

Министерство цифрового развития, связи и массовых коммуникаций Российской  
Федерации

Сибирский Государственный Университет Телекоммуникаций и Информатики  
СибГУТИ

Кафедра прикладной математики и кибернетики

Расчетно-графическая работа

Вариант №11: Собачьи бега UK

Выполнил: студент 2 курса группы ИП-011

Каргин Роман Александрович

Проверил: Милешко Антон Владимирович

Новосибирск, 2022

Оглавление	
Задание.....	3
Работа.....	4
Этап 1.....	4
Этап 2.....	6
Этап 3.....	9
Этап 4.....	12

## Задание

Создать ПО для отображения и обработки статистических данных для определённого вида спорта. ПО должно включать 2 основных окна: окно, отображающее таблицы БД со статистической информацией и результаты запросов к БД, переключение таблиц и результатов должно быть реализовано через вкладки; и окно для менеджера запросов к БД. Первое окно должно давать возможность просматривать и изменять все таблицы БД, а также просматривать результаты запросов к БД. Должна иметься возможность удалить вкладки с результатами запросов, но не вкладки с таблицами. Также должна иметься возможность перейти к окну менеджера запросов. Окно менеджера запросов должно предоставлять интерфейс для создания, сохранения, удаления, редактирования запросов. Созданные запросы должны отображаться в виде списка с названиями запросов, в который можно добавлять новые запросы, удалять, просматривать существующие. Для создания и редактирования запросов должен предоставляться визуальный интерфейс, а не язык запросов. Редактор запросов должен поддерживать операции выборки, соединения, группирования, подзапросы (в качестве подзапроса используются ранее сохранённые запросы).

Ход работы:

1. Исследование предметной области и создание ER диаграммы.
2. Перевод ER диаграммы в реляционную модель, создание и заполнение БД.
3. Проработка визуального интерфейса приложения
4. Создание диаграммы классов приложения
5. Реализация основного окна приложения
6. Реализация менеджера запросов
7. Тестирование и отладка

Вариант:

