

УО «Белорусский государственный университет информатики и
радиоэлектроники»

Кафедра ПОИТ

Отчёт по лабораторной работе № 2

Базы данных

Вариант 4

Выполнил:

Бобко А.В.

751001

Минск 2020

Задание № 1. Операции объединений INNER JOIN.

Вывести на экран неповторяющийся список должностей в каждом отделе, отсортированный по названию отдела. Посчитайте количество сотрудников, работающих в каждом отделе.

```
SELECT DISTINCT
    d.Name as DepartmentName,
    JobTitle,
    COUNT(*) OVER(PARTITION BY d.Name) as EmpCount
FROM HumanResources.Department AS d
INNER JOIN HumanResources.EmployeeDepartmentHistory AS edh
ON d.DepartmentID = edh.DepartmentID
INNER JOIN HumanResources.Employee AS e
ON e.BusinessEntityID = edh.BusinessEntityID
ORDER BY d.Name;
```

	DepartmentName	JobTitle	EmpCount
1	Document Control	Control Specialist	5
2	Document Control	Document Control Assistant	5
3	Document Control	Document Control Manager	5
4	Engineering	Design Engineer	7
5	Engineering	Engineering Manager	7
6	Engineering	Senior Design Engineer	7
7	Engineering	Senior Tool Designer	7
8	Engineering	Vice President of Enginee...	7
9	Executive	Chief Executive Officer	2
10	Executive	Chief Financial Officer	2
11	Facilities and Ma...	Facilities Administrative A...	7
12	Facilities and Ma...	Facilities Manager	7
13	Facilities and Ma...	Janitor	7
14	Facilities and Ma...	Maintenance Supervisor	7
15	Finance	Accountant	11
16	Finance	Accounts Manager	11
17	Finance	Accounts Payable Specia...	11
18	Finance	Accounts Receivable Sp...	11

Вывести на экран сотрудников, которые работают в ночную смену.

```
SELECT
    e.BusinessEntityID,
    JobTitle,
    Name,
    StartTime,
    EndTime
FROM HumanResources.Employee AS e
INNER JOIN HumanResources.EmployeeDepartmentHistory AS edh
ON e.BusinessEntityID = edh.BusinessEntityID
INNER JOIN HumanResources.Shift AS s
ON s.ShiftID = edh.ShiftID
WHERE Name = 'Night'
```

	BusinessEntityID	JobTitle	Name	StartTime	EndTime
1	40	Production Supervisor - WC60	Night	23:00:00.0000000	07:00:00.0000000
2	41	Production Technician - WC60	Night	23:00:00.0000000	07:00:00.0000000
3	42	Production Technician - WC60	Night	23:00:00.0000000	07:00:00.0000000
4	43	Production Technician - WC60	Night	23:00:00.0000000	07:00:00.0000000
5	44	Production Technician - WC60	Night	23:00:00.0000000	07:00:00.0000000
6	45	Production Technician - WC60	Night	23:00:00.0000000	07:00:00.0000000
7	46	Production Technician - WC60	Night	23:00:00.0000000	07:00:00.0000000
8	55	Production Supervisor - WC50	Night	23:00:00.0000000	07:00:00.0000000
9	56	Production Technician - WC50	Night	23:00:00.0000000	07:00:00.0000000
10	57	Production Technician - WC50	Night	23:00:00.0000000	07:00:00.0000000
11	58	Production Technician - WC50	Night	23:00:00.0000000	07:00:00.0000000
12	59	Production Technician - WC50	Night	23:00:00.0000000	07:00:00.0000000
13	60	Production Technician - WC50	Night	23:00:00.0000000	07:00:00.0000000
14	61	Production Technician - WC50	Night	23:00:00.0000000	07:00:00.0000000

Вывести на экран почасовые ставки сотрудников. Добавить столбец с информацией о предыдущей почасовой ставке для каждого сотрудника. Добавить еще один столбец с указанием разницы между текущей ставкой и предыдущей ставкой для каждого сотрудника.

```

SELECT
    e.BusinessEntityID,
    JobTitle,
    eph.Rate,
    ISNULL(prev_eph.Rate, 0.00) as PrevRate,
    eph.Rate - ISNULL(prev_eph.Rate, 0.00) as Increased
FROM HumanResources.Employee as e
INNER JOIN HumanResources.EmployeePayHistory as eph
ON e.BusinessEntityID = eph.BusinessEntityID
LEFT JOIN HumanResources.EmployeePayHistory as prev_eph
ON e.BusinessEntityID = prev_eph.BusinessEntityID
AND prev_eph.RateChangeDate < eph.RateChangeDate;

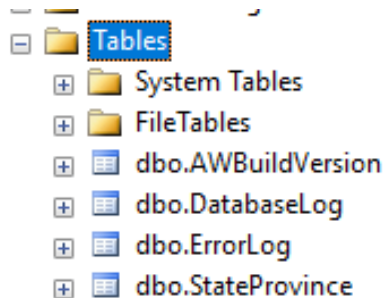
```

