МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования «Полоцкий государственный университет»

Факультет информационных технологий

Кафедра технологий программирования

**ЛАБОРАТОРНАЯ РАБОТА №13**

по дисциплине: **«Базы данных»**

ВЫПОЛНИЛ студент группы 18-ИТ-3

Яковлев Н.А.

ПРОВЕРИЛ ст. преподаватель

Борисенок К.С,

Новополоцк, 2021 г.

**Цель работы**: Создать скрипт средствами языка Transact SQL в MicrosoftSQLServer2014 для внесения в нее данных.

**Ход работы**:

**Вариант 11**

Запросы/процедуры:

\*1. Упорядочение по полям: диагноз, номер палаты, дата поступления.

\*2. Поиск: все сведения о больном ХХ.

\*3. Поиск: данные о лекарстве УУ.

\*4. Выборка: больные, поступившие в день УУ с диагнозом ММ (выдать ФИО и палату).

\*5. Выборка: список выписанных на текущий день больных.

\*6. Вычисления: количество больных в палате ХХ.

\*7. Вычисления: суммарная стоимость лечения больного УУ.

\*8. Коррекция: удаление данных о больных, выписанных в прошлом году.

\*9. Коррекция: изменение цены лекарств на ХХ %.

\*10. Табличный отчет: список больных, сгруппированных по палатам (ФИО, диагноз, дата поступления, лечащий врач).

\*11. Произвольный отчет: счет за лечение (ФИО, диагноз, номер страховки и сумма за лечение).

БД Больница

клиент

- id\_клиента - INT (PK)

- id\_страхового\_полиса - INT (FK)

- id\_карты - INT (FK)

страховая\_карта

- id\_страхового\_полиса - INT (PK)

- номер\_страхового\_полиса - VARCHAR(10)

медицинская\_карта

- id\_карты - INT (PK)

- фамилия - VARCHAR(20)

- имя - VARCHAR(20)

- отчество - VARCHAR(20)

направление

- id\_направления - INT (PK)

- id\_карты - INT (FK)

- id\_палаты - INT (FK)

- id\_врача - INT (FK)

- дата\_поступления - DATE

- дата\_выписки - DATE

палата

- id\_палаты - INT (PK)

- номер\_палаты - VARCHAR(5)

- кол-во\_мест - INT

- id\_врача - INT (FK)

врач

- id\_врача - INT (PK)

- фамилия - VARCHAR(20)

- имя - VARCHAR(20)

- id\_типа - INT (FK)

тип\_врача

- id\_типа - INT (PK)

- тип\_врача - VARCHAR(20)

лечение

- id\_лечения - INT (PK)

- id\_направления - INT (FK)

лечение\_диагноз

- id\_лечения - INT (FK)

- id\_диагноза - INT (FK)

диагноз

- id\_диагноза - INT (PK)

- диагноз - VARCHAR(30)

лечение\_лекарство

- id\_лечения - INT (FK)

- id\_лекарства - INT (FK)

лекарство

- id\_лекарства - INT (PK)

- название - VARCHAR(30)

- цена - DECIMAL(10,2)

- показание - VARCHAR(45)

