

**ГОСУДАРСТВЕННОЕ УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ  
«ИНСТИТУТ БИЗНЕСА БЕЛОРУССКОГО ГОСУДАРСТВЕННОГО  
УНИВЕРСИТЕТА»**

**КАФЕДРА ЦИФРОВЫХ СИСТЕМ И ТЕХНОЛОГИЙ**

**ОТЧЕТ**

**о выполнении Лабораторной работы №18**

**«Описание структуры базы данных в среде пакета автоматизированного  
проектирования информационных систем»  
по дисциплине «Проектирование информационных систем»**

Севрюк Александры Петровны  
студентки 4 курса, группа 852  
специальность «Управление  
информационными ресурсами»

Минск, 2021

## Оглавление

1. Цель работы.....	3
2. Алгоритм формирования новой структуры данных .....	3
3. Объектная модель .....	3
4. Диаграмма модели данных .....	4
5. Триггер.....	4
6. Хранимая процедура.....	5
7. DLL Script.....	5
8. Результат обратного проектирования .....	9
9. Ответы на контрольные вопросы .....	9
10. Выводы .....	11

## 1. Цель работы

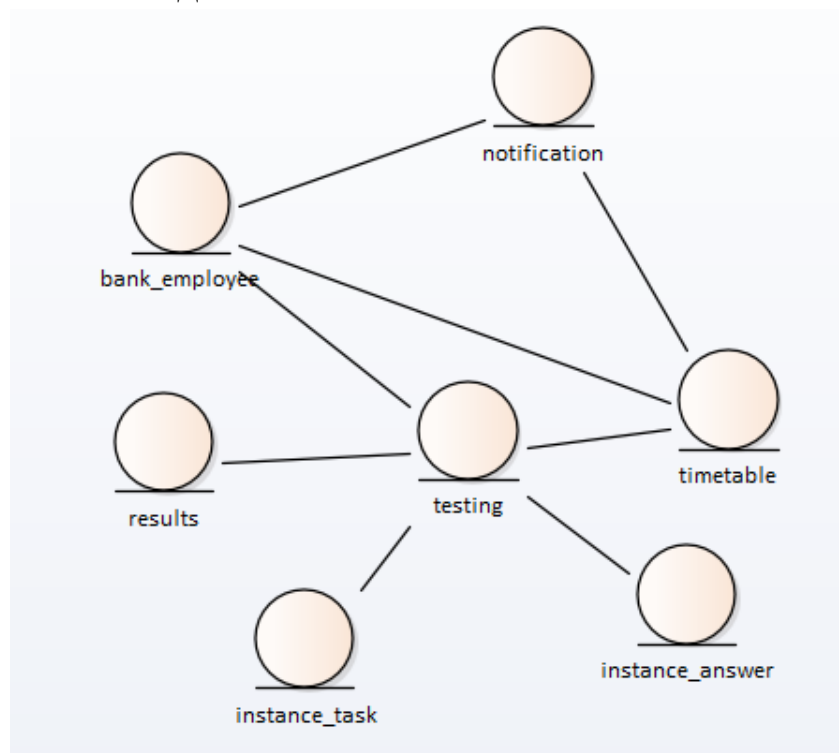
Целью данной лабораторной работы является изучение технологии round-trip для проектирования структуры базы данных, а также приобретение практических навыков проектирования структуры базы данных.

