

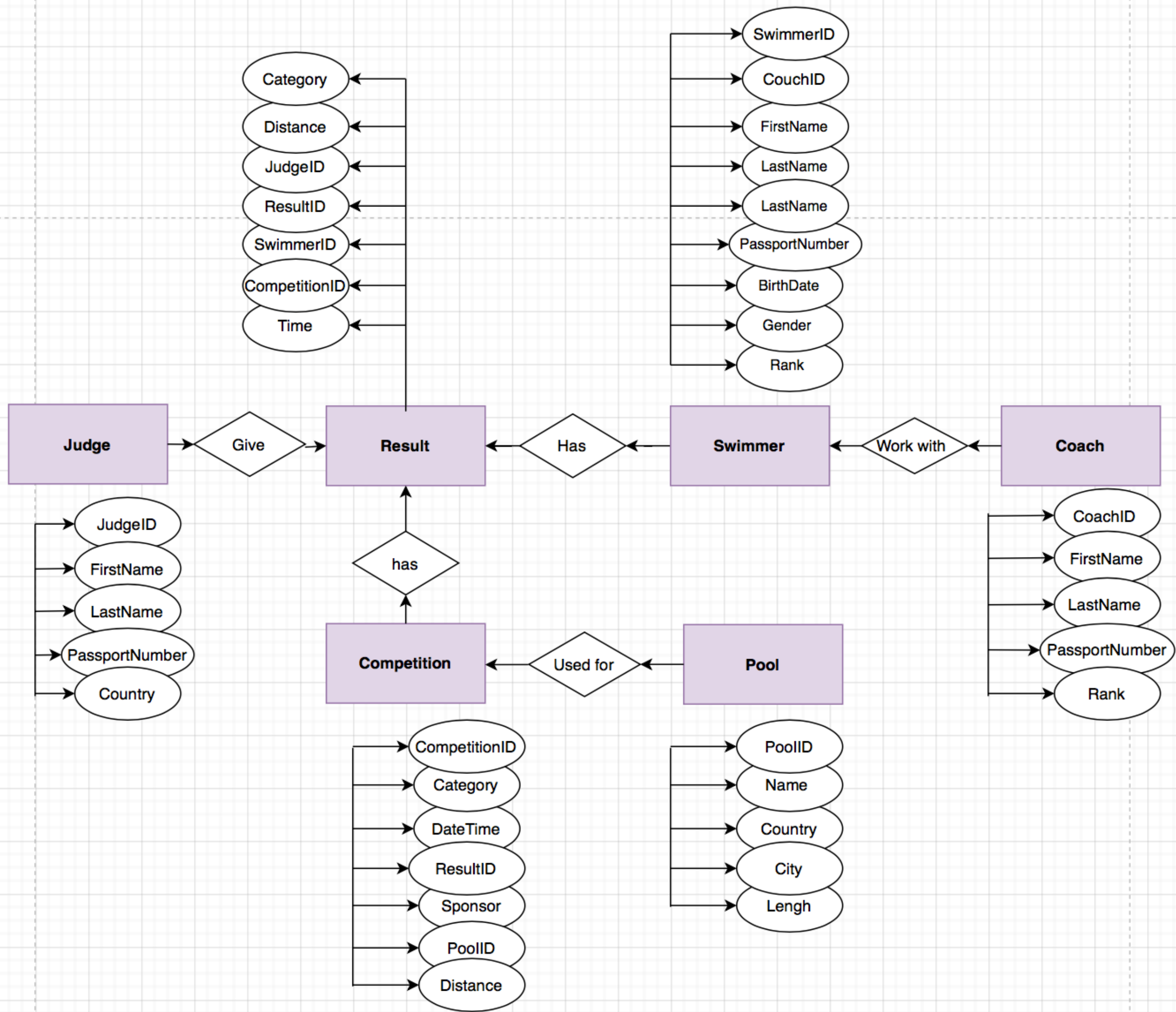
Содержание проекта

- Концептуальное проектирование базы данных (БД)
- Логическое проектирование БД
- Физическое проектирование БД
- Заполнение таблиц тестовыми данными
- Тестирование БД
- Создание объектов БД
- Создание пакетов ETL
- Визуализация данных в Power BI
- Визуализация данных в Tableau

