

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО
ОБРАЗОВАНИЯ

САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ

Факультет «Инфокоммуникационных технологий»

Направление подготовки «09.03.03 Мобильные и сетевые технологии»

О Т Ч Е Т

ЛАБОРАТОРНАЯ РАБОТА №4

Тема задания: SQL запросы

Выполнил:

Студент _____ Гончаров В.А. _____ К3240
(Фамилия И.О.) номер группы

Проверил:

Преподаватель _____ Говоров А.И.
(Фамилия И.О.)

Санкт-Петербург

2020

Постановка задания: создать программную систему, предназначенную для работников приемной комиссии колледжа. Она должна обеспечивать хранение, просмотр и изменение сведений об абитуриентах. Результатом работы приемной комиссии должен быть список абитуриентов, зачисленных в колледж.

Секретарь приемной комиссии регистрирует абитуриентов. Для каждого абитуриента в базу данных заносятся следующие сведения: фамилия, имя, отчество, паспортные данные, какое учебное заведение, где и когда окончил, наличие золотой или серебряной медали, название специальности, на которые поступает абитуриент. При подаче заявления абитуриент указывает форму обучения (очная, очно-заочная (вечерняя), заочная), поступление на бюджет или контракт. Абитуриент может поступать вне конкурса (инвалиды, сироты). Также существуют абитуриенты-целевики, которые поступают по договорам с направляющими организациями, и обучаются на коммерческой основе.

Абитуриенты, поступающие на базе 9 классов, участвуют в конкурсе аттестатов. Для них указывается информация по 4-м профильным дисциплинам и средний балл по всем остальным дисциплинам аттестата. На основе этих данных строится рейтинг абитуриентов.

Абитуриенты, поступающие на базе 11 классов, предоставляют сертификаты ЕГЭ по 2 дисциплинам, на основе чего строится рейтинг абитуриентов. Конкурс для абитуриентов на базе 9 и 11 классов отдельный, т.к. они поступают на разные курсы. Абитуриент может не только подать, но и забрать документы, а также перевести их на другую специальность.

Известно количество мест на каждый факультет. Приемная комиссия по результатам экзаменов должна сформировать списки абитуриентов, зачисленных в колледж.

Схема БД: схема БД расположена в приложенном файле с названием «Приложение»

Запросы:

1. Выбор значений, заданных атрибутов из более, чем двух таблиц, с сортировкой – от 1 балла;

Вывод абитуриентов и их паспортных данных

```
1 select enroll."Enrollee"."ID", enroll."Enrollee"."Name", enroll."Passport"."Number", enroll."Passport"."Date" from enroll."Enrollee"  
2 inner join enroll."Passport" on enroll."Enrollee"."PassportNumber" = enroll."Passport"."Number"  
3 order by enroll."Enrollee"."PassportNumber";
```

	ID integer	Name character varying (100)	Number character (10)	Date date
1	3	Putin Vladimir	1337322333	2015-03...
2	1	Zubenko Michael Petrovich	3222444444	2015-04...
3	2	Zmishenko Valery	3222554444	2015-04...
4	4	Banketniy Vladimir	3277654444	2015-04...
5	5	Udmurt Vladimir	3277658224	2015-04...

2. Использование условий WHERE, состоящих из более, чем одного условия – от 1 балла;

Вывод всех абитуриентов и их паспортных данных, кроме абитуриента «Putin Vladimir» и абитуриента, чей номер паспорта «3277658224»

```
1 select enroll."Enrollee"."ID", enroll."Enrollee"."Name", enroll."Passport"."Number", enroll."Passport"."Date" from enroll."Enrollee"
2   inner join enroll."Passport" on enroll."Enrollee"."PassportNumber" = enroll."Passport"."Number"
3 where enroll."Enrollee"."Name" != 'Putin Vladimir' and enroll."Passport"."Number" != '3277658224'
4 order by enroll."Enrollee"."PassportNumber";
```

	ID integer	Name character varying (100)	Number character (10)	Date date
1	1	Zubenko Michael Petrovich	3222444444	2015-04...
2	2	Zmishenko Valery	3222554444	2015-04...
3	4	Banketniy Vladimir	3277654444	2015-04...

3. Использование функций для работы с датами – от 2 баллов;

Вывод абитуриентов, номера их паспорта и времени, прошедшего со дня выдачи

```
1 select enroll."Enrollee"."ID", enroll."Enrollee"."Name", enroll."Passport"."Number", age(current_date, enroll."Passport"."Date") from enroll."Enrollee"
2   inner join enroll."Passport" on enroll."Enrollee"."PassportNumber" = enroll."Passport"."Number"
3 order by enroll."Enrollee"."PassportNumber";
```

	ID integer	Name character varying (100)	Number character (10)	age interval
1	3	Putin Vladimir	1337322333	5 years 1 m...
2	1	Zubenko Michael Petrovich	3222444444	5 years 18 ...
3	2	Zmishenko Valery	3222554444	5 years 18 ...
4	4	Banketniy Vladimir	3277654444	5 years 18 ...
5	5	Udmurt Vladimir	3277658224	5 years 18 ...