## 2. Алгоритм формирования новой структуры данных

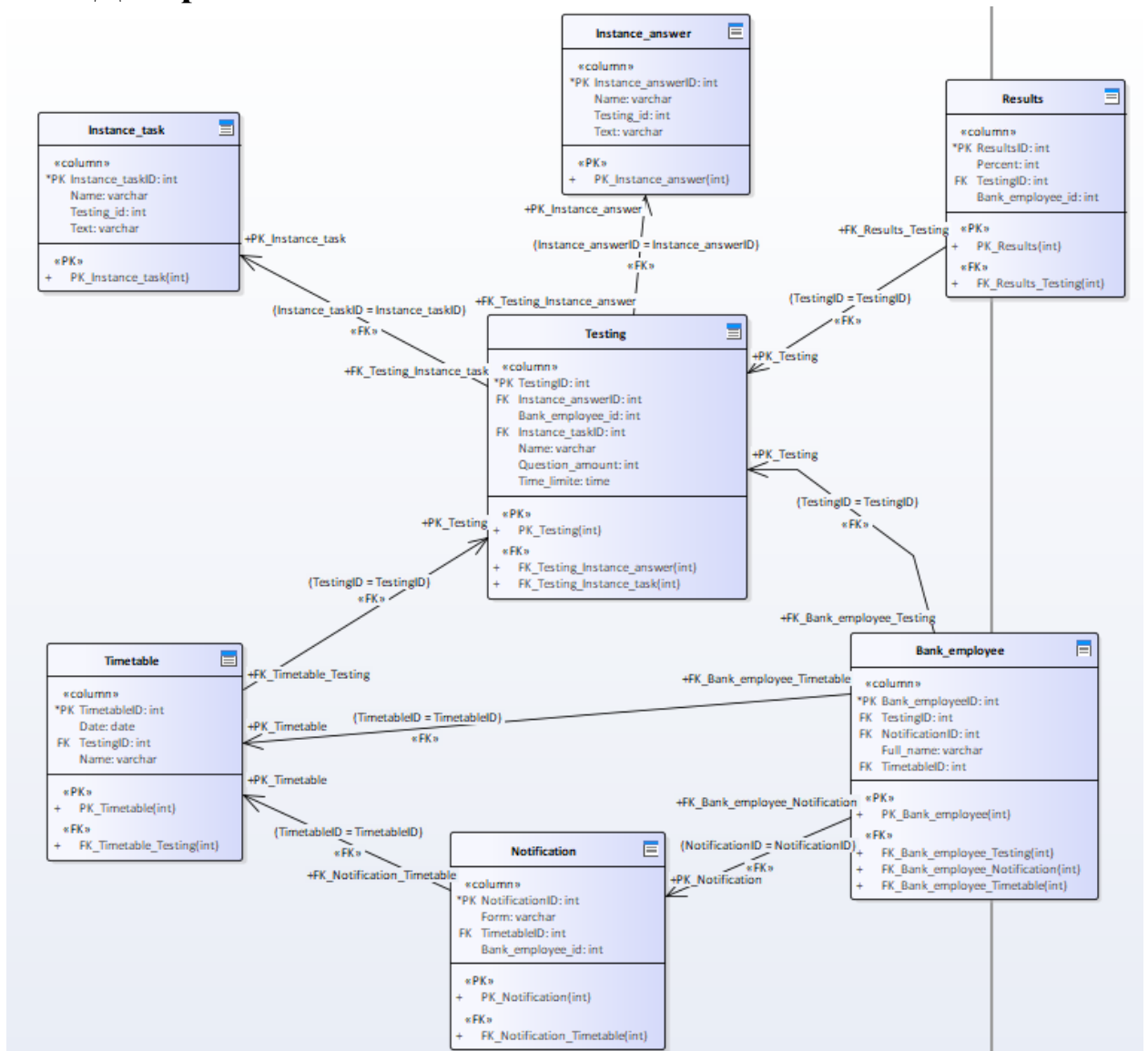
Шаги формирования новой структуры данных:

1. Нажимаем правой кнопкой на сформированную папку «Entity»
2. Переходим в пункт Advanced
3. Нажимаем Transform Package
4. Выбираем DDL
5. Нажимаем Do transform
6. Нажимаем правой кнопкой на папку DDL
7. Выбираем Code Engineering
8. Нажимаем Generate DDL
9. Выбираем место сохранения
10. Нажимаем Generate

