

Направление «Специалист отдела поддержки MES», «Норникель Спутник»

Аннотация

Добро пожаловать на виртуальную стажировку **Shift + Enter** от «Норникель Спутника». Тебе предстоит примерить роль младшего специалиста поддержки MES.

MES (manufacturing execution system) – система управления производственными процессами, которая решает задачи по синхронизации, координации, анализу и оптимизации выпуска продукции.

В твои обязанности как младшего специалиста поддержки MES входит:

- Анализ работы системы и проведение оптимизации для улучшения ее показателей.
- Участие в доработке функционала системы.
- Написание и редактирование запросов SQL/T-SQL.
- Администрирование систем управления базами данных (СУБД) MS SQL, PostgreSQL.

Развиваемые компетенции

По результатам выполнения заданий ты научишься:

1. Добавлять значения в таблицы, используя синтаксис T-SQL.
2. Составлять запросы, которые выполняли бы заданные условия.

А также сможешь прокачать такие навыки, как работа с информацией, аналитические способности, внимательность и аккуратность к деталям.

Описание подзадач

В «Норникель Спутнике» каждый сотрудник – часть большой и дружной команды. Помоги коллегам разобраться с SQL-запросами и продемонстрируй знание этого языка на уровне написания процедур и функций.

Выполнение всего блока заданий займет у тебя не более 100–130 минут.

Рекомендуемый тайминг

1. 20–25 минут на первое задание.
2. 30–40 минут на второе задание.
3. 60–75 минут на третье задание.

Информация о загрузке решения

Этот проект содержит несколько подзадач. Можно загрузить файл, содержащий решение части заданий, но по возможности постарайся сделать их все.

Желаем удачи!

Задание 1. Добавь значения в таблицу и выбери из них самые новые по дате изменения

Утром после вводной встречи, на которой ты познакомился с командой службы поддержки MES, твой руководитель прислал тебе письмо с первым заданием стажировки.

Привет!

Как ты слышал на встрече с коллегами, «Норникель Спутник» использует MS SQL Server – систему управления реляционными базами данных, основным языком запросов которой является Transact-SQL, или T-SQL. Это расширенный стандарт SQL, включающий процедурное программирование¹, локальные переменные, различные вспомогательные функции для обработки строк, дат и математических выражений.

При работе над твоим первым заданием ты должен:

- Добавить значения в таблицу, которая должна содержать информацию о некоторых показателях (Tag – имя / Value – значение) и дату их изменения (DateTimeChange). В итоге таблица должна выглядеть следующим образом:

	Tag	Value	DateTimeChange
1	Income	5	2022-01-01 00:00:00.000
2	Income	7	2022-03-05 00:00:00.000
3	Income	15	2022-08-02 00:00:00.000
4	Award	5	2022-02-03 00:00:00.000
5	Award	2	2022-04-16 00:00:00.000
6	Outcome	1	2022-01-01 00:00:00.000
7	Outcome	7	2022-02-14 00:00:00.000
8	Outcome	10	2022-07-29 00:00:00.000
9	Outcome	14	2022-08-02 00:00:00.000

Твой коллега решил помочь тебе и уже использовал оператор DECLARE для объявления локальной переменной в SQL Server с указанием ее имени (@Tb1) и типа данных:

```
DECLARE @Tb1 TABLE
(
  Tag varchar(128),
  Value int,
  DateTimeChange datetime
)
```

После чего он посоветовал тебе использовать функцию INSERT INTO.

- Выбрать из таблицы последние значения для каждого показателя (то есть с наиболее поздней датой). В итоге должна получиться следующая выборка:

	Tag	Value	DateTimeChange
1	Award	2	2022-04-16 00:00:00.000
2	Income	15	2022-08-02 00:00:00.000
3	Outcome	14	2022-08-02 00:00:00.000

Hints. Запрос должен работать и при добавлении дополнительных строк с показателями.

В T-SQL функция SELECT для возврата данных аналогична той, что есть в SQL:

¹ Процедурное программирование – это программирование, при котором последовательно выполняемые операторы можно собрать в подпрограммы, то есть более крупные целостные единицы кода, с помощью механизмов самого языка.

```
SELECT Tag, Value, DateTimeChange FROM @Tb1
```

Задачу нужно решить одним запросом, изменив исходный SELECT.

Пожалуйста, пришли мне свое решение как можно быстрее. Спасибо!

Полезные материалы

- Для проверки правильности запросов можно использовать сервис [SQLite Online](#), в левом меню которого нужно выбрать MS SQL.
- [Примеры](#) использования оператора INSERT.
- [Статья](#) о команде INSERT INTO.

Формат конечного результата

Запрос в файле формата .docx.

Форма загрузки результата

Пожалуйста, загрузи свой вариант ответа в формате zip-архива, используя инструмент «Загрузить решение». Необходимо сформировать единый zip-архив, содержащий решение одного или всех заданий по выбранной специальности.

Пример решения

У тебя будет возможность ознакомиться с примером решения задания от эксперта после отправки собственной версии.

Задание 2. Напиши запрос для отображения динамики финансовых показателей

Твое следующее задание тоже связано с написанием запроса для работы с данными. Позже ты увидел письмо от руководителя.

