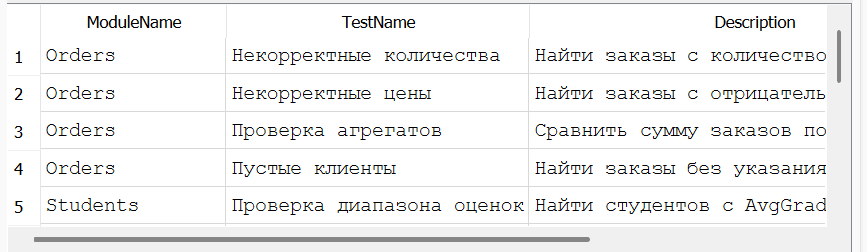
Description,

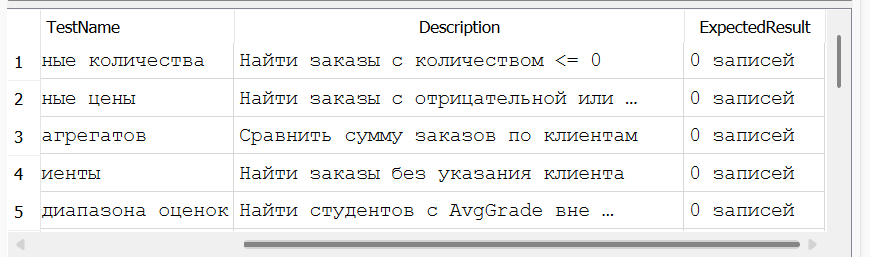
ExpectedResult

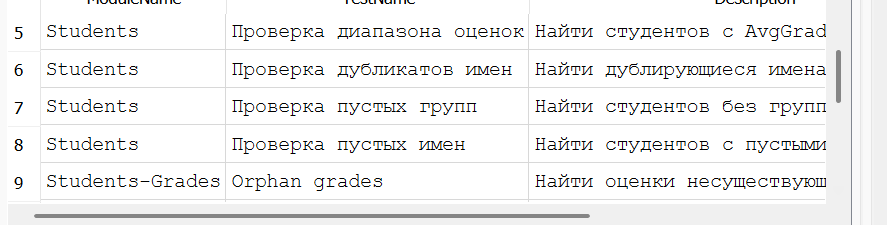
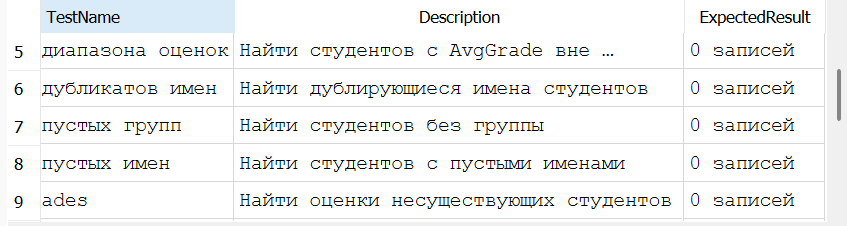
FROM TestCases