4. Использование строковых функций – от 2 баллов;

Вывод имени абитуриентов, с удаленными пробелами

```
1 select replace(enroll."Enrollee"."Name", ' ', '') as Name from enroll."Enrollee" ;
```

	name text
1	ZubenkoMichaelPetrovich
2	ZmishenkoValery
3	BanketniyVladimir
4	UdmurtVladimir
5	PutinVladimir

5. Запрос с использованием подзапросов – от 2 баллов (многострочный подзапрос - от 2 баллов);

```
1 select enroll."Faculty"."Email", enroll."Faculty"."FacultyName" from enroll."Faculty"
2 where enroll."Faculty"."FacultyName" IN
3 (select enroll."Programm"."FacultyName" from enroll."Programm" inner join
4 (select count(*), enroll."ProgramToEnrollee"."Programm" from enroll."ProgramToEnrollee" group by enroll."ProgramToEnrollee"."Programm") as "svd"
5 on enroll."Programm"."ProgrammName" = "svd"."Programm");
```

Вывод факультетов, на чьи программы были поданы заявления

	Email character varying (40)	FacultyName [PK] character varying (40)
1	math@university.edu	Faculty of Math
2	history@university.edu	Faculty of history
3	beerpong@university.edu	Faculty of beerpong

6. Вычисление групповой (агрегатной) функции – от 1 балла (с несколькими таблицами – от 2 баллов);

Вывод имени абитуриента и его среднего балла по экзаменам ЕГЭ

```
1 select enroll."Enrollee"."Name", AVG(enroll."DisciplineResult"."Result") from enroll."Enrollee"
2 inner join enroll."CertificateUSE"
3 on enroll."Enrollee"."CertificateUSE" = enroll."CertificateUSE"."ID"
4 inner join enroll."DisciplineResult"
5 on enroll."DisciplineResult"."CertificateUSEID" = enroll."CertificateUSE"."ID"
6 group by enroll."Enrollee"."Name"
```

	Name character varying (100)	avg numeric
1	Banketnyi Vladimir	4.0000000000000000
2	Zmishenko Valery	5.0000000000000000
3	Udmurt Vladimir	3.0000000000000000
4	Putin Vladimir	4.0000000000000000
5	Zubenko Michael Petrovich	5.0000000000000000

7. Вычисление групповой (агрегатной) функции с условием HAVING – от 2 баллов;

Вывод имени и среднего балла абитуриента по экзаменам ЕГЭ, если средний балл больше 3.5

```

1 select enroll."Enrollee"."Name", AVG(enroll."DisciplineResult"."Result") as "res" from enroll."Enrollee"
2 inner join enroll."CertificateUSE"
3 on enroll."Enrollee"."CertificateUSE" = enroll."CertificateUSE"."ID"
4 inner join enroll."DisciplineResult"
5 on enroll."DisciplineResult"."CertificateUSEID" = enroll."CertificateUSE"."ID"
6 group by enroll."Enrollee"."Name" having AVG(enroll."DisciplineResult"."Result") > '3.5'

```

	Name character varying (100)	res numeric
1	Banketnyi Vladimir	4.0000000000000000
2	Zmishenko Valery	5.0000000000000000
3	Putin Vladimir	4.0000000000000000
4	Zubenko Michael Petrovich	5.0000000000000000

8. Использование предикатов EXISTS, ALL, SOME и ANY - от 2 баллов;

Вывод факультетов, на чьи программы были поданы заявления

```

1 select enroll."Faculty"."Email", enroll."Faculty"."FacultyName" from enroll."Faculty"
2 where enroll."Faculty"."FacultyName" = ANY
3 (select enroll."Programm"."FacultyName" from enroll."Programm" inner join |
4 (select count(*), enroll."ProgramToEnrollee"."Programm" from enroll."ProgramToEnrollee" group by enroll."ProgramToEnrollee"."Programm") as "svd"
5 on enroll."Programm"."ProgrammName" = "svd"."Programm")

```

	Email character varying (40)	FacultyName [PK] character varying (40)
1	math@university.edu	Faculty of Math
2	history@university.edu	Faculty of history
3	beerpong@university.edu	Faculty of beerpong

9. Использование запросов с операциями реляционной алгебры (объединение, пересечение и т.д.) - от 2 баллов;

Вывод смежной таблицы факультетов и программ, включая факультеты, для которых пока нет программ

```

1 select * from enroll."Faculty"
2 left join enroll."Programm"
3 on enroll."Faculty"."FacultyName" = enroll."Programm"."FacultyName"

```

	Email character varying (40)	FacultyName character varying (40)	ProgrammName character varying (300)	FacultyName character varying (300)	Capacity integer
1	zmih@university.edu	Faculty of Zmihovedenie	[null]	[null]	[null]
2	math@university.edu	Faculty of Math	IT	Faculty of Math	30
3	math@university.edu	Faculty of Math	Math	Faculty of Math	30
4	tranzmihaciya@university.edu	Faculty of tranzmihaciya	[null]	[null]	[null]
5	history@university.edu	Faculty of history	Modern history	Faculty of history	30
6	history@university.edu	Faculty of history	Ancient History	Faculty of history	30
7	beerpong@university.edu	Faculty of beerpong	Beerpong Technologies	Faculty of beerpong	20

10. Использование объединений запросов (inner join и т.д.) - от 2 баллов.

Вывод имени абитуриента и всех данных о его выпуске из школы.

```

1 select enroll."Enrollee"."Name", enroll."School".* from enroll."Enrollee"
2 inner join enroll."School"
3 on enroll."Enrollee"."School" = enroll."School"."SchoolName"

```

	Name character varying (100)	SchoolName character varying (40)	GraduationDate date	City character varying (60)
1	Zubenko Michael Petrovich	School #1	2019-06-25	Rostov-on-Don
2	Zmishenko Valery	School #282	2019-06-21	Saint-Petersburg
3	Banketnyi Vladimir	School of Ahmat Kadyrov	2019-05-25	Grozny
4	Udmurt Vladimir	School of Ahmat Kadyrov	2019-05-25	Grozny
5	Putin Vladimir	School of Ahmat Kadyrov	2019-05-25	Grozny

Вывод: в ходе выполнения лабораторной работы были получены навыки формирования SQL запросов к базе данных в среде pgAdmin 4