## 3. Объектная модель



## 4. Диаграмма модели данных



## 5. Триггер

Был разработан триггер, запрещающий модификацию данных в столбце «Percent» таблицы «Results».

```

CREATE TRIGGER tr_NoModifyResultsPercent
ON Warehouses
FOR UPDATE AS
IF UPDATE(Percent)
BEGIN
PRINT 'Обновление столбца Percent запрещено'
ROLLBACK TRAN
END
GO
    
```

## 6. Хранимая процедура

Была разработана хранимая процедура, которая будет вставлять новые строки в таблицу «Timetable».

```
CREATE PROCEDURE pr_TimetableInsertion
@TimetableID INT,
@Date DATE,
@Name VARCHAR(50),
@TestingID INT
AS
INSERT INTO Timetable
VALUES (@TimetableID, @Date, @Name, @TestingID)
GO
```

## 7. DDL Script

```
/* ----- */
/* Generated by Enterprise Architect Version 13.0 */
/* Created On : 27-ноя-2021 15:43:25 */
/* DBMS : SQL Server 2012 */
/* ----- */

/* Drop Foreign Key Constraints */

IF EXISTS (SELECT 1 FROM dbo.sysobjects WHERE id =
object_id(N'[FK_Bank_employee_Testing]') AND OBJECTPROPERTY(id, N'IsForeignKey') = 1)
ALTER TABLE [Bank_employee] DROP CONSTRAINT [FK_Bank_employee_Testing]
GO

IF EXISTS (SELECT 1 FROM dbo.sysobjects WHERE id =
object_id(N'[FK_Bank_employee_Notification]') AND OBJECTPROPERTY(id, N'IsForeignKey') =
1)
ALTER TABLE [Bank_employee] DROP CONSTRAINT [FK_Bank_employee_Notification]
GO

IF EXISTS (SELECT 1 FROM dbo.sysobjects WHERE id =
object_id(N'[FK_Bank_employee_Timetable]') AND OBJECTPROPERTY(id, N'IsForeignKey') = 1)
ALTER TABLE [Bank_employee] DROP CONSTRAINT [FK_Bank_employee_Timetable]
GO

IF EXISTS (SELECT 1 FROM dbo.sysobjects WHERE id =
object_id(N'[FK_Notification_Timetable]') AND OBJECTPROPERTY(id, N'IsForeignKey') = 1)
ALTER TABLE [Notification] DROP CONSTRAINT [FK_Notification_Timetable]
GO

IF EXISTS (SELECT 1 FROM dbo.sysobjects WHERE id = object_id(N'[FK_Results_Testing]') AND
OBJECTPROPERTY(id, N'IsForeignKey') = 1)
ALTER TABLE [Results] DROP CONSTRAINT [FK_Results_Testing]
GO

IF EXISTS (SELECT 1 FROM dbo.sysobjects WHERE id =
object_id(N'[FK_Testing_Instance_answer]') AND OBJECTPROPERTY(id, N'IsForeignKey') = 1)
ALTER TABLE [Testing] DROP CONSTRAINT [FK_Testing_Instance_answer]
GO

IF EXISTS (SELECT 1 FROM dbo.sysobjects WHERE id =
object_id(N'[FK_Testing_Instance_task]') AND OBJECTPROPERTY(id, N'IsForeignKey') = 1)
ALTER TABLE [Testing] DROP CONSTRAINT [FK_Testing_Instance_task]
```

```

GO

IF EXISTS (SELECT 1 FROM dbo.sysobjects WHERE id = object_id(N'[FK_Timetable_Testing]')
AND OBJECTPROPERTY(id, N'IsForeignKey') = 1)
ALTER TABLE [Timetable] DROP CONSTRAINT [FK_Timetable_Testing]
GO

/* Drop Tables */

IF EXISTS (SELECT 1 FROM dbo.sysobjects WHERE id = object_id(N'[Bank_employee]') AND
OBJECTPROPERTY(id, N'IsUserTable') = 1)
DROP TABLE [Bank_employee]
GO

IF EXISTS (SELECT 1 FROM dbo.sysobjects WHERE id = object_id(N'[Instance_answer]') AND
OBJECTPROPERTY(id, N'IsUserTable') = 1)
DROP TABLE [Instance_answer]
GO

IF EXISTS (SELECT 1 FROM dbo.sysobjects WHERE id = object_id(N'[Instance_task]') AND
OBJECTPROPERTY(id, N'IsUserTable') = 1)
DROP TABLE [Instance_task]
GO

IF EXISTS (SELECT 1 FROM dbo.sysobjects WHERE id = object_id(N'[Notification]') AND
OBJECTPROPERTY(id, N'IsUserTable') = 1)
DROP TABLE [Notification]
GO

IF EXISTS (SELECT 1 FROM dbo.sysobjects WHERE id = object_id(N'[Results]') AND
OBJECTPROPERTY(id, N'IsUserTable') = 1)
DROP TABLE [Results]
GO

IF EXISTS (SELECT 1 FROM dbo.sysobjects WHERE id = object_id(N'[Testing]') AND
OBJECTPROPERTY(id, N'IsUserTable') = 1)
DROP TABLE [Testing]
GO

IF EXISTS (SELECT 1 FROM dbo.sysobjects WHERE id = object_id(N'[Timetable]') AND
OBJECTPROPERTY(id, N'IsUserTable') = 1)
DROP TABLE [Timetable]
GO

/* Create Tables */

CREATE TABLE [Bank_employee]
(
    [Bank_employeeID] int NOT NULL,
    [TestingID] int NULL,
    [NotificationID] int NULL,
    [Full_name] varchar(50) NULL,
    [TimetableID] int NULL
)
GO

CREATE TABLE [Instance_answer]
(
    [Instance_answerID] int NOT NULL,
    [Name] varchar(50) NULL,
    [Testing_id] int NULL,
    [Text] varchar(1000) NULL
)
GO

