

Министерство науки и высшего образования Российской Федерации
Муромский институт (филиал)
Федерального государственного бюджетного образовательного учреждения
высшего образования
«Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»

Факультет _____ ИТР _____

Кафедра _____ ПИН _____

ЛАБОРАТОРНАЯ РАБОТА №5

По Организация баз данных _____

Руководитель

Быков А.А.

(фамилия, инициалы)

(подпись)

(дата)

Студент _____ ПИН - 121 _____

(группа)

Ермилов М.В.

(фамилия, инициалы)

(подпись)

(дата)

Муром 2023

Лабораторная работа №5

Тема: Агрегатные функции

Цель: Изучить агрегатные функции и получить опыт работы с ними.

Задание: Функция AVG

1. Средняя стипендия студентов из заданного города.
2. Средний рейтинг университетов из города, где учится заданный студент.

Функция COUNT

1. Общее количество университетов.
2. Количество студентов из заданного университета.

Функция LIST

1. Список фамилий студентов без повторений.
2. Список фамилий студентов с заданного университета.

Функция MAX

1. Вывести студентов с максимальной оценкой.
2. Подсчитать количество студентов с максимальной оценкой.

Функция MIN

1. Вывести название университета с минимальным рейтингом и город, в котором он расположен.
2. Вывести всех студентов университета с минимальным рейтингом.

Функция SUM

1. Вывести сумму рейтингов университетов.
2. Вывести сумму оценок по заданному университету.

					МИ ВлГУ 09.03.04			
Изм.	Лист	№ докум.	Подпись	Дата				
Разраб.		Ермилов М.В.						
Провер.		Быков А.А.						
Реценз.								
Н. Контр.								
Утверд.								
						Лит.	Лист	Листов
							2	8
						ПИН-121		

Ход работы

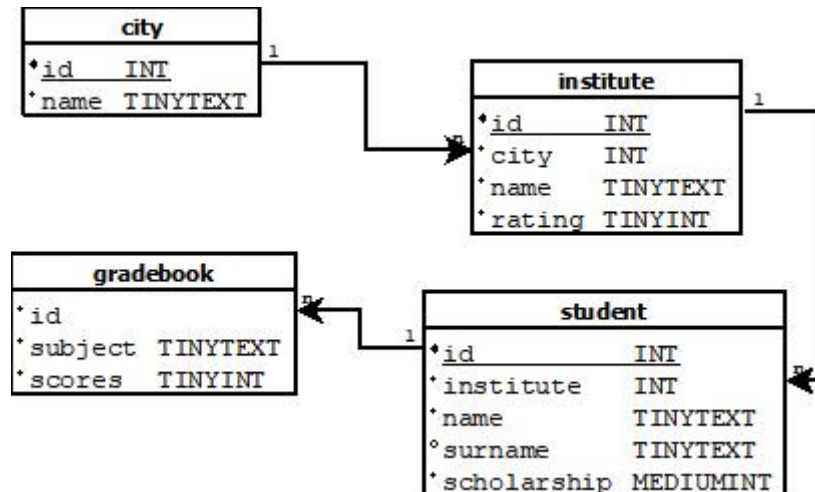


Рис 1 - схема базы данных

Листинг функций по заданию, написанных на СУБД MySQL:

```

CREATE DEFINER=`root`@`localhost` FUNCTION `AVG_1` (
    `CityName` TINYTEXT
)
RETURNS double
LANGUAGE SQL
NOT DETERMINISTIC
CONTAINS SQL
SQL SECURITY DEFINER
COMMENT ''
BEGIN
RETURN (SELECT AVG(scholarship) FROM student WHERE institute =
        ANY (SELECT id FROM institute WHERE city =
        ANY (SELECT id FROM city WHERE `name` = CityName)));
END

CREATE DEFINER=`root`@`localhost` FUNCTION `AVG_2` (
    `StudentName` TINYTEXT,
    `StudentSurname` TINYTEXT
)
RETURNS double
LANGUAGE SQL
NOT DETERMINISTIC
CONTAINS SQL
SQL SECURITY DEFINER
COMMENT ''
BEGIN
RETURN (SELECT AVG(rating) FROM institute WHERE id =
        ANY (SELECT id FROM institute WHERE city =
        ANY (SELECT city FROM institute WHERE id =
        ANY (SELECT institute FROM student WHERE `name` = StudentName AND
surname = StudentSurname))));
END

CREATE DEFINER=`root`@`localhost` FUNCTION `COUNT_1` ()
RETURNS int(11)
LANGUAGE SQL
    
```

```

NOT DETERMINISTIC
CONTAINS SQL
SQL SECURITY DEFINER
COMMENT ''
BEGIN
RETURN (SELECT COUNT(*) FROM institute);
END

CREATE DEFINER=`root`@`localhost` FUNCTION `COUNT_2` (
    `InstituteName` TINYTEXT
)
RETURNS int(11)
LANGUAGE SQL
NOT DETERMINISTIC
CONTAINS SQL
SQL SECURITY DEFINER
COMMENT ''
BEGIN
RETURN (SELECT COUNT(*) FROM student WHERE institute =
        (SELECT id FROM institute WHERE `name` = InstituteName));
END

CREATE DEFINER=`root`@`localhost` PROCEDURE `LIST_1`()
LANGUAGE SQL
NOT DETERMINISTIC
CONTAINS SQL
SQL SECURITY DEFINER
COMMENT ''
BEGIN
SELECT DISTINCT surname FROM student;
END

CREATE DEFINER=`root`@`localhost` PROCEDURE `LIST_2` (
    IN `InstituteName` TINYTEXT
)
LANGUAGE SQL
NOT DETERMINISTIC
CONTAINS SQL
SQL SECURITY DEFINER
COMMENT ''
BEGIN
SELECT surname FROM student WHERE institute = (SELECT id FROM institute WHERE
`name` = InstituteName);
END

CREATE DEFINER=`root`@`localhost` PROCEDURE `MAX_1`()
LANGUAGE SQL
NOT DETERMINISTIC
CONTAINS SQL
SQL SECURITY DEFINER
COMMENT ''
BEGIN
SELECT * FROM student WHERE id = ANY (SELECT DISTINCT id FROM gradebook WHERE
scores = (SELECT MAX(scores) FROM gradebook));
END

CREATE DEFINER=`root`@`localhost` FUNCTION `MAX_2`()
RETURNS int(11)
LANGUAGE SQL

