

Министерство цифрового развития
Федеральное государственное бюджетное образовательное учреждение высшего
образования
Сибирский Государственный Университет Телекоммуникаций и Информатики

Кафедра прикладной математики и кибернетики

Расчетно-графическая работа.

Вариант 21.

Выполнил: студент 2 курса группы ИП-017

Хьюго Михаил Алексеевич

Преподаватель:

Милешко Антон Владимирович

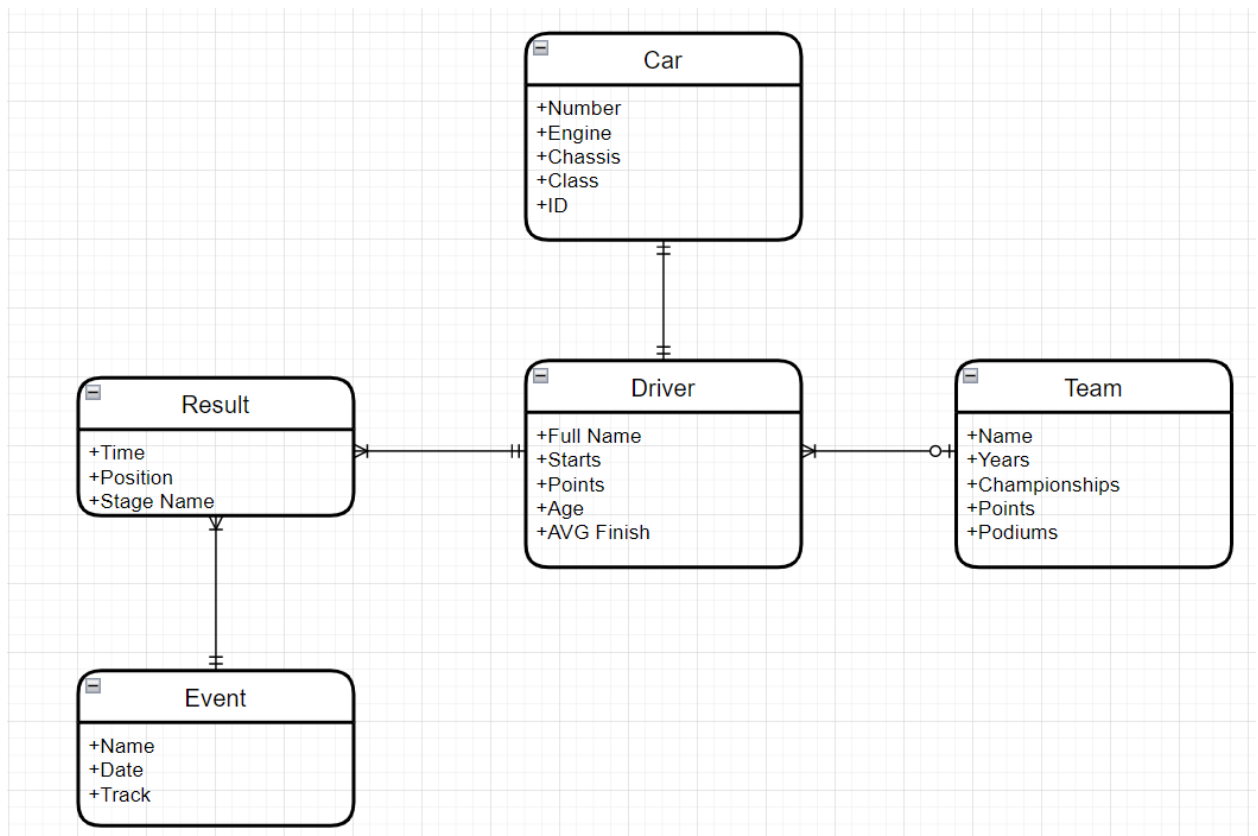
Новосибирск, 2022

Оглавление

1. Исследование предметной области и создание ER диаграммы.....	3
2. Перевод ER диаграммы в реляционную модель, создание и заполнение БД	4
3. Проработка визуального интерфейса.	5
3.1. Окно с таблицами БД.	5
3.2. Окно менеджера запросов.	6
4. Диаграмма классов.....	6

1. Исследование предметной области и создание ER диаграммы

Чемпионат ралли состоит из нескольких событий в разных странах, каждое событие состоит из нескольких стадий. В них участвуют пилоты, от определенных команд на своих машинах, результат заезда пилота на конкретной стадии фиксируется.



