

a) Процедура без параметров, формирующая список преподавателей и дисциплин, которые они ведут в текущем семестре (если преподаватель в тек.семестре ничего не ведет, то вместо названия дисциплины вывести «нет нагрузки»)

```

CREATE PROCEDURE [Дисциплины в текущем семестре по преподавателям]
AS
BEGIN
    DECLARE @Year INT = YEAR(GETDATE())
    DECLARE @Month INT = MONTH(GETDATE())
    DECLARE @Semester INT =
        CASE
            WHEN @Month BETWEEN 8 AND 12 OR @Month = 1 THEN 1
            WHEN @Month BETWEEN 2 AND 7 THEN 2
        END
    DECLARE @YearString NVARCHAR(9)

    IF @Semester = 1
        SET @YearString = CAST(@Year AS NVARCHAR(4)) + '-' + CAST(@Year + 1 AS NVARCHAR(4))
    ELSE
        SET @YearString = CAST(@Year - 1 AS NVARCHAR(4)) + '-' + CAST(@Year AS NVARCHAR(4))

    SELECT
        p.[ФИО] AS Преподаватель,
        CASE
            WHEN EXISTS (
                SELECT *
                FROM [dbo].[Нагрузка] n JOIN [dbo].[Учебный_план] np ON n.[Учебный_план_ID] = np.[ID]
                WHERE
                    n.[Преподаватель_ID] = p.[ID]
                    AND np.[Учебный год] = @YearString
                    AND np.[Семестр] = @Semester
            ) THEN (
                SELECT STRING_AGG(d.[Название], ', ')
                FROM (
                    SELECT DISTINCT
                        d.[Название]
                    FROM [dbo].[Нагрузка] n JOIN [dbo].[Учебный_план] np ON n.[Учебный_план_ID] = np.[ID]
                    JOIN [dbo].[Дисциплина] d ON np.[Дисциплина_ID] = d.[ID]
                    WHERE
                        n.[Преподаватель_ID] = p.[ID]
                        AND np.[Учебный год] = @YearString
                        AND np.[Семестр] = @Semester
                ) d
            )
            ELSE 'нет нагрузки'
        END AS [Дисциплины в текущем семестре]
    FROM [dbo].[Преподаватель] p
    ORDER BY p.[ФИО]
END

```

EXEC [Дисциплины в текущем семестре по преподавателям]

	Преподаватель	Дисциплины в текущем семестре
1	Васильев Игорь Николаевич	нет нагрузки
2	Громов Павел Сергеевич	нет нагрузки
3	Зайцева Татьяна Викторовна	нет нагрузки
4	Иванов Александр Сергеевич	нет нагрузки
5	Козлова Елена Петровна	нет нагрузки
6	Никитин Алексей Викторович	нет нагрузки
7	Петрова Мария Владимировна	Математический анализ
8	Сидоров Дмитрий Иванович	нет нагрузки
9	Смирнова Анна Михайловна	нет нагрузки
10	Фёдорова Ольга Дмитриевна	нет нагрузки

b) Процедура, на входе получающая название специальности и номер курса и формирующая список экзаменов и зачетов, которые она должна сдавать в следующую сессию в виде: дисциплина, вид отчетности

```

CREATE PROCEDURE [Сессия по специальности и курсу]
    @Specialty NVARCHAR(100),
    @Course INT
AS
BEGIN
    DECLARE @Year INT = YEAR(GETDATE())
    DECLARE @Month INT = MONTH(GETDATE())
    DECLARE @Semester INT =
        CASE
            WHEN @Month BETWEEN 8 AND 12 OR @Month = 1 THEN 1
            WHEN @Month BETWEEN 2 AND 7 THEN 2
        END
    DECLARE @YearString NVARCHAR(9)

    IF @Semester = 1
        SET @YearString = CAST(@Year AS NVARCHAR(4)) + '-' + CAST(@Year + 1 AS NVARCHAR(4))
    ELSE
        SET @YearString = CAST(@Year - 1 AS NVARCHAR(4)) + '-' + CAST(@Year AS NVARCHAR(4))

    SELECT
        d.[Название] AS Дисциплина,
        vo.[Название] AS [Тип отчётности]
    FROM
        [dbo].[Учебный_план] up
    JOIN [dbo].[Дисциплина] d ON up.[Дисциплина_ID] = d.[ID]
    JOIN [dbo].[Тип_отчётности] vo ON up.[Тип_отчётности_ID] = vo.[ID]
    JOIN [dbo].[Специальность] s ON up.[Специальность_ID] = s.[ID]
    WHERE
        s.[Название] = @Specialty
        AND up.[Курс] = @Course
        AND up.[Учебный год] = @YearString
        AND up.[Семестр] = @Semester
    END

    EXEC [Сессия по специальности и курсу]
        @Specialty = 'Прикладная математика и информатика',
        @Course = 1

    EXEC [Сессия по специальности и курсу]
        @Specialty = 'Математика и компьютерные науки',
        @Course = 2

    --Пример если нет экзаменов и зачётов
    EXEC [Сессия по специальности и курсу]
        @Specialty = 'Бизнес-информатика',
        @Course = 2