```

					МИ ВлГУ 09.03.04	Лист
						4
Изм.	Лист	№ докум.	Подпись	Дата		

```

NOT DETERMINISTIC
CONTAINS SQL
SQL SECURITY DEFINER
COMMENT ''
BEGIN
RETURN (SELECT COUNT(DISTINCT id) FROM gradebook WHERE scores = (SELECT
MAX(scores) FROM gradebook));
END

CREATE DEFINER=`root`@`localhost` PROCEDURE `MIN_1`()
LANGUAGE SQL
NOT DETERMINISTIC
CONTAINS SQL
SQL SECURITY DEFINER
COMMENT ''
BEGIN
SELECT institute.name AS `name`, city.name AS city FROM institute
INNER JOIN city ON institute.city = city.id
WHERE institute.rating = (SELECT MIN(rating) FROM institute);
END

CREATE DEFINER=`root`@`localhost` PROCEDURE `MIN_2`(
    IN `InstituteName` TINYTEXT
)
LANGUAGE SQL
NOT DETERMINISTIC
CONTAINS SQL
SQL SECURITY DEFINER
COMMENT ''
BEGIN
DECLARE ins INT;
DECLARE scor INT;
SET @ins = (SELECT id FROM institute WHERE `name` = InstituteName);
SET @scor = (SELECT MIN(scores) FROM gradebook WHERE id = ANY (SELECT id FROM
student WHERE institute = @ins));
SELECT `name`, surname FROM student WHERE institute = @ins AND id = ANY (SELECT
id FROM gradebook WHERE scores = @scor);
END

CREATE DEFINER=`root`@`localhost` FUNCTION `SUM_1`()
RETURNS int(11)
LANGUAGE SQL
NOT DETERMINISTIC
CONTAINS SQL
SQL SECURITY DEFINER
COMMENT ''
BEGIN
RETURN (SELECT SUM(rating) FROM institute);
END

CREATE DEFINER=`root`@`localhost` FUNCTION `SUM_2`(
    `InstituteName` TINYTEXT
)
RETURNS int(11)
LANGUAGE SQL
NOT DETERMINISTIC
CONTAINS SQL
SQL SECURITY DEFINER
COMMENT ''
BEGIN

```

					МИ ВлГУ 09.03.04	Лист
						5
Изм.	Лист	№ докум.	Подпись	Дата		

```

RETURN (SELECT SUM(scores) FROM gradebook WHERE id =
        ANY (SELECT id FROM student WHERE institute =
              ANY (SELECT id FROM institute WHERE `name` = InstituteName)));
END

```

1	SELECT AVG_1('муром')
Результат #1 (1r × 1c)	
AVG_1('муром')	400

Рисунок 2 - пример работы функции

1	SELECT AVG_2('виктор', 'цой')
Результат #1 (1r × 1c)	
AVG_2('виктор', 'цой')	43

Рисунок 3 - пример работы функции

1	SELECT COUNT_1()
Результат #1 (1r × 1c)	
COUNT_1()	3

Рисунок 4 - пример работы функции

1	SELECT COUNT_2('влгу')
Результат #1 (1r × 1c)	
COUNT_2('влгу')	3

Рисунок 5 - пример работы функции

1	call LIST_1()
student (5r × 1c)	
surname	
Петров	
Малахов	
Рожков	
Назаров	
Цой	

Рисунок 6 - пример работы функции

1	call LIST_2('мкрп')
student (1r × 1c)	
surname	
Малахов	

Рисунок 7 - пример работы функции

1 call MAX_1()

student (3r × 5c)

id	name	surname	scholarship	institute
1	Василий	Петров	1 500	2
3	Кирил	Рожков	1 300	2
6	Андрей	Малахов	1 000	3

Рисунок 8 - пример работы функции

1	SELECT MAX_2()
Результат #1 (1r × 1c)	
MAX_2()	
3	

Рисунок 9 - пример работы функции

1	SELECT SUM_1()
Результат #1 (1r × 1c)	
SUM_1()	
119	

Рисунок 10 - пример работы функции

1	SELECT SUM_2('мивлгу')
Результат #1 (1r × 1c)	
SUM_2('мивлгу')	
33	

Рисунок 11 - пример работы функции

1	CALL MIN_1()
Результат #1 (1r × 2c)	
name	city
мивлгу	Муром

Рисунок 12 - пример работы функции

1	CALL MIN_2('мивлгу')
student (2r × 2c)	
name	surname
Дмитрий	Назаров
Виктор	Цой

Рисунок 13 - пример работы функции