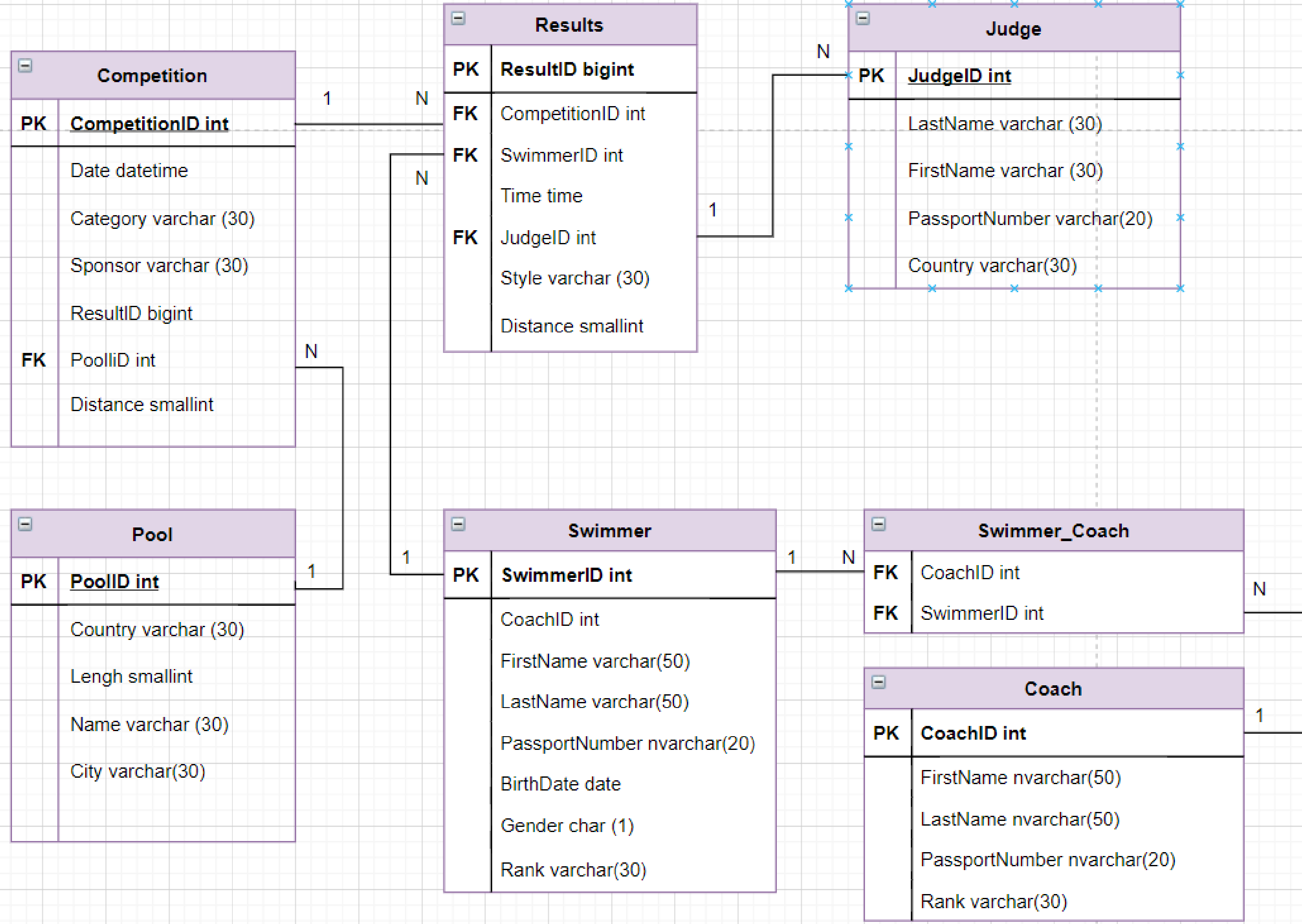
Концептуальная модель базы данных Swimming Competition

ER-диаграмма
«Сущность-связь»

Использовалось
веб-приложение draw.io



Логическая
модель базы
данных
**Swimming
Competition**



Использовалось
веб-приложение
draw.io

Физическая модель базы данных Swimming Competition

Создание таблиц

- Coach
- Pool
- Judge

MS SQL Server
Management Studio

```
CREATE DATABASE SwimmingCompetition3
GO

USE SwimmingCompetition3

CREATE TABLE dbo.Coach (
    CoachID int PRIMARY KEY IDENTITY (1,1),
    FirstName varchar(50),
    LastName varchar(50),
    PassportNumber varchar(60) NOT NULL,
    [Rank] varchar(3)
)
GO

CREATE TABLE dbo.[Pool] (
    PoolID int PRIMARY KEY IDENTITY (1,1),
    City varchar(60),
    Country varchar(60),
    [Name] varchar(60) NOT NULL,
    [Length] smallint NOT NULL CHECK ([Length] > 0
)
GO

CREATE TABLE dbo.Judge (
    JudgeID int PRIMARY KEY IDENTITY (1,1),
    FirstName varchar (30),
    LastName varchar (30),
    Country varchar (60) NOT NULL,
    PassportNumber varchar(60) NOT NULL
)
GO
```

Создание таблиц

- Swimmer
- Competition
- Results
- Swimmer_Coach

MS SQL Server
Management Studio

```
CREATE TABLE dbo.Swimmer (  
    SwimmerID int PRIMARY KEY IDENTITY(1,1),  
    FirstName varchar(50),  
    LastName varchar (50),  
    PassportNumber nvarchar(20) NOT NULL,  
    BirtDate date NOT NULL,  
    Gender Varchar(3) NOT NULL CHECK (Gender = 'M' OR Gender = 'F'),  
    [Rank] varchar(3),  
    CoachID int  
)  
GO  
  
CREATE TABLE dbo.Competition (  
    CompetitionID int PRIMARY KEY IDENTITY(1,1),  
    [Date] datetime NOT NULL,  
    Category varchar(60) NOT NULL,  
    Sponsor varchar(60),  
    PoolID int FOREIGN KEY (PoolID) REFERENCES [Pool] (PoolID),  
    Distance smallint NOT NULL CHECK (Distance > 0)  
)  
GO  
  
CREATE TABLE dbo.Results (  
    ResultID bigint PRIMARY KEY IDENTITY (1,1),  
    CompetitionID int FOREIGN KEY (CompetitionID) REFERENCES Competition (CompetitionID),  
    SwimmerID int FOREIGN KEY (SwimmerID) REFERENCES Swimmer (SwimmerID),  
    [Time] time NOT NULL,  
    JudgeID int FOREIGN KEY (JudgeID) REFERENCES Judge (JudgeID),  
    Style varchar(30) NOT NULL,  
    Distance smallint NOT NULL CHECK (Distance > 0)  
)  
  
CREATE TABLE Swimmer_Coach(  
    SwimmerID int FOREIGN KEY (SwimmerID) REFERENCES Swimmer (SwimmerID),  
    CoachID int FOREIGN KEY (CoachID) REFERENCES Coach (CoachID)  
)
```

