**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ**

**БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**Факультет прикладной математики и информатики**

**Ястребова Вероника**

**Генерирование кода проектируемого программного обеспечения и разработка приложения**

Отчет по лабораторной работе №6

Вариант 9

(«Проектирование программных систем»)

студентки 2 курса 13 группы

**Преподаватель**

**Давидовская М.И**

**Минск 2018**

1. **Описание предметной области**

**АИС «Справочник абитуриента»**

Описание предметной области.

Высшее учебное заведения для предоставления на сайте информации абитуриентам ведет банк данных со следующей информацией:

* список специальностей (Код, Название, Краткое название), на которых осуществляется обучение в вузе. Специальности привязаны к учебным подразделениям – факультетам, кафедрам (Код, Название, Краткое название), и распределены по формам обучения (очная, очно-заочная, заочная);
* адрес учебных подразделений;
* телефоны учебных подразделений;
* если есть – адрес сайта учебного подразделения;
* ФИО, ученая степень, ученое звание руководителя учебного заведения (декан факультета, заведующий кафедрой). При этом необходимо вести историю всех руководителей – дата начала работы, дата окончания; -
* по каждой форме обучения:
  + план приема на специальность на каждый год;
  + перечень предметов, по которым необходимо сдавать вступительные экзамены (ЕГЭ);
  + проходной балл на специальность по годам с разбивкой по предметам.

Необходимо осуществлять следующую обработку данных:

* на заданный год – список специальностей заданной формы обучения и планы приема;
* на заданный год наименование специальности, на которую был максимальный проходной балл по математике;
* на заданный год список руководителей учебных подразделений, имеющих ученую степень «доктор наук» и ученое звание «профессор».

**2. Примеры сгенерированного кода**

* **Класс LoginController**

///////////////////////////////////////////////////////////

// LoginController.cs

// Implementation of the Class LoginController

// Generated by Enterprise Architect

// Created on: 23-май-2018 13:49:22

// Original author: Veronika

///////////////////////////////////////////////////////////

**using** **System**;

**using** **System.Collections.Generic**;

**using** **System.Text**;

**using** **System.IO**;

**using** **DesignModel**;

**namespace** **LoginClassDiagram** {

**public** **class** **LoginController** {

///

/// <param name="login"></param>

**public** **bool** **checkLogin**(**string** login){

**return** **false**;

}

///

/// <param name="password"></param>

**public** **bool** **checkPassword**(**string** password){

**return** **false**;

}

**public** DesignModel.TypeOfUser makeType(){

**return** **null**;

}

}//end <<controller>> LoginController

}//end namespace LoginClassDiagram

* Класс ShowingInfo

///////////////////////////////////////////////////////////

// ShowingInfo.cs

// Implementation of the Class ShowingInfo

// Generated by Enterprise Architect

// Created on: 23-май-2018 13:53:50

// Original author: Veronika

///////////////////////////////////////////////////////////

**using** **System**;

**using** **System.Collections.Generic**;

**using** **System.Text**;

**using** **System.IO**;

**using** **DesignModel**;

**namespace** **RequestClassDiagram** {

/// <summary>

/// Вывод информации на экран

/// </summary>

**public** **class** **ShowingInfo** {

///

/// <param name="year"></param>

**public** **void** **showListOfLeaders**(**int** year){

}

///

/// <param name="year"></param>

/// <param name="form"></param>

**public** **void** **showListOfSpecialties**(**int** year, FormOfEducation form){

}

///

/// <param name="year"></param>

/// <param name="form"></param>

**public** **void** **showPlanOfAdmission**(**int** year, FormOfEducation form){

}

///

/// <param name="year"></param>

**public** **void** **showSpecialityMaxScore**(**int** year){

}

}//end ShowingInfo

}//end namespace RequestClassDiagram

**3. Sql-сценарий базы данных**

/\* ---------------------------------------------------- \*/

/\* Generated by Enterprise Architect Version 13.5 \*/

/\* Created On : 23-май-2018 14:45:15 \*/

/\* DBMS : SQLite \*/

/\* ---------------------------------------------------- \*/

/\* Drop Tables \*/

**DROP** **TABLE** IF **EXISTS** 'ContactInfo'

;

**DROP** **TABLE** IF **EXISTS** 'Department'

;