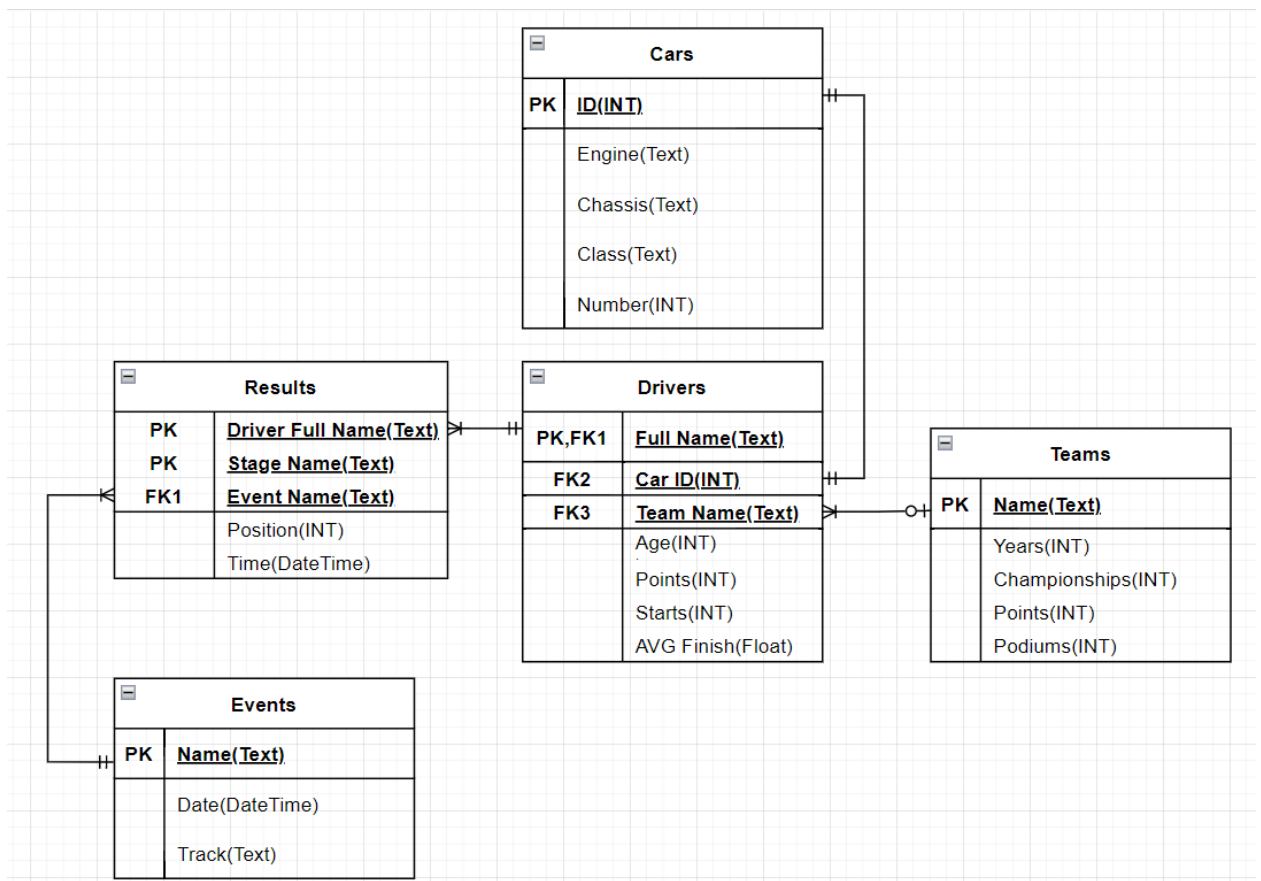
После изучения источников информации была составлена ER диаграмма, в которой отображены 5 сущностей:

- Driver
- Team
- Car
- Result
- Event

В этой диаграмме отображены следующие связи:

- Driver – Team, многие к одному(опционально), пилот не находится вообще или находится только в одной команде, в команде может быть несколько пилотов.
- Driver – Car, один к одному, пилот управляет только одной машиной, машина принадлежит только одному пилоту.
- Driver – Result, один ко многим, у пилота может быть несколько результатов по разным заездам, результат принадлежит только одному пилоту.
- Result – Event, многие к одному, у события турнира может быть несколько результатов, результат относится только к одному событию.