Заполнение таблиц
тестовыми данными

Таблица Pool

```
-- ТАБЛИЦА POOL

-- WITH CTE_City
AS
(
SELECT DISTINCT City
FROM [AdventureWorks2017].Person.Address
),
CTE_Country
AS
(
SELECT DISTINCT Name
FROM [AdventureWorks2017].Person.CountryRegion
),
CTE_Pool
AS
(
SELECT TOP 10000 City, Name AS Country,
CONCAT( LEFT(City, 1 ), RIGHT(Name, 3),
SUBSTRING ('qwertyuiopasdfghjklzxcvbnm', (abs(checksum(newid())) % 26)+1,1)) AS Name,
CASE ABS(CHECKSUM(NEWID()))%2
WHEN 0 THEN 25
ELSE 50
END AS Length
FROM CTE_City
CROSS JOIN CTE_Country
ORDER BY NEWID()
)
INSERT INTO [SwimmingCompetition3].[dbo].[Pool] ([City] , [Country] , [Name] , [Length])
SELECT *
FROM CTE_Pool
```

Заполнение таблиц
тестовыми данными

Таблица Coach

```
-- ТАБЛИЦА COACH
```

```
--
;WITH CTE_FirstName AS
(
SELECT DISTINCT [FirstName]
FROM [AdventureWorks2017].[Person].[Person]
),
CTE_LastName AS
(
SELECT DISTINCT [LastName]
FROM [AdventureWorks2017].[Person].[Person]
),
CTE_Coach AS
(
SELECT TOP 10000 *
, CONCAT ( LEFT(FirstName, 1) , LEFT(LastName, 1) ,
SUBSTRING( CAST(ABS(CHECKSUM(NEWID())) AS varchar(20)) , 1 , 6)) AS PassportNumber
, CASE ABS(CHECKSUM(NEWID())) % 2
WHEN 0 THEN 'MS'
ELSE 'WMS'
END AS Rank
FROM CTE_FirstName
CROSS JOIN CTE_LastName
ORDER BY NEWID() )
insert into [SwimmingCompetition3].[dbo].[Coach]([FirstName],[LastName],[PassportNumber],[Rank])
select *
FROM CTE_Coach;
```

Заполнение таблиц
тестовыми данными

Таблица
Judge

```
-- ТАБЛИЦА JUDGE
-----

;WITH CTE_FirstName
AS
(
SELECT DISTINCT FirstName
FROM [AdventureWorks2017].Person.Person
),
CTE_LastName
AS
(
SELECT DISTINCT LastName
FROM [AdventureWorks2017].Person.Person
),
CTE_Judge
AS
(
SELECT TOP 10000 *
, CONCAT(LEFT(LastName,3),LOWER(REVERSE(FirstName))) AS Country
, CONCAT(SUBSTRING(FirstName,1,1)
,SUBSTRING ('QWERTYUIOPASDFGHJKLZXCVBNM',(abs(checksum(newid())) % 26)+1,1)
, ABS(CHECKSUM(NEWID())))) AS PassportNumber
FROM CTE_FirstName
CROSS JOIN CTE_LastName
ORDER BY NEWID()
)
INSERT INTO [SwimmingCompetition3].[dbo].[Judge] ([FirstName] , [LastName] , [Country] , [PassportNumber])
SELECT *
FROM CTE_Judge
```


Заполнение таблиц
тестовыми данными

Таблица
Swimmer

```
--- таблица SWIMMER

;WITH CTE_FirstName AS
(
    SELECT DISTINCT [FirstName]
    FROM [AdventureWorks2017].[Person].[Person]
),
CTE_LastName AS
(
    SELECT DISTINCT [LastName]
    FROM [AdventureWorks2017].[Person].[Person]
),
CTE_Swimmer AS
(
    SELECT TOP 100000 *
        , CONCAT ( LEFT(FirstName, 1) , LEFT(LastName, 1) , SUBSTRING( CAST(ABS(CHECKSUM(NEWID())) AS varchar(20)) , 1 , 6)) AS PassportNumber
        , CAST (DATEADD(second , -ABS(CHECKSUM(NEWID())) , GETDATE()) AS Date) AS BirtDate
        , CASE ABS(CHECKSUM(NEWID())) % 2
            WHEN 0 THEN 'W'
            ELSE 'M'
        END AS Gender
        , CASE ABS(CHECKSUM(NEWID())) % 8
            WHEN 0 THEN 'U1'
            WHEN 1 THEN 'U2'
            WHEN 2 THEN 'U3'
            WHEN 3 THEN '1'
            WHEN 4 THEN '2'
            WHEN 5 THEN '3'
            WHEN 6 THEN 'CMS'
            ELSE 'MS'
        END AS Rank
        , 1 AS CoachID
    FROM CTE_FirstName
    CROSS JOIN CTE_LastName
    ORDER BY NEWID() )
INSERT INTO [SwimmingCompetition3].[dbo].[Swimmer] ([FirstName] , [LastName] , [PassportNumber] , [BirtDate] , [Gender] , [Rank] , [CoachID])
SELECT *
FROM CTE_Swimmer
```