11. Собачьи бега UK

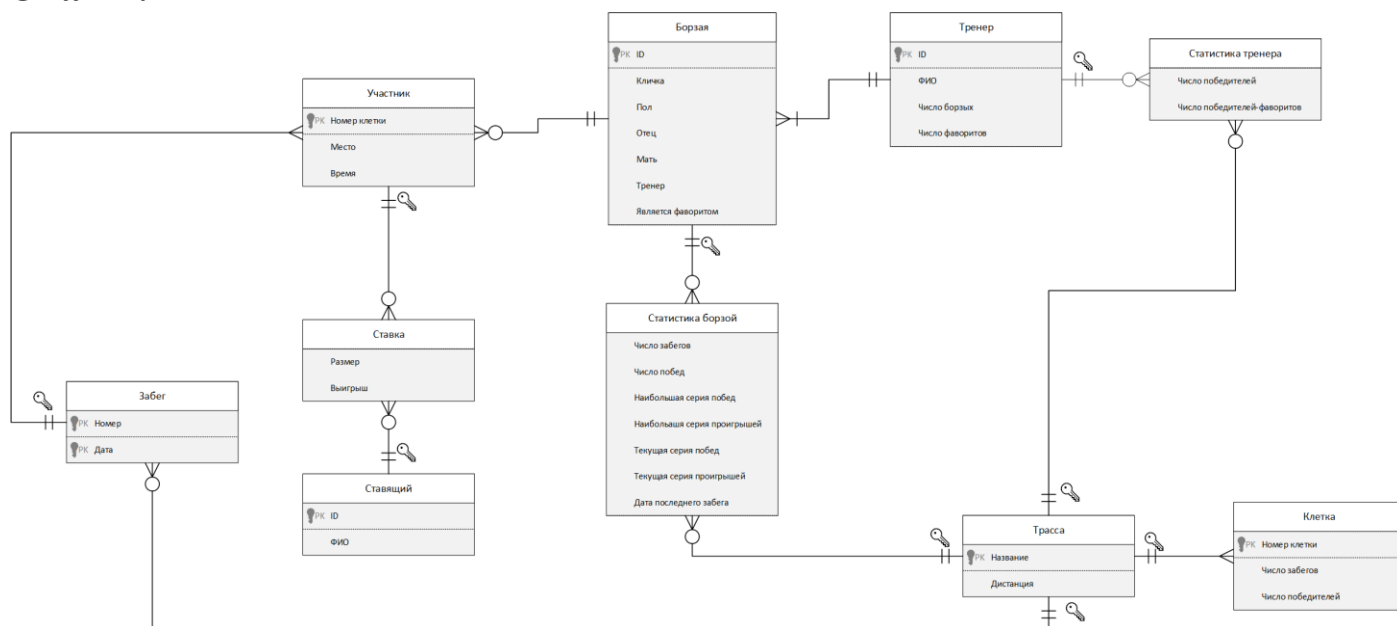
(<https://greyhoundstats.co.uk/>

<https://www.greyhound-data.com/statistics.htm>

<https://www.sportinglife.com/greyhounds/results>)

# Работа

## Этап 1.



Главный элемент предметной области — **борзая**.

Первичный ключ – ID.

У борзой имеется тренер, она может быть участником забегов.

Статистика борзой, а именно число забегов, побед, серии и дата последнего забега борзой зависят не только от ID борзой, но и от названия трассы, где они проходили. Таким образом, для них ID борзой и название трассы являются составным первичным ключом. Само число забегов борзая получает от трассы, которая связана со всеми проходившими на ней забегами.

Для каждой трассы имеется своя статистика, следовательно связь должна быть «0 или более» со стороны борзой к статистике.

Борзая могла ещё не участвовать в забегах, поэтому степень связи, идущей от неё элементу «Участник» равна «0 или более». Связь между борзой и участником существует из-за необходимости хранения ID борзой в элементе «Участник».

У каждой борзой должен быть тренер, и только один, так что степень связи к тренеру будет «только 1».

## Тренер.

Первичный ключ – ID.

Тренер может обладать некоторым числом борзых, которое должно быть строго больше нуля. Поэтому степень связи от тренера к борзой равна «1 или более».

Также у тренера могут быть фавориты, которыми являются те же борзые.

Статистика тренера состоит из числа победителей и победителей-фаворитов, которые зависят от ID тренера и названия трассы. Для них они являются составным первичным ключом.

Борзые тренера могли ещё не участвовать в забегах, или участвовать в забегах на разных трассах, так что степень связи от него к его статистике равна «0 или более», так как статистика тренера разделена по трассам.

### **Трасса.**

Первичный ключ – Название.

Трасса содержит информацию о себе (дистанция) и 6 клеток. Каждая клетка хранит свой номер, число забегов и число победителей на данной трассе.

Ключ **клетки** состоит из названия трассы и номера клетки.

Трасса может иметь несколько клеток (степень связи от неё к клетке – «более 1»), а клетка может принадлежать лишь одной трассе (степень связи – «только 1»), так как её статистика строго зависит от названия трассы.

Трасса может иметь любое количество забегов, а с ним и участников (борзые и тренеры), так что все связи, идущие от неё к остальным элементам, кроме клетки, имеют степень «0 или более».

### **Забег.**

Первичный ключ – название трассы + номер забега + дата забега.

Забег имеет 6 участников, степень связи от забега к участнику равна «более 1».

Так как название трассы является частью ключа всего забега, то степень связи к трассе равна «только 1».

Ключ **участника** состоит из ключа забега и номера клетки, так что степень связи от участника к забегу равна «только 1». Участник имеет место, которым он пришёл на финиш, время. Также участник хранит ID борзой, которой он является, из чего следует, что имеет связь между ним и борзой, степень которого равна «только 1».

Забег может иметь любое количество ставок, как и участник. Степени связей, идущих к ставке, равны «0 или более».