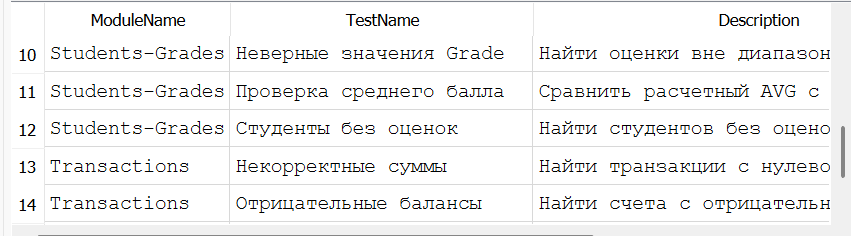
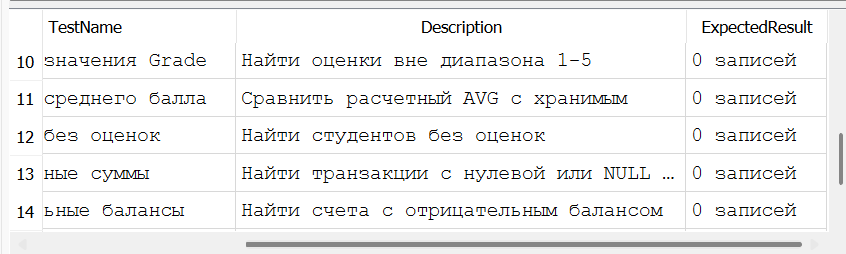
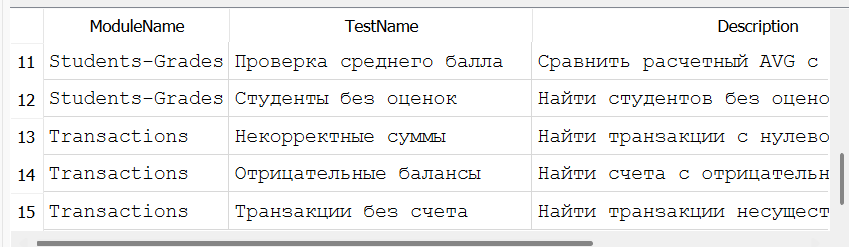
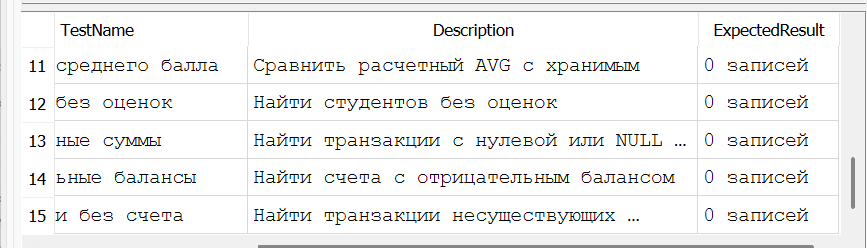
ORDER BY ModuleName, TestName;







**Отчет по найденным ошибкам:**

SELECT

'Students - пустые имена' as ErrorType,

COUNT(\*) as ErrorCount

FROM Students WHERE FullName IS NULL OR trim(FullName) = ""

UNION ALL

SELECT

'Students - некорректные оценки',

COUNT(\*)

FROM Students WHERE AvgGrade < 0 OR AvgGrade > 5 OR AvgGrade IS NULL

UNION ALL

SELECT

'Grades - orphan записи',

COUNT(\*)

FROM Grades AS G LEFT JOIN Students AS S ON G.StudentID = S.ID WHERE S.ID IS NULL

UNION ALL

SELECT

'Grades - некорректные оценки',

COUNT(\*)

FROM Grades WHERE Grade < 1 OR Grade > 5 OR Grade IS NULL

UNION ALL

SELECT

'Orders - пустые клиенты',

COUNT(\*)

FROM Orders WHERE Client IS NULL OR trim(Client) = ""