Заполнение таблиц
тестовыми данными

Продолжение таблицы
Swimmer

```
UPDATE [dbo].[Swimmer]
SET [CoachID] = 1169
WHERE [SwimmerID] < 90000
GO
UPDATE [dbo].[Swimmer]
SET [CoachID] = 11
WHERE [SwimmerID] < 80000
GO
UPDATE [dbo].[Swimmer]
SET [CoachID] = 8625
WHERE [SwimmerID] < 70000
GO
UPDATE [dbo].[Swimmer]
SET [CoachID] = 1895
WHERE [SwimmerID] < 60000
GO
UPDATE [dbo].[Swimmer]
SET [CoachID] = 8695
WHERE [SwimmerID] < 50000
GO
UPDATE [dbo].[Swimmer]
SET [CoachID] = 1313
WHERE [SwimmerID] < 40000
GO
UPDATE [dbo].[Swimmer]
SET [CoachID] = 28
WHERE [SwimmerID] < 30000
go
UPDATE [dbo].[Swimmer]
SET [CoachID] = 963
WHERE [SwimmerID] < 20000
go
UPDATE [dbo].[Swimmer]
SET [CoachID] = 333
WHERE [SwimmerID] < 10000
```

Заполнение таблиц тестовыми данными

Таблица Results

```
--ТАБЛИЦА RESULTS
-----
declare @Date datetime = '2021-01-01 00:00:01:503'
;WITH CTE_Competition AS
(
SELECT top 100 CompetitionID
FROM dbo.Competition
),
CTE_Swimmer AS
(
SELECT top 10000 SwimmerID
FROM dbo.Swimmer
),
CTE_JudgeID AS
(
SELECT top 1000 JudgeID
FROM dbo.Judge
),
CTE_ResultID AS (
SELECT TOP 1000000 *,
CAST (dateadd(MILLISECOND,rand(checksum(newid()))*999
,dateadd(SECOND,rand(checksum(newid()))*60
,dateadd(minute,rand(checksum(newid()))*15, @Date))) as Time(7)) AS [Time],
CASE ABS(CHECKSUM(NEWID())) % 3
WHEN 0 THEN 'Front Crawl'
WHEN 1 THEN 'Breaststroke'
WHEN 2 THEN 'Butterfly Stroke'
WHEN 3 THEN 'Backstroke'
ELSE 'Sidestroke'
END AS [Style],
CASE ABS(CHECKSUM(NEWID())) % 6
WHEN 0 THEN '50'
WHEN 1 THEN '100'
WHEN 2 THEN '200'
WHEN 3 THEN '400'
ELSE '800'
END AS [Distance]
FROM CTE_Competition
CROSS JOIN CTE_JudgeID
CROSS JOIN CTE_Swimmer
ORDER BY newid())
INSERT INTO [dbo].[Results] ([CompetitionID],[JudgeID], [SwimmerID] , [Time] ,[Style] , [Distance])
SELECT *
FROM CTE_ResultID
```

Заполнение таблиц
тестовыми данными

Таблица Competition

```
--ТАБЛИЦА COMPETITION
--
--
;WITH CTE_Sponsor AS
(
SELECT DISTINCT [Name]
FROM [AdventureWorks2017].[Purchasing].[Vendor]
),
CTE_Pool AS
(
SELECT [PoolID]
FROM [dbo].[Pool]
),
CTE_Competition AS
(
SELECT TOP 10000 *
, CAST (DATEADD(second , -ABS(CHECKSUM(NEWID()))) , GETDATE()) AS Date) AS Date
, CASE ABS(CHECKSUM(NEWID())) % 5
    WHEN 0 THEN 'Front Crawl'
    WHEN 1 THEN 'Breaststroke'
    WHEN 2 THEN 'Butterfly Stroke'
    WHEN 3 THEN 'Backstroke'
    ELSE 'Sidestroke'
END AS Category
, CASE ABS(CHECKSUM(NEWID())) % 6
    WHEN 0 THEN '50'
    WHEN 1 THEN '100'
    WHEN 2 THEN '200'
    WHEN 3 THEN '400'
    ELSE '800'
END AS Distance
FROM CTE_Sponsor
CROSS JOIN CTE_Pool
ORDER BY NEWID() )
insert into [SwimmingCompetition3].[dbo].[Competition]([Sponsor],[PoolID],[Date],[Category],[Distance])
select *
FROM CTE_Competition;
```

Заполнение таблиц тестовыми данными

Bridge Таблица
Swimmer_Coach

```
-- ТАБЛИЦА SWIMMER_COACH
```

```
;WITH CTE_Swimmer  
AS
```

```
(  
SELECT SwimmerID  
FROM dbo.Swimmer  
) ,
```

```
CTE_Coach  
AS
```

```
(  
SELECT CoachID  
FROM dbo.Coach  
) ,
```

```
CTE_Swimmer_Coach AS
```

```
(  
SELECT TOP 1000000 *  
FROM CTE_Swimmer  
CROSS JOIN CTE_Coach  
ORDER BY NEWID()  
)
```

```
INSERT INTO [dbo].[Swimmer_Coach] ([SwimmerID] , [CoachID])  
SELECT *  
FROM CTE_Swimmer_Coach
```