2. Перевод ER диаграммы в реляционную модель, создание и заполнение БД



1. Drivers:

- Full Name – Первичный ключ и внешний ключ в таблицу Results.
- Car ID – внешний ключ в таблицу Cars.
- Team Name – внешний ключ в таблицу Teams.

2. Teams:

- Name – первичный ключ.

3. Cars:

- ID – первичный ключ.

4. Results:

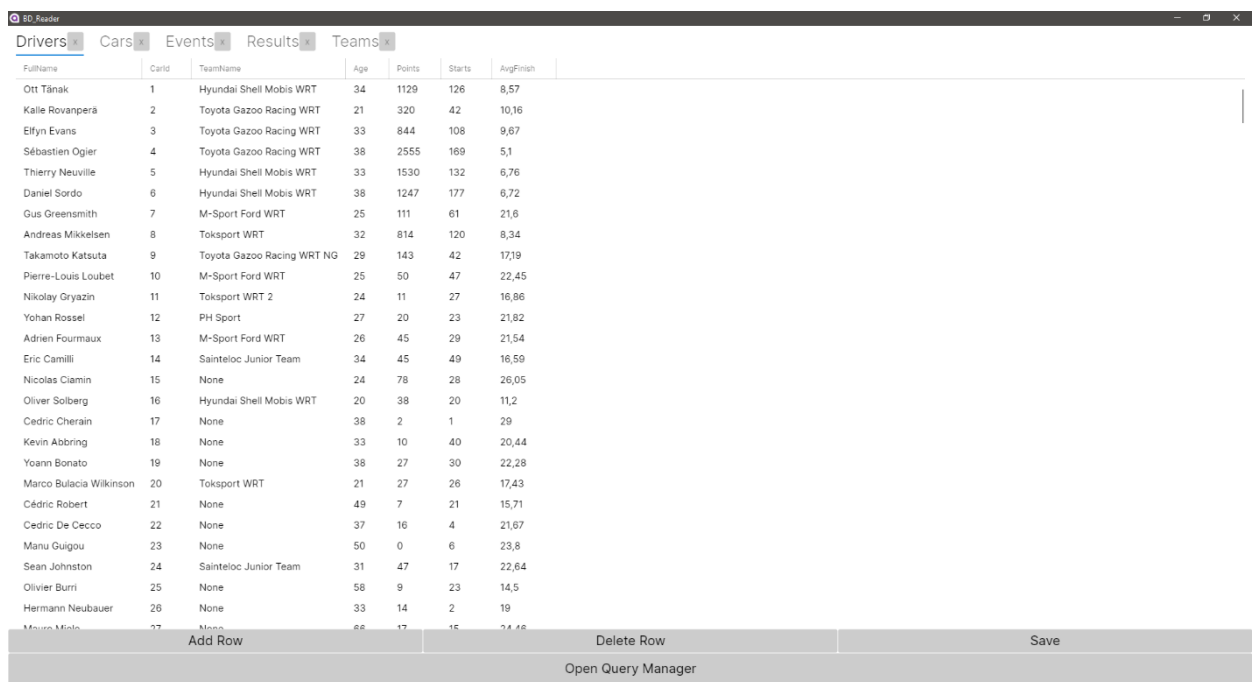
- Driver Full Name + Stage Name – составной первичный ключ.
- Event Name – внешний ключ в таблицу Events.

5. Events:

- Name – первичный ключ.

3. Проработка визуального интерфейса.

3.1. Окно с таблицами БД.