```

```

CREATE TABLE [Instance_task]
(
    [Instance_taskID] int NOT NULL,
    [Name] varchar(50) NULL,
    [Testing_id] int NULL,
    [Text] varchar(1000) NULL
)
GO

CREATE TABLE [Notification]
(
    [NotificationID] int NOT NULL,
    [Form] varchar(50) NULL,
    [TimetableID] int NULL,
    [Bank_employee_id] int NULL
)
GO

CREATE TABLE [Results]
(
    [ResultsID] int NOT NULL,
    [Percent] int NULL,
    [TestingID] int NULL,
    [Bank_employee_id] int NULL
)
GO

CREATE TABLE [Testing]
(
    [TestingID] int NOT NULL,
    [Instance_answerID] int NULL,
    [Bank_employee_id] int NULL,
    [Instance_taskID] int NULL,
    [Name] varchar(50) NULL,
    [Question_amount] int NULL,
    [Time_limite] time(7) NULL
)
GO

CREATE TABLE [Timetable]
(
    [TimetableID] int NOT NULL,
    [Date] date NULL,
    [TestingID] int NULL,
    [Name] varchar(50) NULL
)
GO

/* Create Primary Keys, Indexes, Uniques, Checks */

ALTER TABLE [Bank_employee]
ADD CONSTRAINT [PK_Bank_employee]
PRIMARY KEY CLUSTERED ([Bank_employeeID] ASC)
GO

ALTER TABLE [Instance_answer]
ADD CONSTRAINT [PK_Instance_answer]
PRIMARY KEY CLUSTERED ([Instance_answerID] ASC)
GO

ALTER TABLE [Instance_task]
ADD CONSTRAINT [PK_Instance_task]
PRIMARY KEY CLUSTERED ([Instance_taskID] ASC)