Тестирование

Оформление баг-репорта
в Excel

Swimmer

#	Description	Correction
1	There are no foreign key to link Swimmer and	ALTER TABLE dbo.Swimmer ADD CONSTRAINT FK_Swimmer FOREIGN
2	Some first letters of the LastName in lower	UPDATE [dbo].[Swimmer]
3	Some swimmers BirthDate are later than StartDate of the result.	UPDATE [dbo].[Swimmer] SET [BirthDate] = DATEADD(yy,-20, r.[StartDate]) FROM [dbo].[Swimmer] s JOIN [dbo].[Result] r ON s.[SwimmerId] = r.[SwimmerId] WHERE s.[BirthDate] > r.[StartDate]
4	Do normalization to the FirstName using uniform	UPDATE [dbo].[Swimmer]
5	Table has non-existent CoachID. Assign them ID	UPDATE [dbo].[Swimmer]
6	Field "PssportNumber" are not unique	ALTER TABLE [dbo].[Swimmer]
7	Add check to the BithDate, witch can't be more	ALTER TABLE [dbo].[Swimmer]

Pool

#	Description	Correction
1	Incorrect data format in [DeclaredTime] - float,	Need to convert type from float to time
2	Field "Style" need to fix fristyles to fristyle	UPDATE [dbo].[Result]
3	Create CHECK on field "Style"	ALTER TABLE [dbo].[Result]
4	Checking that StartDate in the Result table is in	SELECT R.[StartDate]

Result

#	Description	Correction
1	Column depth2 has only NULL	ALTER TABLE [dbo].[Pool] DROP column [depths2]

Создание объектов

Временная таблица #Table

Задание

Создайте объект для нахождения общего рейтинга топ 50 тренеров за всю историю соревнований, где рейтинг определяется общим количеством баллов за победы пловцов (1 место – 3 балла, 2 место – 2 балла, 3 место – 1 балл).

```

DROP TABLE IF EXISTS #Coach

SELECT c.LastName, c.FirstName, COUNT(s.SwimmerID ) OVER(PARTITION BY c.[CoachId] ) cnt,
       DENSE_RANK() OVER (PARTITION BY [CompetitionId] ORDER BY [DeclaredTime]) place
INTO #Coach
FROM dbo.Result r
JOIN [dbo].[Swimmer] s ON r.[SwimmerID] = s.[SwimmerID]
JOIN [dbo].[Coach] c ON s.[CoachID]=c.[CoachID]
GO

SELECT LastName, FirstName, cnt,
       CASE place
         WHEN 1 THEN 3
         WHEN 2 THEN 2
         WHEN 3 THEN 1
         ELSE 0
       END AS Mark
INTO #Mark
FROM #Coach
GO

SELECT LastName, FirstName, cnt * Mark AS Rating
INTO #Rating
FROM #Mark
GO

SELECT TOP 50 *
FROM #Rating
ORDER BY Rating DESC

```

Создание объектов

Представление (VIEW)

Для промежуточных
результатов использовалось
обобщенное табличное
выражение (CTE)

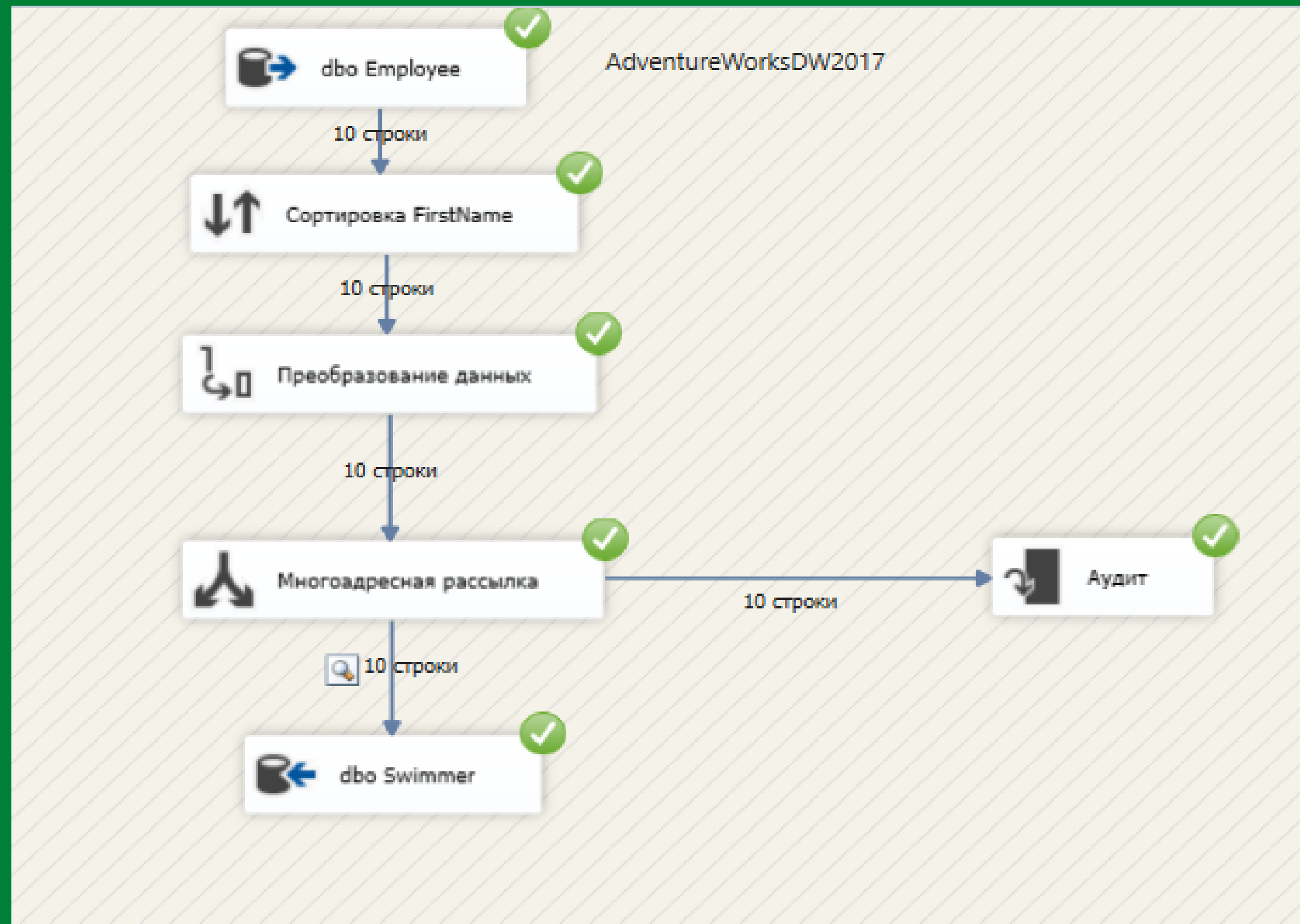
```
CREATE VIEW vTopCoach
AS
WITH CTE_Result
AS
(
    SELECT c.LastName, c.FirstName, COUNT(s.SwimmerID ) OVER(PARTITION BY c.[CoachId] ) cnt,
    DENSE_RANK() OVER (PARTITION BY [CompetitionId] ORDER BY [DeclaredTime]) place
    FROM dbo.Result r
    JOIN [dbo].[Swimmer] s ON r.[SwimmerID] = s.[SwimmerID]
    JOIN [dbo].[Coach] c ON s.[CoachID]=c.[CoachID]
),
CTE_Mark
AS
(
    SELECT LastName, FirstName, cnt,
    CASE place
    WHEN 1 THEN 3
    WHEN 2 THEN 2
    WHEN 3 THEN 1
    ELSE 0
    END AS Mark
    FROM CTE_Result
),
CTE_Rating
AS
(
    SELECT LastName, FirstName, cnt * Mark AS Rating
    FROM CTE_Mark
)
SELECT TOP 50 *
FROM CTE_Rating
ORDER BY Rating DESC

SELECT *
FROM vTopCoach
```