FullName	CarId	TeamName	Age	Points	Starts	AvgFinish
Ott Tänak	1	Hyundai Shell Mobis WRT	34	1129	126	8,57
Kalle Rovanperä	2	Toyota Gazoo Racing WRT	21	320	42	10,16
Efyn Evans	3	Toyota Gazoo Racing WRT	33	844	108	9,67
Sébastien Ogier	4	Toyota Gazoo Racing WRT	38	2555	169	5,1
Thierry Neuville	5	Hyundai Shell Mobis WRT	33	1530	132	6,76
Daniel Sordo	6	Hyundai Shell Mobis WRT	38	1247	177	6,72
Gus Greensmith	7	M-Sport Ford WRT	25	111	61	21,6
Andreas Mikkelsen	8	Toksport WRT	32	814	120	8,34
Takamoto Katsuta	9	Toyota Gazoo Racing WRT NG	29	143	42	17,19
Pierre-Louis Loubet	10	M-Sport Ford WRT	25	50	47	22,45
Nikolay Gryazin	11	Toksport WRT 2	24	11	27	16,86
Yohan Rossel	12	PH Sport	27	20	23	21,82
Adrien Fourmaux	13	M-Sport Ford WRT	26	45	29	21,54
Eric Camilli	14	Sainteloc Junior Team	34	45	49	16,59
Nicolas Ciamin	15	None	24	78	28	26,05
Oliver Solberg	16	Hyundai Shell Mobis WRT	20	38	20	11,2
Cedric Cherain	17	None	38	2	1	29
Kevin Abbring	18	None	33	10	40	20,44
Yoann Bonato	19	None	38	27	30	22,28
Marco Bulacia Wilkinson	20	Toksport WRT	21	27	26	17,43
Cédric Robert	21	None	49	7	21	15,71
Cedric De Cecco	22	None	37	16	4	21,67
Manu Guigou	23	None	50	0	6	23,8
Sean Johnston	24	Sainteloc Junior Team	31	47	17	22,64
Olivier Burri	25	None	58	9	23	14,5
Hermann Neubauer	26	None	33	14	2	19
Mauro Milla	27	None	68	17	15	24,16

Add Row

Delete Row

Save

Open Query Manager

3.2. Окно менеджера запросов.

AND/OR	Column	Condition	Value
	CarId	>	5
AND	CarId	<	10

AND/OR	Column	Condition	Value
	TeamName	Contains	oyo
OR	Chassis	Not Contains	mi

Окно с таблицами БД реализовано через TabControl'ы и DataGrid'ы.

Окно менеджера запросов реализовано с использованием Expander'ов, в которых скрыты элементы создания запроса: ListBox'ы, DataGrid'ы, различные кнопки, поля для ввода и вывода текста и ComboBox'ы.

4. Диаграмма классов

Классы Car, Driver, Team, Event и Result представляют собой записи в соответствующих таблицах БД, содержат в себе данные, метод, возвращающий ключ, и индексатор для получения значения поля по его имени. А класс WRCContext представляет саму базу данных со всеми записями и связями.

Классы CarsTableViewModel, DriversTableViewModel, TeamsTableViewModel, EventsTableViewModel и ResultsTableViewModel являются контекстом для окон с таблицами для вывода данных соответствующей таблицы из БД. Каждый хранит в себе коллекцию записей.

QueryTableViewModel аналогичен, только используется для вывода таблиц запросов.

MainWindowViewModel – контекст главного окна приложения, используется для переключения окна просмотра таблиц и окна менеджера запросов.

DBViewerViewModel является контекстом окна просмотра таблиц, здесь осуществляется загрузка всех списков с записями, добавление, удаление и сохранение таблиц.

QueryManagerViewModel – контекст для окна менеджера запросов, здесь происходит обработка действий пользователя в этом окне и вызов выполнения самих запросов.

ViewModelBase - родительский класс для контекстов всех таблиц, позволяет переопределить метод получения записей таблицы.

Filter класс хранит в себе информацию об одном условии для фильтрации.

Table класс хранит информацию о таблице: её имя, список полей и записей, контекст таблицы в виде ViewModelBase, индикатор о том, таблица запросная или из БД и т.п.

Класс Handler – является абстрактным классом для реализации шаблона проектирования «Цепочка обязанностей», которая последовательно выполняет обработку. Содержит в себе контекст менеджера запросов, объект Handler – следующий обработчик и сам обрабатывающий метод.

FilterHandler – звено цепочки обязанностей выполняющее фильтрацию запроса.

GroupHandler - звено цепочки обязанностей выполняющее группировку запроса.

Классы CarsTableView, DriversTableView, TeamsTableView, EventsTableView и ResultsTableView помимо инициализации своих окон обрабатывают событие выбора строк в таблицах БД для их удаления.

DBViewerView обрабатывает события удаления таба, а также их переключение.

QueryManagerView обрабатывает события выборки таблиц, колонок, добавления и редактирования фильтров, выбора группировки, кнопки отправки запроса, возврата к окну просмотра таблиц и вызывает при необходимости соответствующие функции своего контекста.