**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**НОВОСИБИРСКИЙ  ГОСУДАРСТВЕННЫЙ  ТЕХНИЧЕСКИЙ  УНИВЕРСИТЕТ**

**КАФЕДРА ТЕОРЕТИЧЕСКОЙ И ПРИКЛАДНОЙ ИНФОРМАТИКИ**

**Лабораторная работа №** 5

**по дисциплине**   Технологии баз данных

вариант задания № 2

**Факультет: ФПМИ**

**Группа: ПМИ-72**

**Бригада № 7**

**Студенты: Антонов С.С**

**Боброва А.Н**

**Преподаватели: Стасышина Т.Л.**

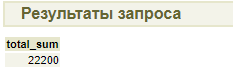
**Сивак М.А.**

**Новосибирск 2020**

1. **Запросы на языке SQL**
2. Получить минимальный вес поставки для каждого изделия и

найти их сумму.

select sum(b.minves)as total\_sum  
from (select t.n\_izd,min(t.kol\*p.ves)minves  
 from spj t  
 join p on p.n\_det = t.n\_det  
 group by t.n\_izd

) b  


1. Для указанного поставщика S\* найти число поставок каждой детали, им поставлявшейся. Вывести номер детали, город детали, название детали, число поставок детали.

Для S3

select a.n\_det,p.town,p.name, a.p\_sr

from (select t.n\_det,t.n\_post,count(\*) p\_sr

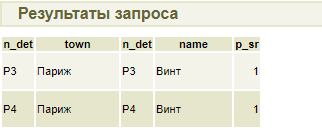
from spj t

join s s1 on t.n\_post = s1.n\_post

where s1.n\_post='S2'

group by t.n\_det

) a join p on a.n\_det = p.n\_det

order by a.n\_det  


1. Ввести номер изделия J\*. Найти поставщиков, поставлявших детали для изделия J\*, и определить, какой процент составляет объемпоставок каждого поставщика от общего количества деталей, поставленных для изделия. Вывести номер поставщика, объем поставок этогопоставщика, общий объем поставок для изделия J\*, процент.

Для J3  
 select a.n\_post, a.value, b.total\_v,

round(a.value \* 100.0 / b.total\_v,2) as procent

from ( select spj.n\_post, sum(spj.kol) as value

from spj

where spj.n\_izd = 'J3'

group by 1

) a

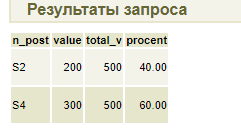
cross join ( select sum(spj.kol) as total\_v

from spj

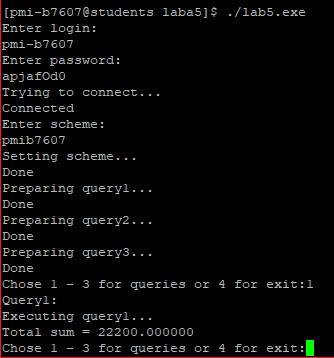
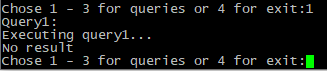
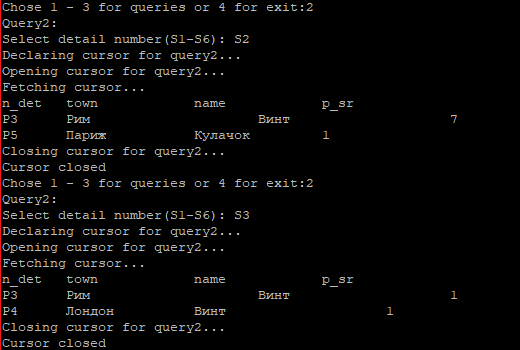
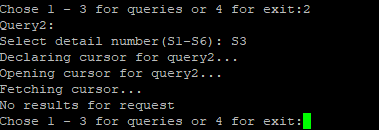
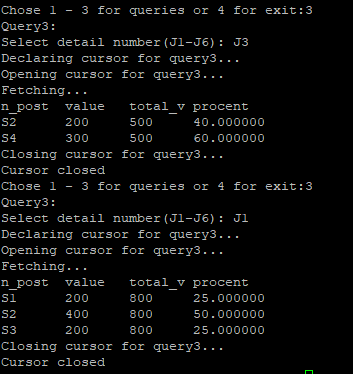
where spj.n\_izd = 'J3'