Создание пакетов ETL

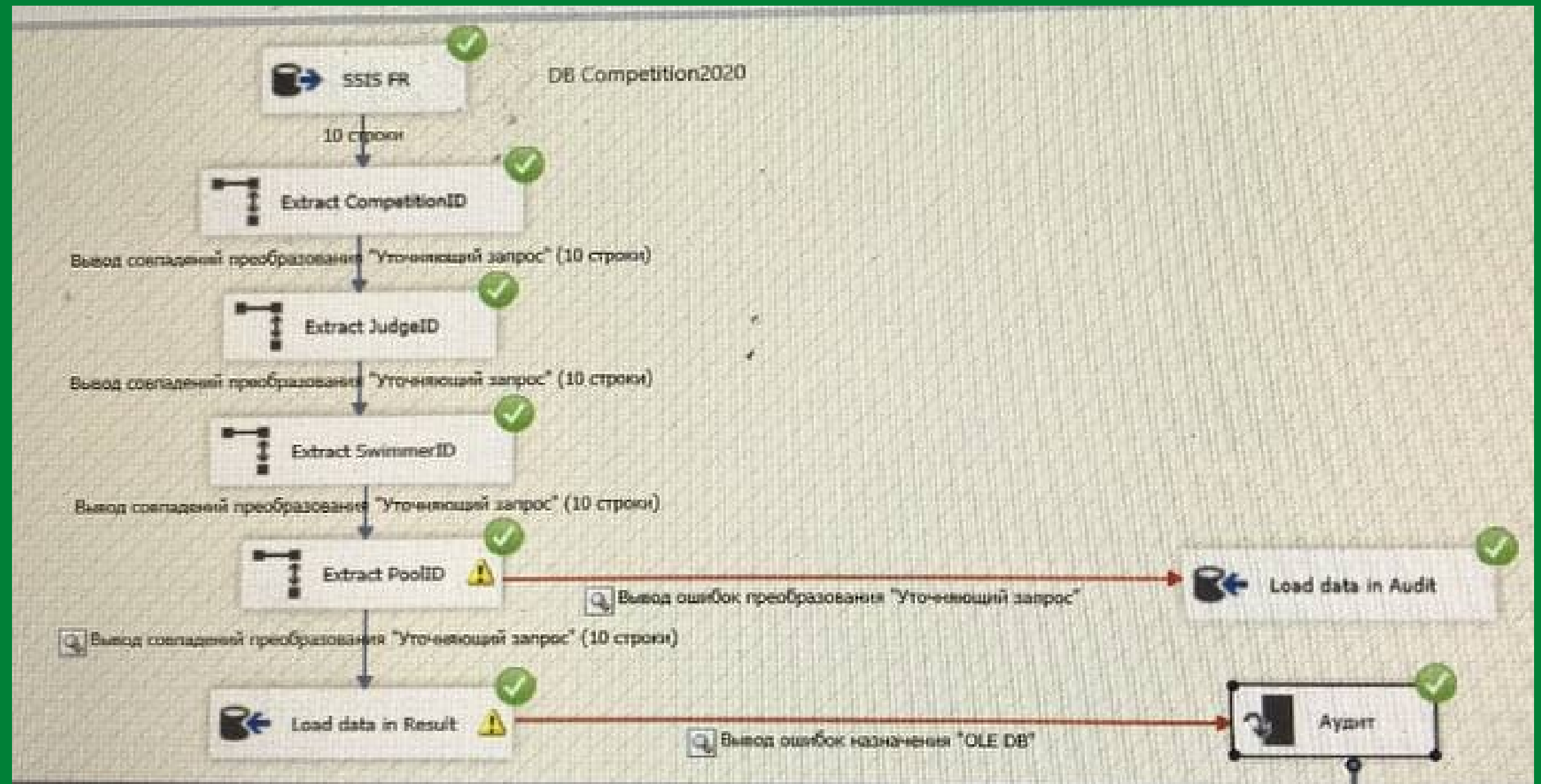
Создать SSIS пакет с добавлением данных (10 новых пловцов) в таблицу Swimmer из AdventureWorksDW2017.dbo.DimEmployee