	BusinessEntityID	JobTitle	Rate	PrevRate	Increased
1	1	Chief Executive Officer	125,50	0,00	125,50
2	2	Vice President of Engineering	63,4615	0,00	63,4615
3	3	Engineering Manager	43,2692	0,00	43,2692
4	4	Senior Tool Designer	8,62	0,00	8,62
5	4	Senior Tool Designer	23,72	8,62	15,10
6	4	Senior Tool Designer	29,8462	8,62	21,2262
7	4	Senior Tool Designer	29,8462	23,72	6,1262
8	5	Design Engineer	32,6923	0,00	32,6923
9	6	Design Engineer	32,6923	0,00	32,6923
10	7	Research and Development Manager	50,4808	0,00	50,4808
11	8	Research and Development Engineer	40,8654	0,00	40,8654

Задание № 2. Создание и изменение таблиц, первичных ключей, ограничений. Оператор INSERT.

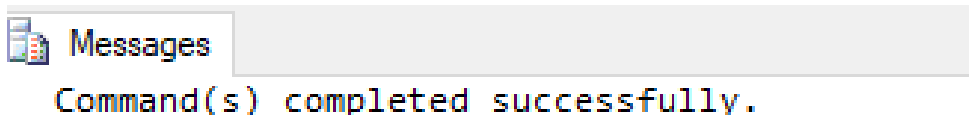
- а) создайте таблицу `dbo.StateProvince` с такой же структурой как `Person.StateProvince`, кроме поля `uniqueidentifier`, не включая индексы, ограничения и триггеры;

```
]CREATE TABLE dbo.StateProvince (  
    StateProvinceID INT NOT NULL,  
    StateProvinceCode NCHAR(3) NOT NULL,  
    CountryRegionCode NVARCHAR(3) NOT NULL,  
    IsOnlyStateProvinceFlag dbo.Flag NOT NULL,  
    Name dbo.Name NOT NULL,  
    TerritoryID INT NOT NULL,  
    ModifiedDate DATETIME NOT NULL  
);  
GO
```



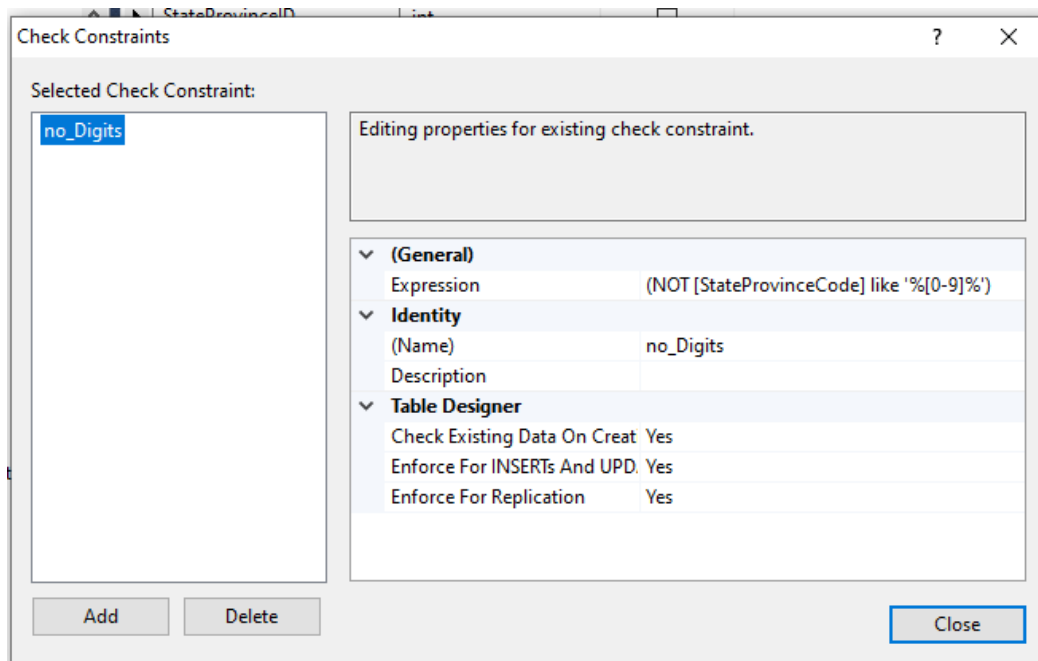
- б) используя инструкцию `ALTER TABLE`, создайте для таблицы `dbo.StateProvince` ограничение `UNIQUE` для поля `Name`;

```
]ALTER TABLE dbo.StateProvince  
ADD CONSTRAINT name_is_unique UNIQUE (Name);  
GO
```