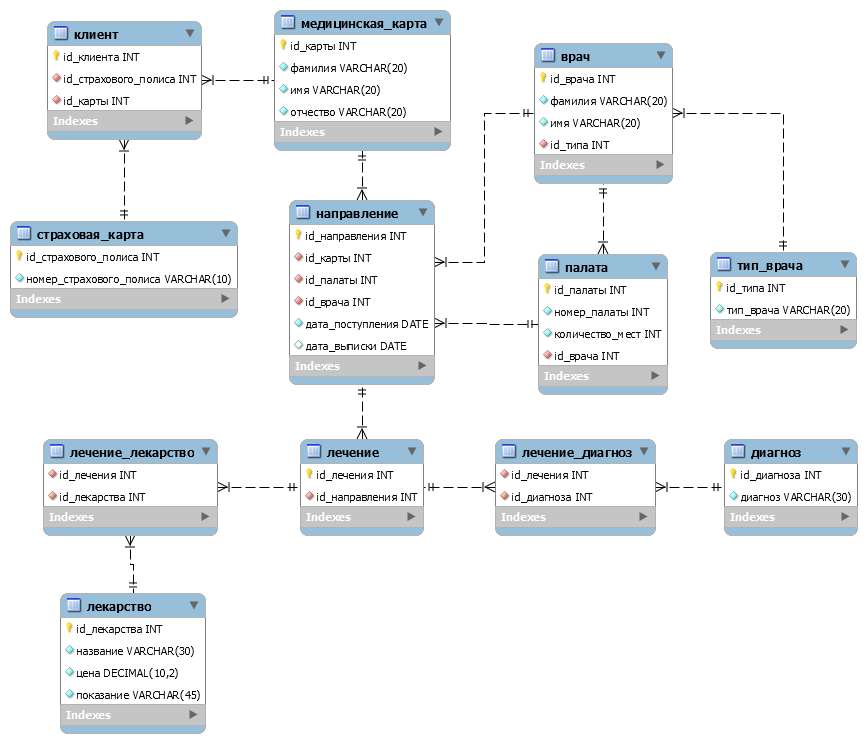


Рисунок 1 – База данных “Больница”

INSERT INTO медицинская\_карта VALUES (0, "Здитовец", "Александр", "Леонтьевич"),

(1, "Блажевич", "Евгений", "Сергеевич"),

(2, "Петричиц", "Владислав", "Витальевич"),

(3, "Смирнова", "Нина", "Сергеевна"),

(4, "Барсегян", "Артур", "Миколаевич"),

(5, "Козлов", "Андрей", "Вениаминович");

INSERT INTO страховая\_карта VALUES (0, 254321),

(1, 228228),

(2, 111111),

(3, 100200),

(4, 999999),

(5, 258852);

INSERT INTO клиент VALUES (0, 0, 0),

(1, 1, 1),

(2, 2, 2),

(3, 3, 3),

(4, 4, 4),

(5, 5, 5);

INSERT INTO тип\_врача VALUES (0, "лечащий"),

(1, "направляющий");

INSERT INTO врач VALUES (0, "Марковец", "Дмитрий", 0),

(1, "Пупкин", "Василий", 0),

(2, "Тахмазов", "Рустам", 1),

(3, "Долманова", "Екатерина", 0);

INSERT INTO палата VALUES (0, 101, 4, 0),

(1, 102, 2, 0),

(2, 103, 2, 1),

(3, 201, 4, 1),

(4, 202, 2, 3),

(5, 203, 2, 3),

(6, 301, 4, 3),

(7, 302, 2, 1),

(8, 303, 2, 0);

INSERT INTO направление VALUES (0, 0, 1, 2, '2019-10-01', '2019-10-22'),

(1, 1, 0, 2, '2019-10-18', '2019-10-25'),

(2, 2, 1, 2, '2019-10-19', '2019-11-05'),

(3, 1, 2, 2, '2019-11-01', '2019-11-15'),

(4, 3, 5, 2, '2019-11-01', '2019-11-29'),

(5, 4, 7, 2, '2019-11-02', '2019-11-18'),

(6, 0, 8, 2, '2019-11-02', '2019-11-08'),

(7, 0, 4, 2, '2019-11-25', '2020-01-12'),

(8, 0, 0, 2, '2020-01-19', '2020-02-10'),

(9, 3, 6, 2, '2020-01-22', '2020-02-19'),

(10, 2, 6, 2, '2020-02-05', null),

(11, 5, 3, 2, '2020-02-18', null);

INSERT INTO лечение VALUES (0, 0),

(1, 1),

(2, 2),

(3, 3),

(4, 4),

(5, 5),

(6, 6),

(7, 7),

(8, 8),

(9, 9),

(10, 10),

(11, 11);

INSERT INTO диагноз VALUES (0, "воспаление сустава"),

(1, "дерматит"),

(2, "остеохондроз"),

(3, "ангина"),

(4, "бронхит"),

(5, "гастрит");

INSERT INTO лечение\_диагноз VALUES (0, 3),

(1, 3),

(2, 1),

(3, 5),

(4, 0),

(5, 1),

(6, 2),

(7, 3),

(8, 4),

(9, 5),

(10, 4),

(11, 4);

INSERT INTO лекарство VALUES (0, "Диклофенак Натрия", 5, "против воспаление сустава"),