```

```

GO

ALTER TABLE [Notification]
ADD CONSTRAINT [PK_Notification]
PRIMARY KEY CLUSTERED ([NotificationID] ASC)
GO

ALTER TABLE [Results]
ADD CONSTRAINT [PK_Results]
PRIMARY KEY CLUSTERED ([ResultsID] ASC)
GO

ALTER TABLE [Testing]
ADD CONSTRAINT [PK_Testing]
PRIMARY KEY CLUSTERED ([TestingID] ASC)
GO

ALTER TABLE [Timetable]
ADD CONSTRAINT [PK_Timetable]
PRIMARY KEY CLUSTERED ([TimetableID] ASC)
GO

/* Create Foreign Key Constraints */

ALTER TABLE [Bank_employee] ADD CONSTRAINT [FK_Bank_employee_Testing]
FOREIGN KEY ([TestingID]) REFERENCES [Testing] ([TestingID]) ON DELETE No Action
ON UPDATE No Action
GO

ALTER TABLE [Bank_employee] ADD CONSTRAINT [FK_Bank_employee_Notification]
FOREIGN KEY ([NotificationID]) REFERENCES [Notification] ([NotificationID]) ON
DELETE No Action ON UPDATE No Action
GO

ALTER TABLE [Bank_employee] ADD CONSTRAINT [FK_Bank_employee_Timetable]
FOREIGN KEY ([TimetableID]) REFERENCES [Timetable] ([TimetableID]) ON DELETE No
Action ON UPDATE No Action
GO

ALTER TABLE [Notification] ADD CONSTRAINT [FK_Notification_Timetable]
FOREIGN KEY ([TimetableID]) REFERENCES [Timetable] ([TimetableID]) ON DELETE No
Action ON UPDATE No Action
GO

ALTER TABLE [Results] ADD CONSTRAINT [FK_Results_Testing]
FOREIGN KEY ([TestingID]) REFERENCES [Testing] ([TestingID]) ON DELETE No Action
ON UPDATE No Action
GO

ALTER TABLE [Testing] ADD CONSTRAINT [FK_Testing_Instance_answer]
FOREIGN KEY ([Instance_answerID]) REFERENCES [Instance_answer]
([Instance_answerID]) ON DELETE No Action ON UPDATE No Action
GO

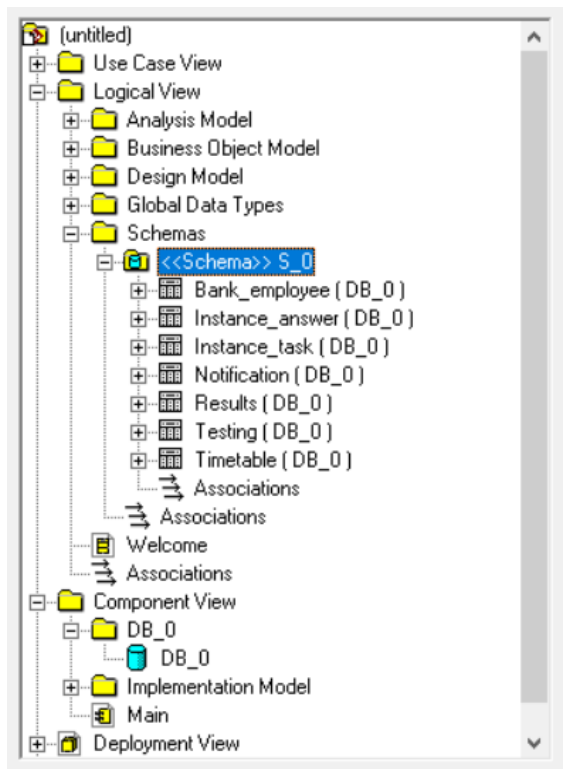
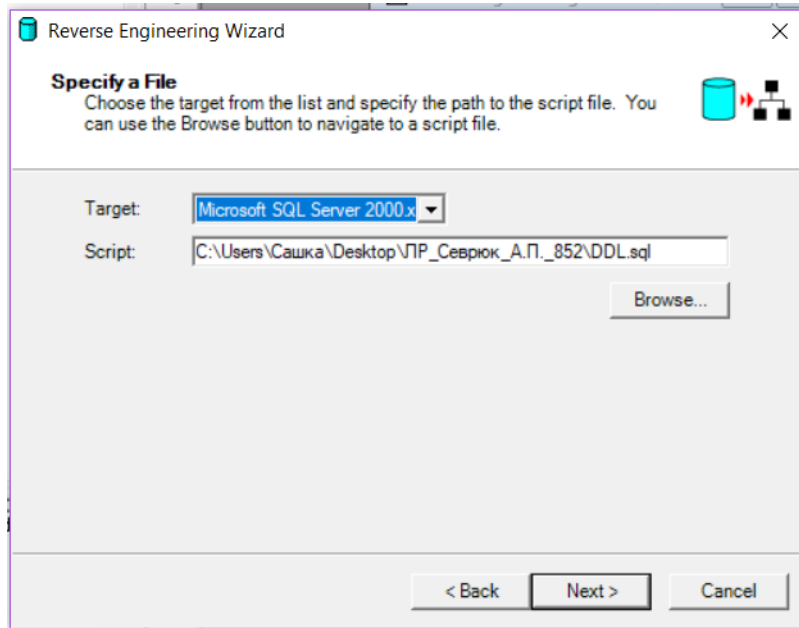
ALTER TABLE [Testing] ADD CONSTRAINT [FK_Testing_Instance_task]
FOREIGN KEY ([Instance_taskID]) REFERENCES [Instance_task] ([Instance_taskID]) ON
DELETE No Action ON UPDATE No Action
GO

ALTER TABLE [Timetable] ADD CONSTRAINT [FK_Timetable_Testing]
FOREIGN KEY ([TestingID]) REFERENCES [Testing] ([TestingID]) ON DELETE No Action
ON UPDATE No Action
GO