MS SQL Server Integration Services (SSIS)



Создание пакетов ETL

Создать SSIS пакет с задачами “Уточняющий запрос” для добавления 10 новых результатов на используя БД, где вместо id записаны естественные ключи.



MS SQL Server Integration
Services (SSIS)

Визуализация данных в Power BI

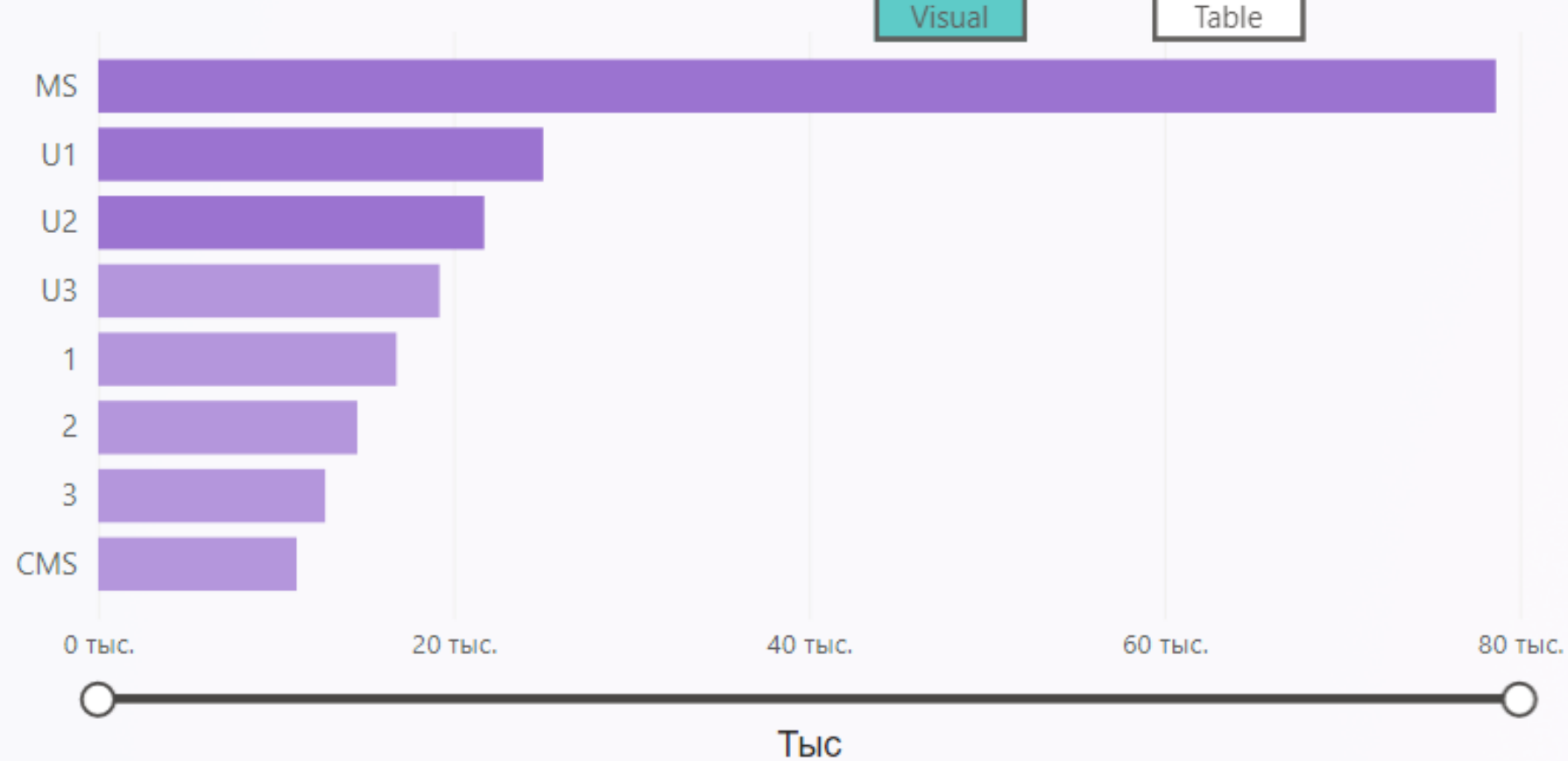


Swimming Competition

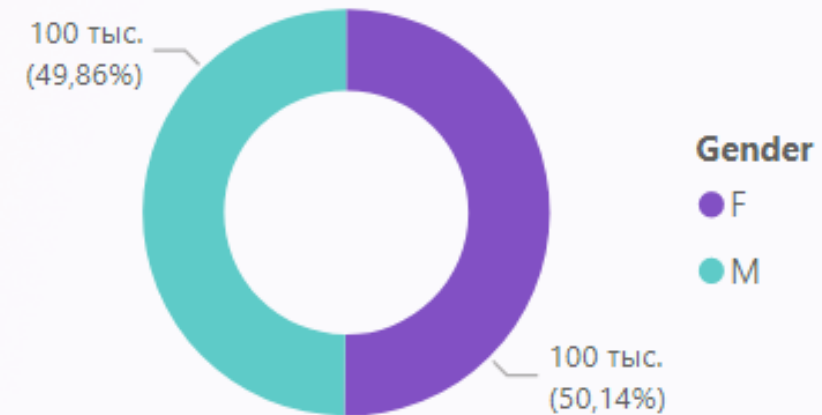
200,01 тыс.

Number of Swimmers

Total Rank of Swimmers



Gender of Swimmers



Age of Swimmers

Bce

Rank of Swimmers

- ☐ Выбрать все
- ☐ 1
- ☐ 2
- ☐ 3
- ☐ CMS
- ☐ MS
- ☐ U1
- ☐ U2
- ☐ U3

Swimmers

FirstName	LastName
Aaron	Abbas
Abraham	Abbas
Adam	Abbas
Adrian	Abbas
Adrienne	Abbas
Aidan	Abbas
Alexandra	Abbas
Alexandria	Abbas
Alexis	Abbas
Alicia	Abbas
Alyssa	Abbas
Amanda	Abbas
Amir	Abbas
Andrew	Abbas
Ann	Abbas
Anthony	Abbas
Arthur	Abbas
Audrey	Abbas

Top coach of swimmers by total age of swimmers

3722100

Age of Swimmers

1

CoachID