(1, "Индометацин", 3, "против воспаление сустава, при сильных болей"),

(2, "Дексалгин", 4, "против воспаление сустава, при легких болей"),

(3, "Бепантен", 8, "против дерматита, при чесотке"),

(4, "Азакс", 2, "против дерматита"),

(5, "Амизон", 5, "против остеохондроза, когда-нибудь"),

(6, "Артифлекс", 4, "против остеохондроза, при случае"),

(7, "Денебол", 5, "против остеохондроза"),

(8, "Цетил", 8, "против ангины"),

(9, "Азитромицин", 9, "против ангины, при температуре"),

(10, "Лоросан", 7, "против ангины, при сухом кашле"),

(11, "Бисептол", 2, "против бронхита"),

(12, "Флуимуцил", 10, "против бронхита, для облегчения"),

(13, "Цетил", 15, "против бронхита, при тяжолом случае"),

(14, "Алмагель", 20, "против гастрита, при тяжолом случае"),

(15, "Но-шпа", 2, "против гастрита");

INSERT INTO лечение\_лекарство VALUES (0, 8),

(0, 10),

(1, 9),

(1, 10),

(2, 3),

(3, 14),

(4, 0),

(4, 2),

(5, 3),

(5, 4),

(6, 5),

(7, 9),

(8, 12),

(8, 13),

(9, 14),

(9, 15),

(10, 11),

(10, 12),

(11, 13);

Запросы/процедуры:

**\*1. Упорядочение по полям: диагноз, номер палаты, дата поступления.**

SELECT

`a`.`диагноз` AS `Диагноз`,

`e`.`номер\_палаты` AS `Номер палаты`,

`d`.`дата\_поступления` AS `Дата поступления`

FROM

((((`больница\_лаба\_13`.`диагноз` `a`

INNER JOIN `больница\_лаба\_13`.`лечение\_диагноз` `b` ON ((`a`.`id\_диагноза` = `b`.`id\_диагноза`)))

INNER JOIN `больница\_лаба\_13`.`лечение` `c` ON ((`b`.`id\_лечения` = `c`.`id\_лечения`)))

INNER JOIN `больница\_лаба\_13`.`направление` `d` ON (c.id\_направления = d.id\_направления))

INNER JOIN `больница\_лаба\_13`.`палата` `e` ON (d.id\_палаты = e.id\_палаты))

ORDER BY `a`.`диагноз` , `e`.`номер\_палаты` , `d`.`дата\_поступления`;

**\*2. Поиск: все сведения о больном ХХ.**

CREATE DEFINER=`root`@`localhost` PROCEDURE `new\_procedure\_02`(IN Surname VARCHAR(20), IN Firstname VARCHAR(20))

BEGIN

SELECT c.фамилия "Фамилия", c.имя "Имя", c.отчество "Отчество", a.номер\_страхового\_полиса "Номер страховки", COUNT(d.id\_карты) "Кол-во лечений", GROUP\_CONCAT(g.диагноз) AS `История болезней`

FROM ((((((страховая\_карта a

INNER JOIN клиент b ON (a.id\_страхового\_полиса = b.id\_страхового\_полиса))

INNER JOIN медицинская\_карта c ON (b.id\_карты = c.id\_карты))

INNER JOIN направление d ON (c.id\_карты = d.id\_карты))

INNER JOIN лечение e ON (d.id\_направления = e.id\_направления))

INNER JOIN лечение\_диагноз f ON (e.id\_лечения = f.id\_лечения))

INNER JOIN диагноз g ON (f.id\_диагноза = g.id\_диагноза))

WHERE c.фамилия = Surname AND c.имя = Firstname

GROUP BY d.id\_карты;

END

**\*3. Поиск: данные о лекарстве УУ.**

CREATE DEFINER=`root`@`localhost` PROCEDURE `new\_procedure\_03`(IN Medicine VARCHAR(30))

BEGIN

SELECT a.название "Название", a.цена "Цена", a.показание "Показание", COUNT(b.id\_лекарства) "Кол-во использований"

FROM ((лекарство a

INNER JOIN лечение\_лекарство b ON (a.id\_лекарства = b.id\_лекарства))

INNER JOIN лекарство c ON (b.id\_лекарства = c.id\_лекарства))

WHERE c.название = Medicine

GROUP BY b.id\_лекарства;

END

**\*4. Выборка: больные, поступившие в день УУ с диагнозом ММ (выдать ФИО и палату).**

CREATE DEFINER=`root`@`localhost` PROCEDURE `new\_procedure\_04`(IN DateYY DATE, IN Diagnosis VARCHAR(30))