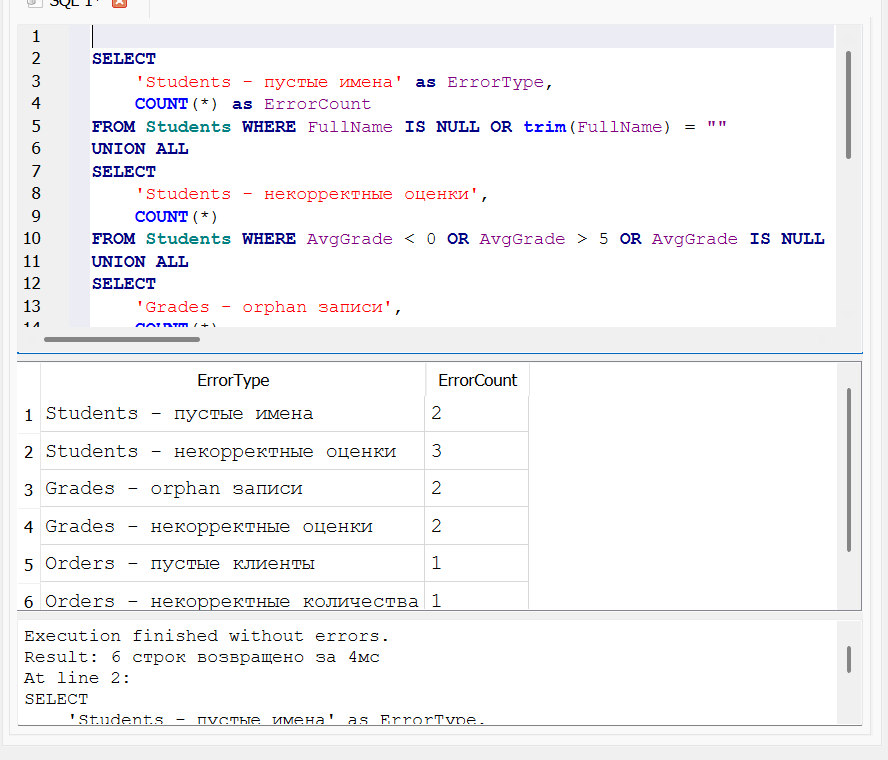
UNION ALL

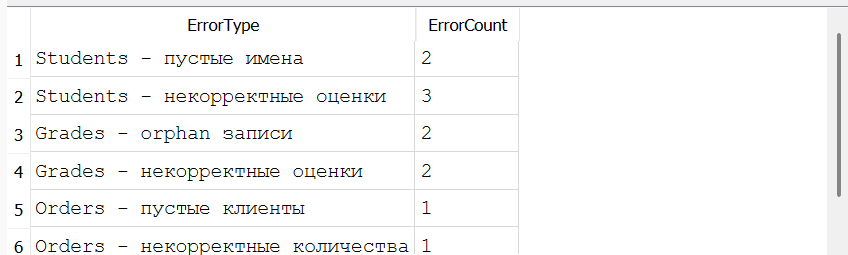
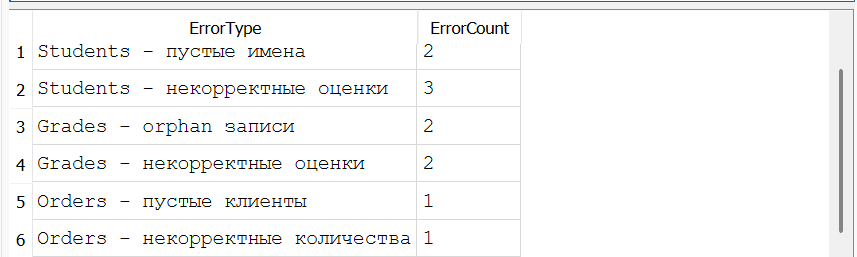
SELECT

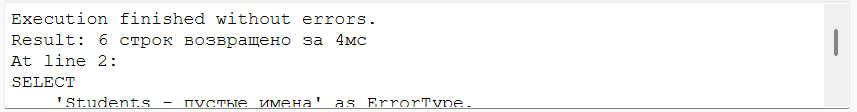
'Orders - некорректные количества',

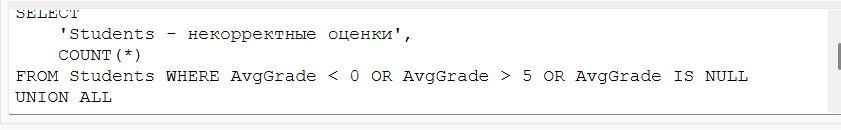
COUNT(\*)

FROM Orders WHERE Qty <= 0 OR Qty IS NULL;