**DROP** **TABLE** IF **EXISTS** 'Faculty'

;

**DROP** **TABLE** IF **EXISTS** 'FormOfEducation'

;

**DROP** **TABLE** IF **EXISTS** 'Leader'

;

**DROP** **TABLE** IF **EXISTS** 'Speciality'

;

**DROP** **TABLE** IF **EXISTS** 'Subject'

;

**DROP** **TABLE** IF **EXISTS** 'TypeOfUser'

;

**DROP** **TABLE** IF **EXISTS** 'User'

;

/\* Create Tables with Primary and Foreign Keys, Check and Unique Constraints \*/

**CREATE** **TABLE** 'ContactInfo'

(

'InfoID' INTEGER **NOT** **NULL** **PRIMARY** **KEY**,

'Adress' TEXT **NOT** **NULL**,

'PhoneNumber' TEXT **NOT** **NULL**,

'Website' TEXT **NULL**,

'SpecID' INTEGER **NULL**,

**CONSTRAINT** 'FK\_ContactInfo\_Speciality' **FOREIGN** **KEY** ('SpecID') **REFERENCES** 'Speciality' ('SpecID') **ON** **DELETE** **No** Action **ON** **UPDATE** **No** Action

)

;

**CREATE** **TABLE** 'Department'

(

'DepID' INTEGER **NOT** **NULL** **PRIMARY** **KEY**,

'DepCode' INTEGER **NOT** **NULL**,

'DepName' TEXT **NOT** **NULL**,

'DepShortName' TEXT **NOT** **NULL**

)

;

**CREATE** **TABLE** 'Faculty'

(

'FacID' INTEGER **NOT** **NULL** **PRIMARY** **KEY**,

'FacCode' INTEGER **NOT** **NULL**,

'FacName' TEXT **NOT** **NULL**,

'FacShortName' TEXT **NOT** **NULL**,

'DepID' INTEGER **NULL**,

**CONSTRAINT** 'FK\_Faculty\_Department' **FOREIGN** **KEY** ('DepID') **REFERENCES** 'Department' ('DepID') **ON** **DELETE** **No** Action **ON** **UPDATE** **No** Action

)

;

**CREATE** **TABLE** 'FormOfEducation'

(

'FormID' INTEGER **NOT** **NULL** **PRIMARY** **KEY**,

'Type' INTEGER **NOT** **NULL**,

'SubjID' INTEGER **NULL**,

**CONSTRAINT** 'FK\_FormOfEducation\_Subject' **FOREIGN** **KEY** ('SubjID') **REFERENCES** 'Subject' ('SubjID') **ON** **DELETE** **No** Action **ON** **UPDATE** **No** Action

)

;

**CREATE** **TABLE** 'Leader'

(

'LeaderID' INTEGER **NOT** **NULL** **PRIMARY** **KEY**,

'Name' TEXT **NOT** **NULL**,

'ScDegree' TEXT **NOT** **NULL**,

'ScRank' TEXT **NOT** **NULL**,

'StartDate' TEXT **NOT** **NULL**,

'EndDate' TEXT **NULL**,

'SpecID' INTEGER **NULL**,

'DepID' INTEGER **NULL**,

'FacID' INTEGER **NULL**,

**CONSTRAINT** 'FK\_Leader\_Department' **FOREIGN** **KEY** ('DepID') **REFERENCES** 'Department' ('DepID') **ON** **DELETE** **No** Action **ON** **UPDATE** **No** Action,

**CONSTRAINT** 'FK\_Leader\_Faculty' **FOREIGN** **KEY** ('FacID') **REFERENCES** 'Faculty' ('FacID') **ON** **DELETE** **No** Action **ON** **UPDATE** **No** Action,

**CONSTRAINT** 'FK\_Leader\_Speciality' **FOREIGN** **KEY** ('SpecID') **REFERENCES** 'Speciality' ('SpecID') **ON** **DELETE** **No** Action **ON** **UPDATE** **No** Action

)

;