BEGIN

SELECT a.фамилия "Фамилия", a.имя "Имя", a.отчество "Отчество", c.номер\_палаты "Палата"

FROM (((((медицинская\_карта a

INNER JOIN направление b ON (a.id\_карты = b.id\_карты))

INNER JOIN палата c ON (b.id\_палаты = c.id\_палаты))

INNER JOIN лечение d ON (b.id\_направления = d.id\_направления))

INNER JOIN лечение\_диагноз e ON (d.id\_лечения = e.id\_лечения))

INNER JOIN диагноз f ON (e.id\_диагноза = f.id\_диагноза))

WHERE b.дата\_поступления = DateYY AND f.диагноз = Diagnosis;

END

**5. Выборка: список выписанных на текущий день больных.**

CREATE

ALGORITHM = UNDEFINED

DEFINER = `root`@`localhost`

SQL SECURITY DEFINER

VIEW `больница\_лаба\_13`.`new\_view\_05` AS

SELECT

`a`.`фамилия` AS `Фамилия`,

`a`.`имя` AS `Имя`,

`a`.`отчество` AS `Отчество`

FROM

(`больница\_лаба\_13`.`медицинская\_карта` `a`

JOIN `больница\_лаба\_13`.`направление` `b` ON ((`a`.`id\_карты` = `b`.`id\_карты`)))

WHERE

`b`.`id\_карты` IN (SELECT

`больница\_лаба\_13`.`направление`.`id\_карты`

FROM

`больница\_лаба\_13`.`направление`

WHERE

(`больница\_лаба\_13`.`направление`.`дата\_выписки` IS NULL))

IS FALSE

GROUP BY `b`.`id\_карты`

**\*6. Вычисления: количество больных в палате ХХ.**

CREATE DEFINER=`root`@`localhost` PROCEDURE `new\_procedure\_06`(IN RoomNumber INT)

BEGIN

SELECT RoomNumber "Палата", COUNT(b.номер\_палаты) "Кол-во больных"

FROM направление a

INNER JOIN палата b ON (a.id\_палаты = b.id\_палаты)

WHERE b.номер\_палаты = RoomNumber

AND a.id\_карты IN (SELECT

`больница\_лаба\_13`.`направление`.`id\_карты`

FROM

`больница\_лаба\_13`.`направление`

WHERE

(`больница\_лаба\_13`.`направление`.`дата\_выписки` IS NULL))

IS TRUE

GROUP BY b.номер\_палаты ;

END

**\*7. Вычисления: суммарная стоимость лечения больного УУ.**

CREATE DEFINER=`root`@`localhost` PROCEDURE `new\_procedure\_07`(IN Surname VARCHAR(20),IN Firstname VARCHAR(20))

BEGIN

SELECT e.фамилия "Фамилия", e.имя "Имя", SUM(d.цена) "Сумма"

FROM ((((медицинская\_карта e

INNER JOIN направление a ON (e.id\_карты = a.id\_карты))

INNER JOIN лечение b ON (a.id\_направления = b.id\_направления))

INNER JOIN лечение\_лекарство c ON (b.id\_лечения = c.id\_лечения))

INNER JOIN лекарство d ON (c.id\_лекарства = d.id\_лекарства))

WHERE e.фамилия = Surname AND e.имя = Firstname

GROUP BY e.id\_карты;

END

**\*8. Коррекция: удаление данных о больных, выписанных в прошлом году.**

CREATE DEFINER=`root`@`localhost` PROCEDURE `new\_procedure\_08`()

BEGIN

DELETE FROM `больница\_лаба\_13`.лечение\_лекарство a WHERE a.id\_лечения = (SELECT id\_лечения FROM лечение WHERE id\_направления = (SELECT id\_направления FROM направление WHERE (TO\_DAYS(CURDATE()) - TO\_DAYS(дата\_выписки)) > 365));

DELETE FROM `больница\_лаба\_13`.лечение\_диагноз a WHERE a.id\_лечения = (SELECT id\_лечения FROM лечение WHERE id\_направления = (SELECT id\_направления FROM направление WHERE (TO\_DAYS(CURDATE()) - TO\_DAYS(дата\_выписки)) > 365));

DELETE FROM `больница\_лаба\_13`.лечение WHERE id\_направления = (SELECT id\_направления FROM направление WHERE (TO\_DAYS(CURDATE()) - TO\_DAYS(дата\_выписки)) > 365);