**Ставка** имеет размер и выигрыш. Ключом ставки будет комбинация ключа участника (номер забега и номер клетки) и ID ставящего. Ставка может относиться лишь к одному участнику, забегу и ставящему, следовательно степени связей, идущих от неё к ним, равны «только 1».

## Этап 2.

В ER-модели отсутствуют взаимоисключающие связи, супертипы и подтипы, связи «многие ко многим». Также ER-модель изначально сделана в третьей нормальной форме. Достаточно будет преобразовать все сущности в таблицы, указывая типы данных, отмечая уникальные идентификаторы первичными ключами и связи «к одному» внешними ключами.

Борзая	
ID	<b>Перв. ключ</b> , INTEGER
Кличка	VARCHAR (32)
Пол	CHAR
Отец	VARCHAR (32)
Мать	VARCHAR (32)
Тренер	<b>Вн. ключ</b> , INTEGER
Является фаворитом	BOOL

Статистика борзой	
ID борзой	<b>Перв. и вн. ключ</b> , INTEGER
Название трассы	<b>Перв. и вн. ключ</b> , VARCHAR (20)
Число забегов	UNSIGNED SMALLINT
Число побед	UNSIGNED SMALLINT
Наибольшая серия побед	UNSIGNED SMALLINT
Наибольшая серия проигрышей	UNSIGNED SMALLINT
Текущая серия побед	UNSIGNED SMALLINT
Текущая серия проигрышей	UNSIGNED SMALLINT
Дата последнего забега	DATE

Тренер	
ID	<b>Перв. ключ</b> , INTEGER
ФИО	VARCHAR (64)
Число борзых	UNSIGNED SMALLINT
Число фаворитов	UNSIGNED SMALLINT

Статистика тренера	
ID тренера	<b>Перв. и вн. ключ</b> , INTEGER
Название трассы	<b>Перв. и вн. ключ</b> , VARCHAR (20)
Число победителей	UNSIGNED SMALLINT
Число победителей-фаворитов	UNSIGNED SMALLINT

Трасса	
Название	Перв. ключ, VARCHAR (20)
Дистанция	DOUBLE

Клетка	
Название трассы	Перв. и вн. ключ, VARCHAR (20)
Номер клетки	Перв. ключ, SMALLINT
Число забегов	UNSIGNED INTEGER
Число победителей	UNSIGNED INTEGER

Забег	
Название трассы	Перв. и вн. ключ, VARCHAR (20)
Номер	Перв. ключ, UNSIGNED SMALLINT
Дата	Перв. ключ, DATE

Участник	
Название трассы	Перв. и вн. ключ, VARCHAR (20)
Номер забега	Перв. и вн. ключ, UNSIGNED SMALLINT
Дата забега	Перв. и вн. ключ, DATE
Номер клетки	Перв. ключ, UNSIGNED SMALLINT
ID борзой	Вн. ключ, INTEGER
Место	UNSIGNED SMALLINT
Время	TIME

Ставка	
Название трассы	Перв. и вн. ключ, VARCHAR (20)
Номер забега	Перв. и вн. ключ, UNSIGNED SMALLINT
Дата забега	Перв. и вн. ключ, DATE
Номер клетки	Перв. и вн. ключ, UNSIGNED SMALLINT
ID ставящего	Перв. и вн. ключ, INTEGER
Размер	DOUBLE
Выигрыш	DOUBLE

Ставящий	
ID	Перв. ключ, INTEGER
ФИО	VARCHAR (64)

Для автоматического обновления числа побед и участия для таблиц были прописаны триггеры:

### Триггер участника

```
UPDATE "Dog statistics" SET
"Race count" =
(SELECT COUNT(*)
FROM Participant WHERE "Dog ID" == "Dog statistics"."Dog ID"),

"Win count" =
(SELECT COUNT(*)
FROM Participant WHERE "Dog ID" == "Dog statistics"."Dog ID"
and Participant.Place==1),

"Last race date" =
(SELECT MAX("Race date")
FROM Participant WHERE "Dog ID" == "Dog statistics"."Dog ID");

UPDATE Trap SET
"Race count" =
(SELECT COUNT(*)
FROM Participant WHERE "Track name"=Trap."Track name"
and "Trap number"=Trap."Trap number"),

"Winner count" =
(SELECT COUNT(*)
FROM Participant WHERE "Track name"=Trap."Track name"
and "Trap number"=Trap."Trap number"
and Place=1);
```