**CREATE** **TABLE** 'Speciality'

(

'SpecID' INTEGER **NOT** **NULL** **PRIMARY** **KEY**,

'Name' TEXT **NOT** **NULL**,

'ShortName' TEXT **NOT** **NULL**,

'Code' INTEGER **NOT** **NULL**,

'FacID' INTEGER **NULL**,

'FormID' INTEGER **NULL**,

'DepID' INTEGER **NULL**,

**CONSTRAINT** 'FK\_Speciality\_Department' **FOREIGN** **KEY** ('DepID') **REFERENCES** 'Department' ('DepID') **ON** **DELETE** **No** Action **ON** **UPDATE** **No** Action,

**CONSTRAINT** 'FK\_Speciality\_Faculty' **FOREIGN** **KEY** ('FacID') **REFERENCES** 'Faculty' ('FacID') **ON** **DELETE** **No** Action **ON** **UPDATE** **No** Action,

**CONSTRAINT** 'FK\_Speciality\_FormOfEducation' **FOREIGN** **KEY** ('FormID') **REFERENCES** 'FormOfEducation' ('FormID') **ON** **DELETE** **No** Action **ON** **UPDATE** **No** Action

)

;

**CREATE** **TABLE** 'Subject'

(

'SubjID' INTEGER **NOT** **NULL** **PRIMARY** **KEY**,

'PassingScore' NUMERIC **NOT** **NULL**,

'Year' INTEGER **NOT** **NULL**

)

;

**CREATE** **TABLE** 'TypeOfUser'

(

'TypeID' INTEGER **NOT** **NULL** **PRIMARY** **KEY**,

'Admin' TEXT **NOT** **NULL**,

'Aplicant' TEXT **NOT** **NULL**,

'UnversityWorker' TEXT **NOT** **NULL**

)

;

**CREATE** **TABLE** 'User'

(

'UserId' INTEGER **NOT** **NULL** **PRIMARY** **KEY**,

'Login' TEXT **NOT** **NULL**,

'Password' TEXT **NOT** **NULL**,

'TypeID' INTEGER **NOT** **NULL**,

**CONSTRAINT** 'FK\_User\_TypeOfUser' **FOREIGN** **KEY** ('TypeID') **REFERENCES** 'TypeOfUser' ('TypeID') **ON** **DELETE** **No** Action **ON** **UPDATE** **No** Action

)

;

/\* Create Indexes and Triggers \*/

**CREATE** **INDEX** 'IXFK\_ContactInfo\_Speciality'

**ON** 'ContactInfo' ('SpecID' **ASC**)

;

**CREATE** **INDEX** 'IXFK\_Faculty\_Department'

**ON** 'Faculty' ('DepID' **ASC**)

;

**CREATE** **INDEX** 'IXFK\_FormOfEducation\_Subject'

**ON** 'FormOfEducation' ('SubjID' **ASC**)

;

**CREATE** **INDEX** 'IXFK\_Leader\_Department'

**ON** 'Leader' ('DepID' **ASC**)

;

**CREATE** **INDEX** 'IXFK\_Leader\_Faculty'

**ON** 'Leader' ('FacID' **ASC**)

;

**CREATE** **INDEX** 'IXFK\_Leader\_Speciality'

**ON** 'Leader' ('SpecID' **ASC**)

;

**CREATE** **INDEX** 'IXFK\_Speciality\_Department'

**ON** 'Speciality' ('DepID' **ASC**)

;

**CREATE** **INDEX** 'IXFK\_Speciality\_Faculty'

**ON** 'Speciality' ('FacID' **ASC**)

;