Привет, поздравляю с успешным выполнением первого задания! Теперь предлагаю поработать над составлением более сложного запроса.

У нас есть следующая выборка данных, представленных в таблице:

	DT	TotalMoney
1	2020-01-01 00:00:00.000	500
2	2020-02-01 00:00:00.000	1000
3	2020-03-01 00:00:00.000	3000
4	2020-04-01 00:00:00.000	5000
5	2020-05-01 00:00:00.000	6000
6	2020-06-01 00:00:00.000	10000
7	2020-07-01 00:00:00.000	9000
8	2020-08-01 00:00:00.000	13000
9	2020-09-01 00:00:00.000	19000
10	2020-10-01 00:00:00.000	17000
11	2020-11-01 00:00:00.000	26000
12	2020-12-01 00:00:00.000	40000
13	2021-01-01 00:00:00.000	35000
14	2021-02-01 00:00:00.000	41000

Это данные о финансах одного из подразделений компании на определенную дату. Наш коллега из департамента финансов попросил составить запрос, чтобы отобразить прибыль или убыток за каждый месяц. В случае если месяц прибыльный то отобразить число в поле прибыли, если убыточный – в поле убытка.

В результате у тебя должна получиться следующая выборка:

	Period_Start	Period_End	Income	Loss
1	2020-01-01 00:00:00.000	2020-02-01 00:00:00.000	500	0
2	2020-02-01 00:00:00.000	2020-03-01 00:00:00.000	2000	0
3	2020-03-01 00:00:00.000	2020-04-01 00:00:00.000	2000	0
4	2020-04-01 00:00:00.000	2020-05-01 00:00:00.000	1000	0
5	2020-05-01 00:00:00.000	2020-06-01 00:00:00.000	4000	0
6	2020-06-01 00:00:00.000	2020-07-01 00:00:00.000	0	-1000
7	2020-07-01 00:00:00.000	2020-08-01 00:00:00.000	4000	0
8	2020-08-01 00:00:00.000	2020-09-01 00:00:00.000	6000	0
9	2020-09-01 00:00:00.000	2020-10-01 00:00:00.000	0	-2000
10	2020-10-01 00:00:00.000	2020-11-01 00:00:00.000	9000	0
11	2020-11-01 00:00:00.000	2020-12-01 00:00:00.000	14000	0
12	2020-12-01 00:00:00.000	2021-01-01 00:00:00.000	0	-5000
13	2021-01-01 00:00:00.000	2021-02-01 00:00:00.000	6000	0

Запрос должен работать и при добавлении дополнительных строк с показателями.

Hints. Задачу нужно решить одним запросом (изменив исходный `SELECT [DT], [TotalMoney]` `FROM @Tb1` на запрос любой величины). При работе над запросом обрати внимание на операторы `ORDER BY`, `CASE WHEN ... THEN`, `WHERE`.

Спасибо!



Полезные материалы

Подробнее о подобных запросах можно почитать в [этой статье](#).

Формат конечного результата

Запрос в файле формата .docx.

Форма загрузки результата

Пожалуйста, загрузи свой вариант ответа в формате zip-архива, используя инструмент «Загрузить решение». Необходимо сформировать единый zip-архив, содержащий решение одного или всех заданий по выбранной специальности.

Пример решения

У тебя будет возможность ознакомиться с примером решения задания от эксперта после отправки собственной версии.

Задание 3. Составь запрос уровня PRO и посчитай количество рабочих дней сотрудников

Твое финальное задание посвящено работе над более сложным запросом. Открыв почту, ты внимательно изучаешь детали проекта.

Привет!

На следующем этапе стажировки тебе задана выборка:

	Name	Job_start	Job_end
1	Peter	2021-01-01	2021-02-01
2	Ivan	2021-03-01	2021-03-08
3	Boris	2022-02-01	2022-02-15
4	Pavel	2020-07-15	2020-08-25
5	Vadim	2018-12-05	2019-01-20

Требуется написать запрос для вычисления количества рабочих дней, которое проработали сотрудники, присоединившиеся к компании не более 100 дней назад. Нерабочими днями считаются суббота и воскресенье (праздники опускаются, так как при работе сменами они не учитываются). В итоге должна получиться следующая выборка:

	Name	days
1	Boris	11
2	Ivan	6
3	Pavel	30
4	Peter	22
5	Vadim	33

Задачу нужно решить одним запросом (изменив исходный `SELECT [Name],[Job_start],[Job_end] FROM @Tb1` на запрос любой величины). При добавлении новых сотрудников запрос должен работать корректно.

Жду файл, содержащий запрос, на почту.

Спасибо!

Полезные материалы

Подробнее о рекурсии в SQL, которую нужно использовать при составлении запроса можно прочитать в [этой](#) статье и на [этом](#) сайте (на английском языке).

Формат конечного результата

Запрос в файле формата .docx.

Форма загрузки результата

Пожалуйста, загрузи свой вариант ответа в формате zip-архива, используя инструмент «Загрузить решение». Необходимо сформировать единый zip-архив, содержащий решение одного или всех заданий по выбранной специальности.



Пример решения

У тебя будет возможность ознакомиться с примером решения задания от эксперта после отправки собственной версии.