```

	Дисциплина	Тип отчётности
1	Математический анализ	Зачёт

	Дисциплина	Тип отчётности
1	История	Экзамен

	Дисциплина	Тип отчётности
--	------------	----------------

с) Процедура, на входе получающая ФИО преподавателя, выходной параметр – общее количество часов нагрузки в 1-м семестре текущего года

```
CREATE PROCEDURE [Нагрузка в 1 семестре текущего учебного года]
    @Name NVARCHAR(100),
    @AllHours INT OUTPUT
AS
BEGIN
    DECLARE @Year INT = YEAR(GETDATE())
    DECLARE @Month INT = MONTH(GETDATE())
    DECLARE @YearString NVARCHAR(9)

    IF @Month BETWEEN 7 AND 12
        SET @YearString = CAST(@Year AS NVARCHAR(4)) + '-' + CAST(@Year + 1 AS NVARCHAR(4))
    ELSE
        SET @YearString = CAST(@Year - 1 AS NVARCHAR(4)) + '-' + CAST(@Year AS NVARCHAR(4))

    SELECT
        @AllHours = ISNULL(SUM(n.[Назначенные часы]), 0)
    FROM
        [dbo].[Нагрузка] n
        JOIN [dbo].[Преподаватель] p ON n.[Преподаватель_ID] = p.[ID]
        JOIN [dbo].[Учебный_план] up ON n.[Учебный_план_ID] = up.[ID]
    WHERE
        up.[Учебный год] = @YearString
        AND up.Семestr = 1
        AND p.ФИО = @Name
END

DECLARE @Hours INT

EXEC [Нагрузка в 1 семестре текущего учебного года]
    @Name = 'Петрова Мария Владимировна',
    @AllHours = @Hours OUTPUT

PRINT 'Общее количество часов в 1 семестре текущего учебного года: ' + CAST(@Hours AS NVARCHAR(10))

----- 
Общее количество часов в 1 семестре текущего учебного года: 30
Время выполнения: 2025-12-16T14:29:43.6372922+03:00
```

d) Процедура, вызывающая вложенную процедуру, которая подсчитывает среднее количество часов нагрузки в год по преподавателям . Главная процедура выводит список преподавателей, имеющих нагрузку выше средней

```
CREATE PROCEDURE [Средняя нагрузка в текущем году]
    @AvgLoad DECIMAL(10,2) OUTPUT
AS
BEGIN
    DECLARE @Year INT = YEAR(GETDATE())
    DECLARE @Month INT = MONTH(GETDATE())
    DECLARE @YearString NVARCHAR(9)

    IF @Month BETWEEN 7 AND 12
        SET @YearString = CAST(@Year AS NVARCHAR(4)) + '-' + CAST(@Year + 1 AS NVARCHAR(4))
    ELSE
        SET @YearString = CAST(@Year - 1 AS NVARCHAR(4)) + '-' + CAST(@Year AS NVARCHAR(4))

    SELECT
        @AvgLoad = AVG(CAST([Годовая Нагрузка] AS DECIMAL(10,2)))
    FROM (
        SELECT
            p.[ID],
            SUM(n.[Назначенные часы]) AS [Годовая Нагрузка]
        FROM
            [dbo].[Преподаватель] p
            JOIN [dbo].[Нагрузка] n ON p.[ID] = n.[Преподаватель_ID]
            JOIN [dbo].[Учебный_план] up ON n.[Учебный_план_ID] = up.[ID]
        WHERE
            up.[Учебный год] = @YearString
        GROUP BY
            p.[ID]
    ) AS pn
END

CREATE PROCEDURE [Преподаватели с нагрузкой выше средней]
AS
BEGIN
    DECLARE @Avg_Load DECIMAL(10,2)
    DECLARE @Year1 INT = YEAR(GETDATE())
    DECLARE @Month1 INT = MONTH(GETDATE())
    DECLARE @YearString1 NVARCHAR(9)

    IF @Month1 BETWEEN 7 AND 12
        SET @YearString1 = CAST(@Year1 AS NVARCHAR(4)) + '-' + CAST(@Year1 + 1 AS NVARCHAR(4))
    ELSE
        SET @YearString1 = CAST(@Year1 - 1 AS NVARCHAR(4)) + '-' + CAST(@Year1 AS NVARCHAR(4))

    EXEC [Средняя нагрузка в текущем году] @Avg_Load OUTPUT

    SELECT
        p.[ФИО] AS Преподаватель,
        SUM(n.[Назначенные часы]) AS [Годовая нагрузка],
        @Avg_Load AS [Средняя нагрузка],
        @YearString1 AS [Учебный год]
    FROM
        [dbo].[Преподаватель] p
        JOIN [dbo].[Нагрузка] n ON p.[ID] = n.[Преподаватель_ID]
        JOIN [dbo].[Учебный_план] up ON n.[Учебный_план_ID] = up.[ID]
    WHERE
        up.[Учебный год] = @YearString1
    GROUP BY
        p.[ФИО]
    HAVING
        SUM(n.[Назначенные часы]) > @Avg_Load
    ORDER BY
        SUM(n.[Назначенные часы]) DESC
END
```

```
DECLARE @Load DECIMAL(10, 2)

EXEC [Средняя нагрузка в текущем году]
    @AvgLoad = @Load OUTPUT

PRINT 'Средняя нагрузка: ' + CAST(@Load AS NVARCHAR(10)) Средняя нагрузка: 35
```

```
EXEC [Преподаватели с нагрузкой выше средней]
```

	Преподаватель	Годовая нагрузка	Средняя нагрузка	Учебный год
1	Васильев Игорь Николаевич	40	35.00	2025-2026