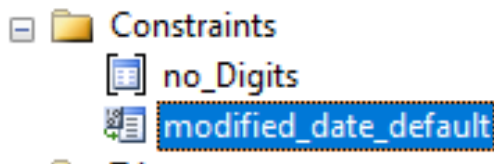
- в) используя инструкцию `ALTER TABLE`, создайте для таблицы `dbo.StateProvince` ограничение для поля `CountryRegionCode`, запрещающее заполнение этого поля цифрами;

```
]ALTER TABLE dbo.StateProvince  
ADD CONSTRAINT no_Digits  
CHECK (StateProvinceCode NOT LIKE '%[0-9]%')
```



d) используя инструкцию ALTER TABLE, создайте для таблицы dbo.StateProvince ограничение DEFAULT для поля ModifiedDate, задайте значение по умолчанию текущую дату и время;

```
ALTER TABLE dbo.StateProvince
ADD CONSTRAINT modified_date_default
    DEFAULT GETDATE() FOR ModifiedDate;
```



e) заполните новую таблицу данными из Person.StateProvince. Выберите для вставки только те данные, где имя штата/государства совпадает с именем страны/региона в таблице CountryRegion;

```
INSERT INTO dbo.StateProvince
SELECT
    st.StateProvinceID,
    st.StateProvinceCode,
    st.CountryRegionCode,
    st.IsOnlyStateProvinceFlag,
    st.Name,
    st.TerritoryID,
    st.ModifiedDate
FROM Person.StateProvince AS st
INNER JOIN Person.CountryRegion AS cr
ON st.Name = cr.Name;
GO
```

```
SELECT * FROM dbo.StateProvince
```

	StateProvinceID	StateProvinceCode	CountryRegionCode	IsOnlyStateProvinceFlag	Name	TerritoryID	ModifiedDate
1	5	AS	AS	1	American Samoa	1	2008-03-11 10:17:21.587
2	84	FR	FR	1	France	7	2002-06-01 00:00:00.000
3	17	GA	US	0	Georgia	5	2008-03-11 10:17:21.587
4	18	GU	US	0	Guam	4	2008-03-11 10:17:21.587
5	34	MH	MH	1	Marshall Islands	9	2008-03-11 10:17:21.587
6	16	FM	FM	1	Micronesia	9	2008-03-11 10:17:21.587
7	38	MP	MP	1	Northem Maria...	9	2008-03-11 10:17:21.587
8	62	PW	PW	1	Palau	9	2008-03-11 10:17:21.587
9	61	PR	US	0	Puerto Rico	5	2008-03-11 10:17:21.587

- f) удалите поле IsOnlyStateProvinceFlag, а вместо него создайте новое CountryNum типа int допускающее null значения.

```
ALTER TABLE dbo.StateProvince DROP COLUMN IsOnlyStateProvinceFlag;

ALTER TABLE dbo.StateProvince ADD CountryNum INT NULL;
```

До:

Columns	
StateProvinceID	(int, not null)
StateProvinceCode	(nchar(3), not null)
CountryRegionCode	(nvarchar(3), not null)
IsOnlyStateProvinceFlag	(Flag(bit), not null)
Name	(Name(nvarchar(50)), not null)
TerritoryID	(int, not null)
ModifiedDate	(datetime, not null)

После:

Columns	
StateProvinceID	(int, not null)
StateProvinceCode	(nchar(3), not null)
CountryRegionCode	(nvarchar(3), not null)
Name	(Name(nvarchar(50)), not null)
TerritoryID	(int, not null)
ModifiedDate	(datetime, not null)
CountryNum	(int, null)