```



## 8. Результат обратного проектирования



## 9. Ответы на контрольные вопросы

### 1. Для чего используется Data Modeler?

Data Modeler используется для моделирования физического описания данных, он позволяет создавать все необходимые объекты базы данных: таблицы, триггеры, хранимые процедуры и представления данных.

### 2. Какие системы управления базами данных поддерживает Data Modeler?

Data Modeler поддерживает работу с основными системами обработки баз данных: IBM DB2 MVS, UDB, Oracle, Microsoft SQL Server, Sybase Adaptive Server.

3. Стоит задача модернизировать эксплуатируемую ИС. Поясните, как использовать существующую базу данных для построения проекта модернизации ИС.

На основе анализа существующей базы данных разработать и внедрить предложения по ее модернизации.

4. Каков порядок создания новой структуры данных?

Если необходимо создать совершенно новую структуру данных, и не существует готовой базы данных для обратного моделирования, то нужно проделать следующие шаги:

1. Создать компонент база данных
2. Создать схему данных.
3. Создать таблицы данных в схеме.
4. Создать связи данных.
5. Перенести созданные объекты в базу данных.

Создаем новый проект в Rational Rose, затем создаем в Component View проекта компонент DataBase, указываем целевую СУБД - MS SQL Server 2000.x В Logical View проекта создаем схему, в ней создаем диаграмму схемы данных. На диаграмме данных создаем таблицы и связываем их.

5. Для чего служит схема данных?

Схема - основной контейнер, для создания модели данных. Первое, что потребуется для создания модели, - это создание схемы. В схеме определяются таблицы, столбцы таблиц, ключи, ограничения целостности для обеспечения ссылочной целостности модели.

6. Как описать встроенную процедуру в проекте создания ИС?

В схеме логического представления нашего проекта создаем новую хранимую процедуру (Stored Procedure), в спецификациях которой задаем имя, выбираем язык (SQL) и в закладке Action Body набираем непосредственно тело процедуры в Action Body.

7. Можно ли с использованием созданного DDL файла сформировать новую базу данных?

Да, можно. Производим импорт данных из базы данных SQL Server. Эту операцию можно выполнять, непосредственно соединившись с базой и с

использованием DDL файла. Создаем модель данных проекта в Rational Rose с использованием DDL файла.

## **10. Выводы**

В результате выполнения данной лабораторной работы я изучила технологию round-trip для проектирования структуры базы данных, а также приобрела практические навыки проектирования структуры базы данных.