### Триггер собаки

```
UPDATE Trainer
SET "Dog count" = (SELECT COUNT(*) FROM Dog WHERE Trainer=Trainer.ID),
"Favorite count" = (SELECT COUNT(*) FROM Dog WHERE Trainer=Trainer.ID and "Is favorite"=1);
```

### Триггер статистики собаки

```
UPDATE "Trainer statistics" SET
"Winner count" =
(SELECT COUNT(*) FROM
(SELECT Dog.ID, Dog."Is favorite", "Dog statistics"."Win count"
from Dog INNER JOIN "Dog statistics" on Dog.ID="Dog statistics"."Dog ID"
and "Trainer statistics"."Track name"="Dog statistics"."Track name"
and Dog.Trainer="Trainer statistics"."Trainer ID")
WHERE "Win count">0),
"Winner-favorite count" =
(SELECT COUNT(*) FROM
(SELECT Dog.ID, Dog."Is favorite", "Dog statistics"."Win count"
from Dog INNER JOIN "Dog statistics" on Dog.ID="Dog statistics"."Dog ID"
and "Trainer statistics"."Track name"="Dog statistics"."Track name"
and Dog.Trainer="Trainer statistics"."Trainer ID")
WHERE "Win count">0 and "Is favorite"=1);
```



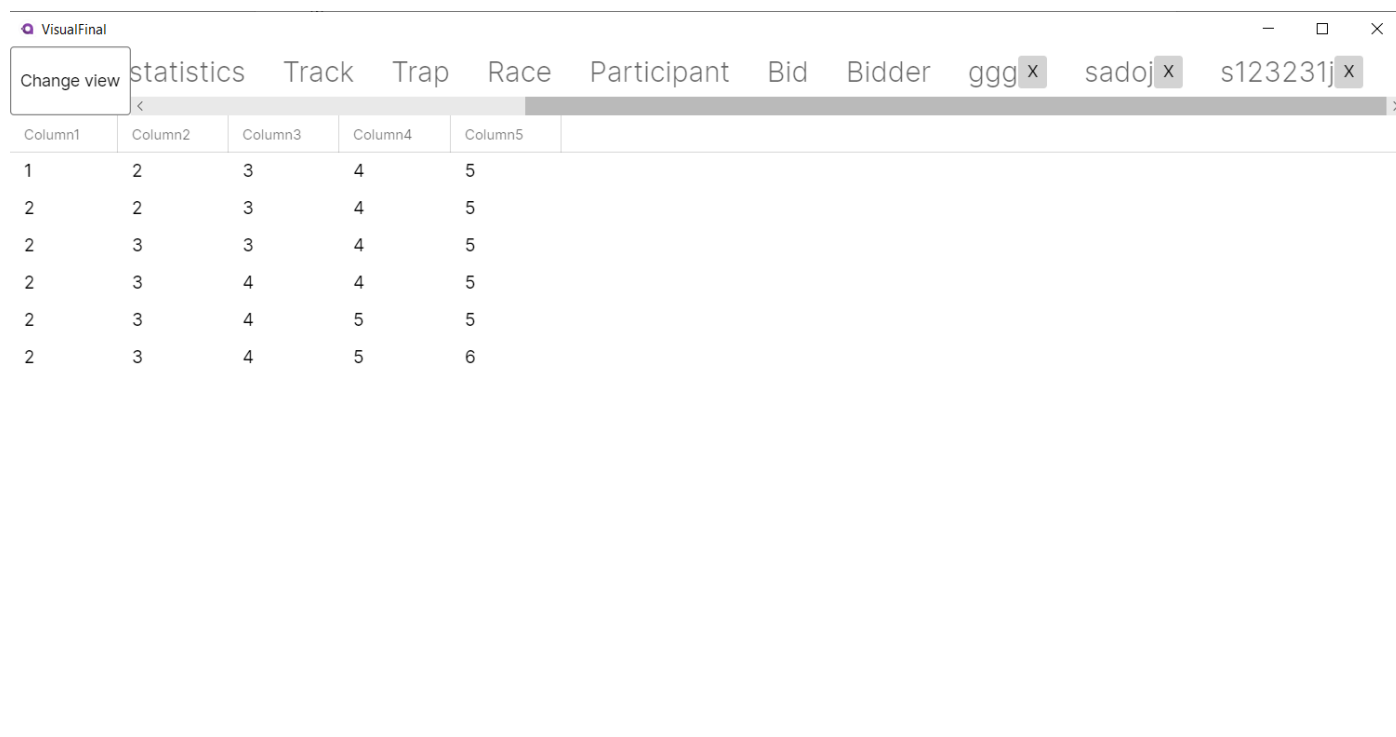
### Этап 3.