**CREATE** **INDEX** 'IXFK\_Speciality\_FormOfEducation'

**ON** 'Speciality' ('FormID' **ASC**)

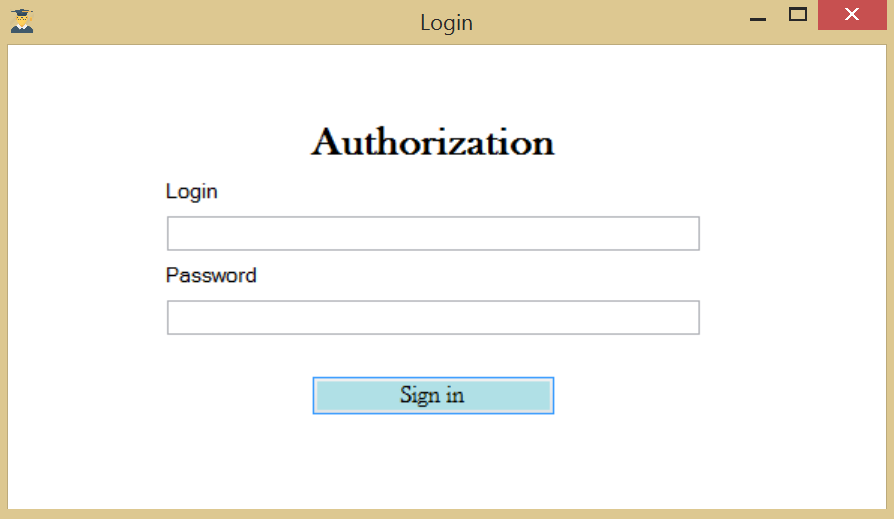
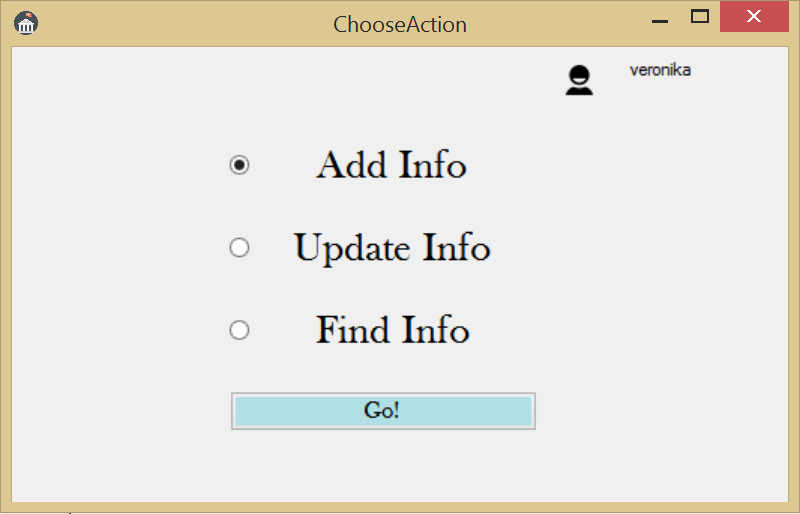