The Swimming Competition Report answers the following questions:

- shows the total number of swimmers for each rank
- gender ratio of swimmers

Визуализация данных в Tableau

SwimmingCompetition

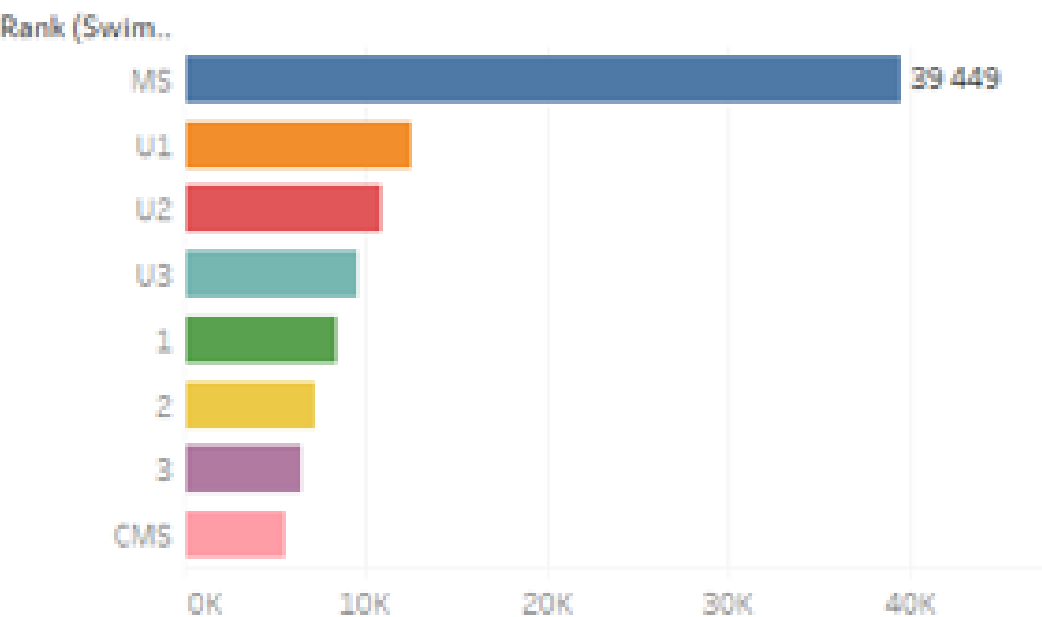


Count of Swimmer: 100 000

Total Rank

Bar Chart

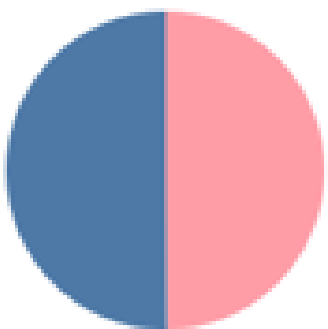
Table



Gender

Gender: M
Count of Swimmers: 50 117

Gender: F
Count of Swimmers: 49 883



Rank (Swimmer)
All

Gender (Swimmer)
All

Swimmers

FirstName (Swimmer)	LastName (Swimmer)	
A.	Alcorn	40
	Allison	34
	Armstrong	14
	Bacon	13
	Bankert	48
	Banks	2
	Barzdukas	33
	Bates	65
	Bauer	34

Analysis by parameters

Level 1	Level 2	Level 3	Level 4
Gabriel	Total		
	Vazquez	Total	
		M	Total
			1
			2
			3
			CMS
			MS
			117

Level 1
FirstName Coach

Level 2
LastName Coach

Level 3
Gender (Swimmer)

Level 4
Rank (Swimmer)

Активация Windows
Чтобы активировать Windows, перейдите на [www.microsoft.com/windows/activation](#)