Интерфейс состоит из двух видов и трёх дополнительных окон.



Первый вид в верхней строке содержит все доступные базы данных и результаты запросов в виде вкладок. При переключении между ними меняется показываемая в центре окна база данных на соответствующую названию вкладки.

Также в левом верхнем углу находится кнопка для смены на вид редактирования запросов.



У вкладок, содержащих таблицы, отсутствует кнопка для закрытия; у вкладок, содержащих результаты запросов, присутствует.

VisualFinal				
1234	Show	Delete	Your query here	
12534	Show	Delete		
123	Show	Delete		
126734	Show	Delete		
			Select	
			Union	
			Group	
			Delete Last Change	
			Clear	
Create new				
Change view				

Второй вид состоит из списка запросов слева и вывода текста запроса справа. Имеется возможность выбрать и редактировать/удалить нужный, создать новый запрос. При редактировании запроса доступна возможность выборки, соединения и группирования таблиц/результатов.

SELECT			
SELECT			
FROM			
Dog			
Dog statistics			
Trainer			
Trainer statistics			
Track			
Trap			
Race			
Participant			
Bid			
Bidder			
WHERE			
Cancel			
Confirm			

При нажатии “SELECT” открывается окно операции выборки.

JOIN

INNER

LEFT OUTER

RIGHT OUTER

FULL OUTER

First Table

Dog

Dog statistics

Trainer

Trainer statistics

Second Table

Dog

Dog statistics

Trainer

Trainer statistics

ON

Cancel

Confirm

При нажатии “JOIN” открывается окно операции соединения.

GROUP

Table to group

Dog

Dog statistics

Trainer

Trainer statistics

Track

Trap

Race

Participant

Bid

Bidder

GROUP BY

Cancel

Confirm

При нажатии “GROUP” открывается окно операции группировки.

## Этап 4.

Диаграмма классов в отдельном файле “ClassDiagramm.pdf”.

Одними из важнейших классов являются сущности базы данных (Dog, Track, Trainer и т.д.), которые содержат друг друга для навигации между взаимосвязанными таблицами из базы данных.

Дальше идёт контекст базы данных, содержащий все предыдущие сущности и заполняющий их списки строками из файла базы данных. Класс связывает приложение с базой данных извне, отсюда «boundary» тип.

После рассмотрим ViewModels. Они являются классами типа «control», поскольку хранят строки для привязки к текст блокам окон и настраивают кнопки окон.

Сами классы типа View взаимодействуют с внешним миром, настраивая эвенты.

Также есть такие классы, как MyTab, StaticTab, DynamicTab, Query. MyTab и его наследники являются вкладками первого окна, содержа в себе каждый таблицу базы данных, с которой он связан. DynamicTab и Query связаны друг с другом, один уничтожается при уничтожении другого. Также DynamicTab зависит от Query и меняется вместе с ним, поскольку при обновлении запроса может обновиться и его результирующая таблица.

Есть ещё и классы ViewLocator, App, Program, которые являются автоматически сгенерированными.