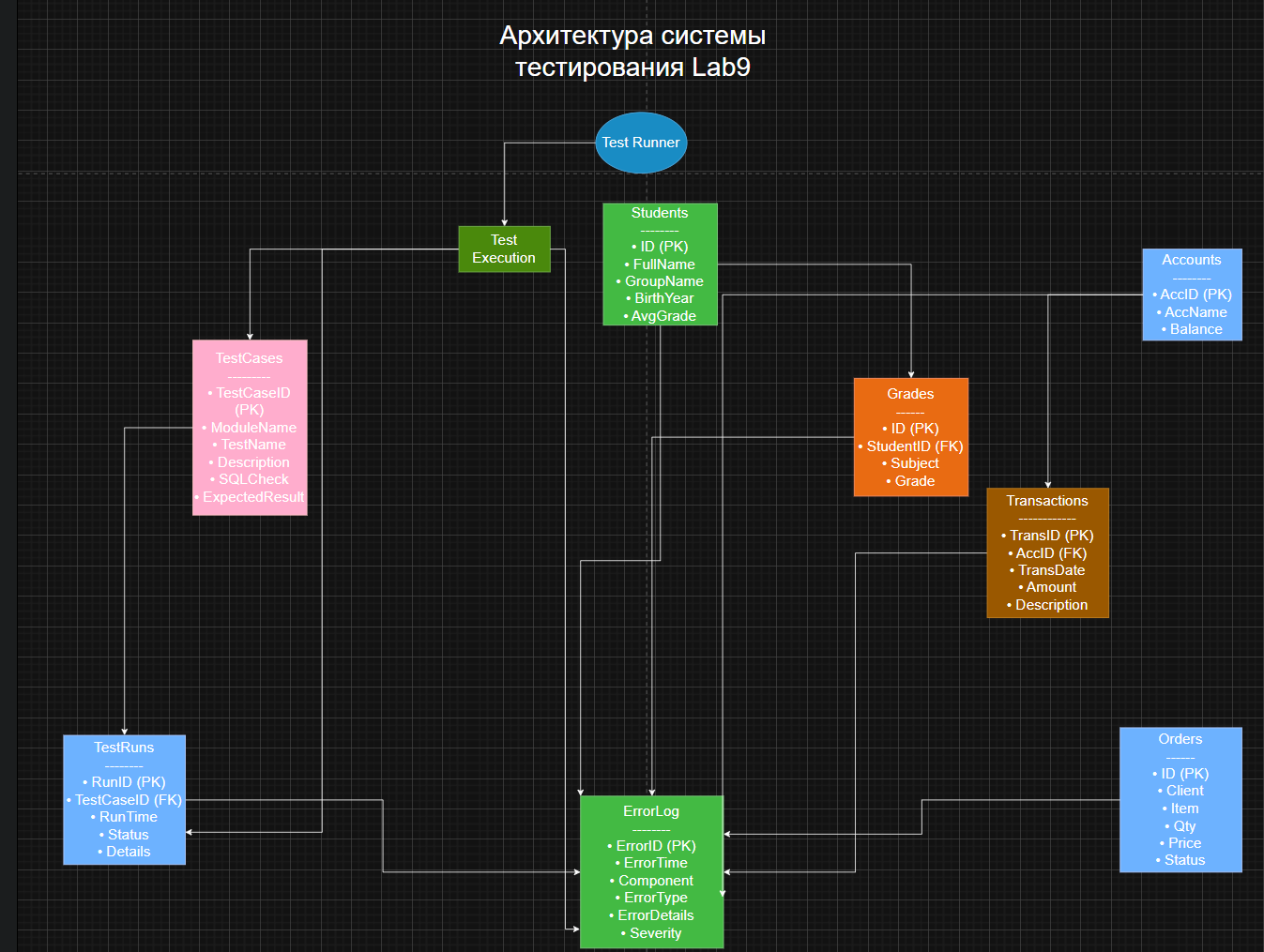


**ШАГ 22**: СОХРАНЕНИЕ РЕЗУЛЬТАТОВ

Создание диаграммы на сайте **DIAGRAMS.NET**

**Архитектура системы тестирования Lab9**

****

Ссылка на мой репозиторий с лабораторными работами 10, 11, 12, 13(8),14(9)

https://github.com/artyr989/Lab\_Exceptions\_Handling