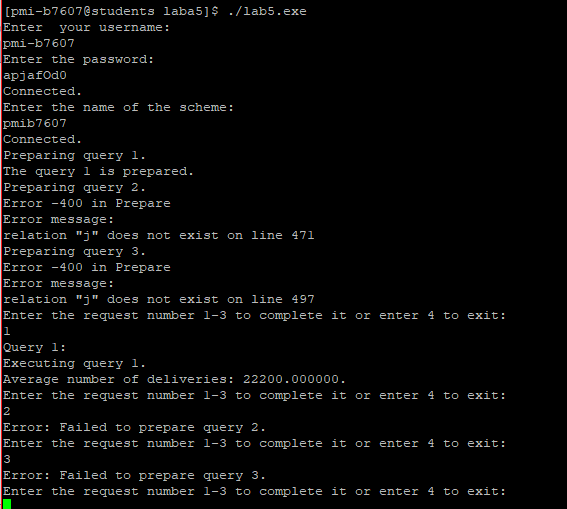
) b

order by 1



**2. Результаты на языке СИ**

1. Результат запроса   
   Запрос ничего не вернул  
   
2. Результат запроса   
     
     
   Запрос ничего не вернул  
   
3. Результат запроса  
   
4. Ошибка подготовки запроса



**3. Текст программы**#include <stdio.h>

#include <stdlib.h>

#include <sqlca.h>

int query1() {

printf("Query1:\n");

//

EXEC SQL begin declare section;

double total\_sum;

int total\_sum\_ind;

EXEC SQL end declare section;

//

EXEC SQL begin work;

printf("Executing query1...\n");

EXEC SQL execute t\_sum into : total\_sum:total\_sum\_ind;

if (sqlca.sqlcode < 0) {

errorHandler("Execute", sqlca.sqlcode);

EXEC SQL rollback work;

return 1;

}

//

EXEC SQL commit work;

if (total\_sum\_ind < 0)

printf("No result\n");

else printf("Total sum = %f\n", total\_sum);

return 0;

}

int query2() {

//

EXEC SQL begin declare section;

char n\_post[6];

char n\_det[6], town[20], name[20];

float p\_sr;

EXEC SQL end declare section;

//

printf("Query2:\nSelect detail number(S1-S6): ");

scanf("%s", n\_post);

//

printf("Declaring cursor for query2...\n");

EXEC SQL declare query2\_cursor cursor for query2\_res;

if (sqlca.sqlcode < 0) {

errorHandler("Declare", sqlca.sqlcode);

return 1;

}

//

printf("Opening cursor for query2...\n");

EXEC SQL begin work;

EXEC SQL open query2\_cursor using :n\_post;

if (sqlca.sqlcode < 0) {

errorHandler("Open cursor", sqlca.sqlcode);

EXEC SQL rollback work;

return 1;

}

//

printf("Fetching cursor...\n");

EXEC SQL fetch next from query2\_cursor into : n\_det, : town, : name, : p\_sr;

if (sqlca.sqlcode == 100) {

printf("No results for request\n");

EXEC SQL close query2\_cursor;

EXEC SQL commit work;

return 0;

}

else if (sqlca.sqlcode < 0) {

errorHandler("Fetch", sqlca.sqlcode);

EXEC SQL rollback work;

EXEC SQL close query2\_cursor;

return 1;

}

//

printf("n\_det\ttown\t\tname\t\tp\_sr\n");

do {

printf("%s\t", n\_det);

printf("%s\t", town);

printf("%s\t", name);

printf("%f\n", p\_sr);

EXEC SQL fetch next from query2\_cursor into : n\_det, : town, : name, : p\_sr;

if (sqlca.sqlcode < 0) {

errorHandler("Fetch", sqlca.sqlcode);

EXEC SQL rollback work;

EXEC SQL close query2\_cursor;

return 1;

}

} while (sqlca.sqlcode != 100);

//

printf("Closing cursor for query2...\n");

EXEC SQL close query2\_cursor;

if (sqlca.sqlcode != 0) {

errorHandler("Close cursor", sqlca.sqlcode);

return 1;

}

EXEC SQL commit work;

printf("Cursor closed\n");

return 0;

}

int query3() {

//

EXEC SQL begin declare section;

char n\_izd[6];

char n\_post[6];

int value, total\_v;

float procent;

EXEC SQL end declare section;

//

printf("Query3:\nSelect detail number(J1-J6): ");

scanf("%s", n\_izd);

//

printf("Declaring cursor for query3...\n");

EXEC SQL declare query3\_cursor cursor for query3\_res;

if (sqlca.sqlcode < 0) {

errorHandler("Declare", sqlca.sqlcode);

return 1;

}

//

printf("Opening cursor for query3...\n");

EXEC SQL begin work;

EXEC SQL open query3\_cursor using :n\_izd, : n\_izd;

if (sqlca.sqlcode < 0) {

errorHandler("Open cursor", sqlca.sqlcode);

EXEC SQL rollback work;

return 1;

}

//

printf("Fetching...\n");

EXEC SQL fetch next from query3\_cursor into : n\_post, : value, : total\_v, : procent;

if (sqlca.sqlcode == 100) {

printf("No results for query3\n");

EXEC SQL close query3\_cursor;

EXEC SQL commit work;

return 0;

}

else if (sqlca.sqlcode < 0) {

errorHandler("Fetch", sqlca.sqlcode);