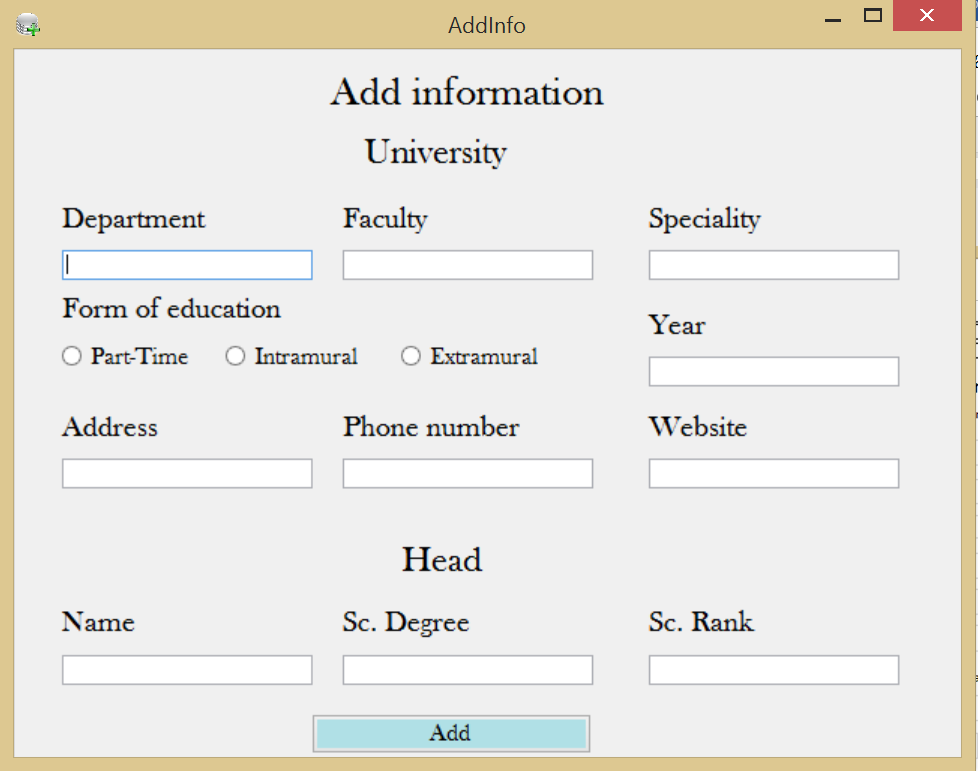
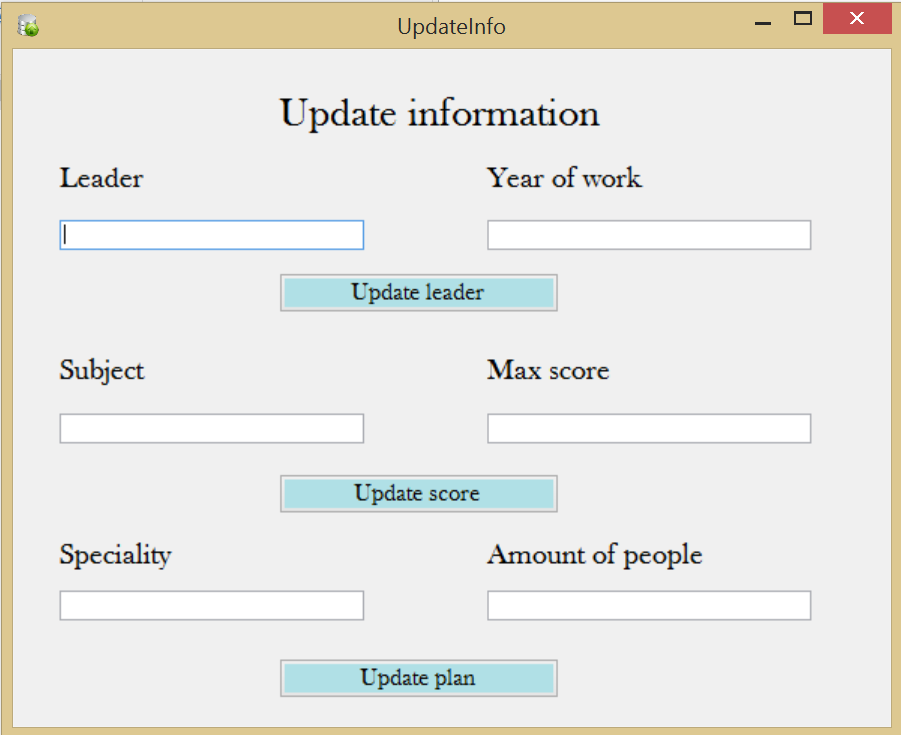
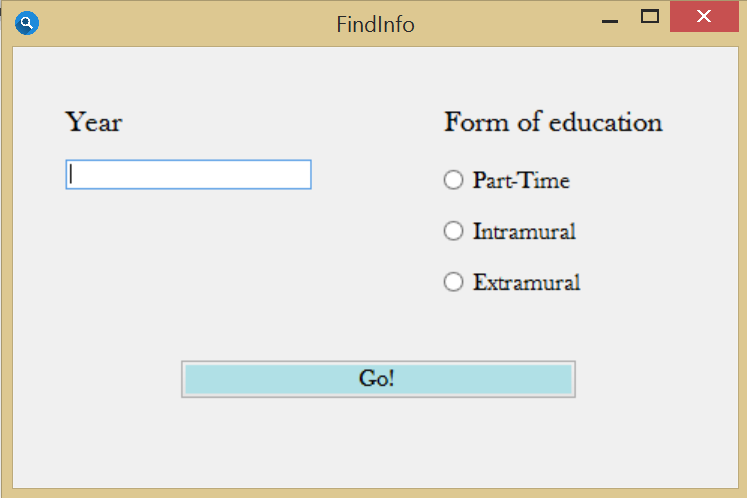
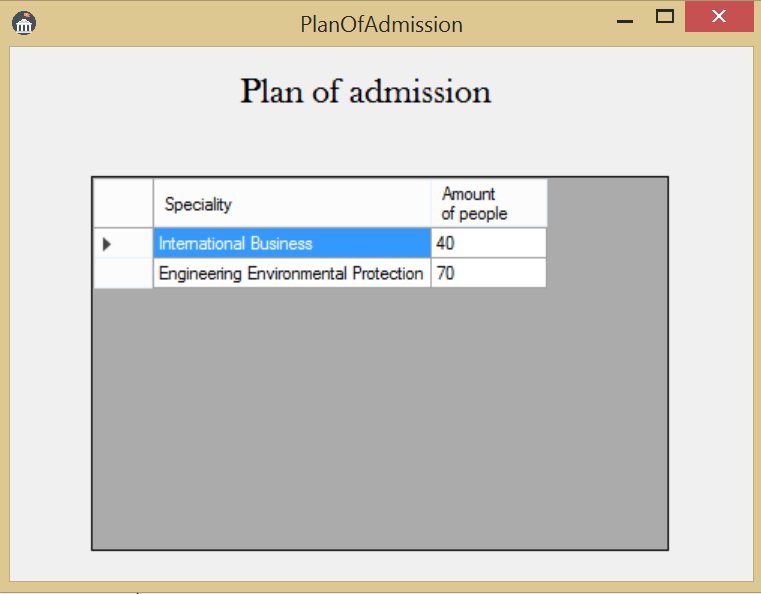
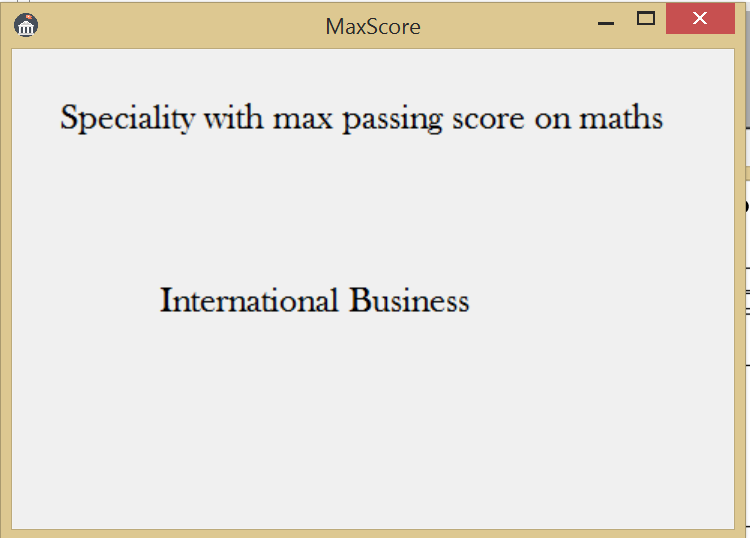
;