DELETE FROM `больница\_лаба\_13`.направление WHERE (TO\_DAYS(CURDATE()) - TO\_DAYS(дата\_выписки)) > 365;

END

**\*9. Коррекция: изменение цены лекарств на ХХ %.**

CREATE DEFINER=`root`@`localhost` PROCEDURE `new\_procedure\_09`(IN Percent INT)

BEGIN

UPDATE лекарство a SET

a.цена = a.цена\*Percent/100;

END

**10. Табличный отчет: список больных, сгруппированных по палатам (ФИО, диагноз, дата поступления, лечащий врач).**

CREATE

ALGORITHM = UNDEFINED

DEFINER = `root`@`localhost`

SQL SECURITY DEFINER

VIEW `больница\_лаба\_13`.`new\_view\_10` AS

SELECT

CONCAT(`a`.`фамилия`,

' ',

`a`.`имя`,

' ',

`a`.`отчество`) AS `ФИО`,

`e`.`диагноз` AS `Диагноз`,

`b`.`дата\_поступления` AS `Дата поступления`,

CONCAT(`g`.`фамилия`, ' ', `g`.`имя`) AS `Врач`

FROM

((((((`больница\_лаба\_13`.`медицинская\_карта` `a`

JOIN `больница\_лаба\_13`.`направление` `b` ON ((`a`.`id\_карты` = `b`.`id\_карты`)))

JOIN `больница\_лаба\_13`.`лечение` `c` ON ((`b`.`id\_направления` = `c`.`id\_направления`)))

JOIN `больница\_лаба\_13`.`лечение\_диагноз` `d` ON ((`c`.`id\_лечения` = `d`.`id\_лечения`)))

JOIN `больница\_лаба\_13`.`диагноз` `e` ON ((`d`.`id\_диагноза` = `e`.`id\_диагноза`)))

JOIN `больница\_лаба\_13`.`палата` `f` ON ((`b`.`id\_палаты` = `f`.`id\_палаты`)))

JOIN `больница\_лаба\_13`.`врач` `g` ON ((`f`.`id\_врача` = `g`.`id\_врача`)))

WHERE

`b`.`id\_направления` IN (SELECT

`больница\_лаба\_13`.`направление`.`id\_направления`

FROM

`больница\_лаба\_13`.`направление`

WHERE

(`больница\_лаба\_13`.`направление`.`дата\_выписки` IS NULL))

**11. Произвольный отчет: счет за лечение (ФИО, диагноз, номер страховки и сумма за лечение).**

CREATE

ALGORITHM = UNDEFINED

DEFINER = `root`@`localhost`

SQL SECURITY DEFINER

VIEW `больница\_лаба\_13`.`new\_view\_11` AS

SELECT

CONCAT(`c`.`фамилия`,

' ',

`c`.`имя`,

' ',

`c`.`отчество`) AS `ФИО`,

`g`.`диагноз` AS `Диагноз`,

`a`.`номер\_страхового\_полиса` AS `Номер Страховки`,

SUM(`j`.`цена`) AS `Сумма за лечния`

FROM

((((((((`больница\_лаба\_13`.`страховая\_карта` `a`

JOIN `больница\_лаба\_13`.`клиент` `b` ON ((`a`.`id\_страхового\_полиса` = `b`.`id\_страхового\_полиса`)))

JOIN `больница\_лаба\_13`.`медицинская\_карта` `c` ON ((`b`.`id\_карты` = `c`.`id\_карты`)))

JOIN `больница\_лаба\_13`.`направление` `d` ON ((`c`.`id\_карты` = `d`.`id\_карты`)))

JOIN `больница\_лаба\_13`.`лечение` `e` ON ((`d`.`id\_направления` = `e`.`id\_направления`)))

JOIN `больница\_лаба\_13`.`лечение\_диагноз` `f` ON ((`e`.`id\_лечения` = `f`.`id\_лечения`)))

JOIN `больница\_лаба\_13`.`диагноз` `g` ON ((`f`.`id\_диагноза` = `g`.`id\_диагноза`)))

JOIN `больница\_лаба\_13`.`лечение\_лекарство` `h` ON ((`e`.`id\_лечения` = `h`.`id\_лечения`)))

JOIN `больница\_лаба\_13`.`лекарство` `j` ON ((`h`.`id\_лекарства` = `j`.`id\_лекарства`)))

GROUP BY `c`.`id\_карты`