EXEC SQL rollback work;

EXEC SQL close query3\_cursor;

return 1;

}

//

printf("n\_post\tvalue\ttotal\_v\tprocent\n");

do {

printf("%s\t%d\t%d\t%f\n", n\_post, value, total\_v, procent);

EXEC SQL fetch next from query3\_cursor into : n\_post, : value, : total\_v, : procent;

if (sqlca.sqlcode < 0) {

errorHandler("Fetch", sqlca.sqlcode);

EXEC SQL rollback work;

EXEC SQL close query3\_cursor;

return 1;

}

} while (sqlca.sqlcode != 100);

//

printf("Closing cursor for query3...\n");

EXEC SQL close query3\_cursor;

if (sqlca.sqlcode != 0) {

errorHandler("Close cursor", sqlca.sqlcode);

return 1;

}

EXEC SQL commit work;

printf("Cursor closed\n");

return 0;

}

void errorHandler(const char\* st\_name, int errnum) {

printf("Error code %d at %s \n", errnum, st\_name);

printf("Error message: %s\n", sqlca.sqlerrm.sqlerrmc);

return;

}

void main() {

//

EXEC SQL begin declare section;

char login[10];

char pass[10];

char scheme[10];

//для первого запроса

double total\_sum;

char text\_query1[] = "select sum(b.minves)as total\_sum from (select t.n\_izd,min(t.kol\*p.ves)minves from spj t join p on p.n\_det = t.n\_det group by t.n\_izd) b";

//для второго запроса

char n\_post[6];

char n\_det[6], name[20], town[20];

float p\_sr;

char text\_query2[] = "select a.n\_det,p.town,p.name, a.p\_sr from (select t.n\_det,t.n\_post,count(\*) p\_sr from spj t join s s1 on t.n\_post = s1.n\_post where s1.n\_post=? group by t.n\_det ) a join p on a.n\_det = p.n\_det order by a.n\_det ";

//для третьего запроса

//char n\_izd[6];

//char n\_post[6];

int value, total\_v;

float procent;

char text\_query3[] = " select a.n\_post, a.value, b.total\_v, round(a.value \* 100.0 / b.total\_v,2) as procent from ( select spj.n\_post, sum(spj.kol) as value from spj where spj.n\_izd = ? group by 1 ) a cross join ( select sum(spj.kol) as total\_v from spj where spj.n\_izd = ? ) b order by 1";

EXEC SQL end declare section;

//

printf("Enter login:\n");

scanf("%s", &login);

printf("Enter password:\n");

scanf("%s", &pass);

printf("Trying to connect...\n");

EXEC SQL connect to students@fpm2.ami.nstu.ru user : login using : pass;

//

if (sqlca.sqlcode != 0) {

errorHandler("Connect to database", sqlca.sqlcode);

return;

}

else printf("Connected\n");

//

printf("Enter scheme:\n");

scanf("%s", &scheme);

//

printf("Setting scheme...\n");

EXEC SQL set search\_path to : scheme;

if (sqlca.sqlcode != 0) {

errorHandler("Set database", sqlca.sqlcode);

return;

}

else printf("Done\n");

//

int n, flag = 0;

bool query1\_ready = true, query2\_ready = true, query3\_ready = true;

//готовим первый запрос

printf("Preparing query1...\n");

EXEC SQL prepare t\_sum from : text\_query1;

if (sqlca.sqlcode < 0) {

errorHandler("Prepare", sqlca.sqlcode);

query1\_ready = false;

}

else printf("Done\n");

//

//готовим второй запрос

printf("Preparing query2...\n");

EXEC SQL prepare query2\_res from : text\_query2;

if (sqlca.sqlcode < 0) {

errorHandler("Prepare", sqlca.sqlcode);

query2\_ready = false;

}

else printf("Done\n");

//

//готовим третий запрос

printf("Preparing query3...\n");

EXEC SQL prepare query3\_res from : text\_query3;

if (sqlca.sqlcode < 0) {

errorHandler("Prepare", sqlca.sqlcode);

query3\_ready = false;

}

else printf("Done\n");

//

do {

printf("Chose 1 - 3 for queries or 4 for exit:");

scanf("%d", &n);

switch (n) {

case 1:

if (query1\_ready) {

query1();

}

else {

printf("Can't execute query1, error occurred on prepare step\n");

}

break;

case 2:

if (query2\_ready) {

query2();

}

else {

printf("Can't execute query2, error occurred on prepare step\n");

}

break;

case 3:

if (query3\_ready) {

query3();

}

else {

printf("Can't execute query3, error occurred on prepare step\n");

}

break;

case 4:

flag = 1;

EXEC SQL disconnect all;

printf("Session closed\n");

break;

default:

printf("Chose 1-4 numbers!\n");

break;

}

} while (flag == 0);

return;

}