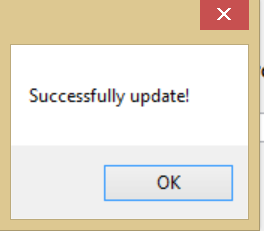
**CREATE** **INDEX** 'IXFK\_User\_TypeOfUser'

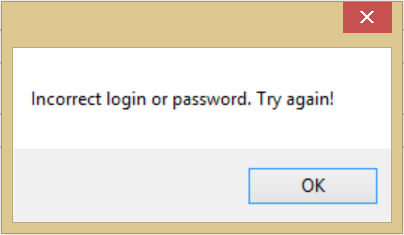
**ON** 'User' ('TypeID' **ASC**)

;

**4. Скриншоты экранов приложения**

* **Авторизация**
* **Выбор действия** 
* **Добавление информации** 
* **Обновление информации** 
* **Ввод информации для поиска** 
* **Выбор информации для просмотра** 
* **Просмотр списка руководителей** 
* **Просмотр плана приема и списка специальностей** 
* **Просмотр названия специальности с максимальным проходным баллом по математике** 
* **Сообщение об успешном/неуспешном выполнении операции**



* **Сообщение об ошибке** 

**Список литературы**

1. **“Пример задачи”** https://drive.google.com/file/d/1KInO1zcFC1m2EC7pzb-W9\_zYy5\_Kjg7v/view
2. **«Диаграммы пакетов» http://www.planerka.info/item/Diagrammy-paketov**
3. **«Диаграмма развертывания»** http://www.planerka.info/item/Diagrammy-razvertyvaniya-UML
4. **«Диаграмма компонентов»** http://www.planerka.info/item/Diagrammy-komponentov-UML