Управление данными в таблицах

Цели урока

В этом уроке вы узнаете как:

- Создавать правила
- Создавать умолчания
- Работать с данными в таблице при помощи
  - Команды INSERT
  - Команды UPDATE
  - Команды DELETE
- Сокращать таблицу

Управление данными в таблицах

CPC 5_3

1. Какие бывают различные типы ограничений?
2. Что такое таблица?
3. Какие два аттрибута необходимо установить для столбца IDENTITY?
4. Какие параметры необходимы для создания пользовательского типа данных?
5. Какая встроенная процедура используется для снятия пользовательского типа данных?

Пример создания правила

Постановка задачи
Почтовый индекс в таблице Newspaper должен быть символьного типа и иметь следующую структуру:
[0-9][0-9][0-9][0-9][0-9]-[0-9][0-9][0-9][0-9]
Например: 42482-4353
Не меняя структуру таблицы, как можно гарантировать, что это требование будет соблюдено?

Управление данными в таблицах

Решение

1. *Определить способ реализации ограничения без изменения структуры таблицы.*
2. *Составить команду для создания правила.*
3. *Создать правило.*
4. *Назначить правило столбцу.*
5. *Проверить ограничения путем вставки данных в таблицу.*

Управление данными в таблицах

Определение способа реализации ограничения без ищменения структуры таблицы

Правило дает механизм обеспечения целостности домена для столбцов или пользовательских типов данных. Правило применяется столбцу или пользовательскому типу данных перед введением команды INSERT или UPDATE.

Результат

- ***Ограничение может быть реализовано при помощи правил.***

Управление данными в таблицах

Составление команды для создания правила.

- *Команда CREATE RULE применяется для создания правила*
- *Синтаксис*

```
CREATE RULE rule_name
AS conditional_expression
```

Действие

- *Правило должно быть применено к атрибуту cNewspaperCode.*
- *Применяемым условием является:*

```
@ZipCode LIKE '[0-9][0-9][0-9][0-9][0-9]-[0-9][0-9][0-9][0-9]'
```

- *Именем правила должно быть rulZipCode.*
- *Правило может быть создано следующим образом:*

```
CREATE RULE rulZipCode
AS
@ZipCode LIKE '[0-9][0-9][0-9][0-9][0-9]- [0-9][0-9][0-9][0-9]'
```

Создание правила

Действие

- *В окне Query Analyzer набрать запрос*

```
CREATE RULE rulZipCode
AS
@ZipCode LIKE
'[0-9][0-9][0-9][0-9][0-9]-[0-9][0-9][0-9][0-9]'
```

- *Нажать F5 для выполнения запроса.*

Управление данными в таблицах

Связывание правила со столбцом.

Связывание правил

- *Правило может быть назначено при помощи системной процедуры sp_bindrule.*
- *Синтаксис*

```
sp_bindrule имя_правила, имя_объекта [, FUTUREONLY]
```

Снятие правил

- *Правило может быть снято со столбца или пользовательского типа данных при помощи системной процедуры sp_unbindrule.*
- *Синтаксис*

```
sp_unbindrule имя_объекта [, FUTUREONLY]
```

Действие

- *В окне Query Analyzer набрать:*

```
sp_bindrule rulZipCode,'Newspaper.cZip'
```

- *Нажать F5 для исполнения кода.*

Управление данными в таблицах

Проверка работы ограничения путем вставки данных в таблицу.

Действие		
Контрольный пример	Вставка cZip	Результат
1	3452345	Строка не вставлена, поскольку почтовый индекс задан в не допустимом формате.
2	34563-5678	Строка будет вставлена, поскольку почтовый индекс задан в допустимом формате.

Пример создания умолчания

Постановка задачи:
Оператор ввода данных жалуется, что в большинстве строк,он постоянно набирает код 001 атрибута cCountryCode таблицы Newspaper. Необходимо упростить задачу ввода данных без изменения структуру таблицы.

Управление данными в таблицах

Решение

1. Определить, как может быть упрощена задача ввода данных.
2. Составить команду для создания умолчания.
3. Создать умолчание.
4. Назначить умолчание для столбца.
5. Проверка того, что умолчание добавило строку со значением DEFAULT

Управление данными в таблицах

Определение, как может быть упрощена задача ввода данных.

Умолчание

- ***Умолчание, это постоянное значение, назначенное для столбца, в который пользователю не требуется вставлять значения.***

Результат

- ***Задача ввода данных может быть упрощена при помощи умолчаний.***

Управление данными в таблицах

Составить команду создания умолчания таблицы

Команда CREATE DEFAULT

- *Синтаксис*

```
CREATE DEFAULT default_name
AS constant_expression
```

Действие

- *Умолчание должно быть применено к таблице Newspaper.*
- *cCountryCode – столбец, к которому будет применено умолчание.*
- *Значение по умолчанию “001”.*
- *Код программы для создания умолчания:*

```
CREATE DEFAULT defCountry
AS '001'
```

Управление данными в таблицах

Создать умолчание

Действие

- ***В окне Query Analyzer набрать запрос***
- ***Нажать F5 для исполнения кода.***

Управление данными в таблицах

Связывание умолчания со столбцом.

- *После создания DEFAULT, оно должно быть связано либо со столбцом, либо с пользовательским типом данных.*
- *Синтаксис*

  ```
  sp_bindefault имя_умолчания, имя_объекта [, FUTUREONLY]
  ```
- *Снятие умолчаний - Умолчание может быть снято со столбца или пользовательского типа данных при помощи системной процедуры sp_unbindefault.*
- *Синтаксис*

  ```
  sp_unbindefault имя_объекта [, FUTUREONLY]
  ```

Действие

- *В окне Query Analyzer набрать:*

  ```
  sp_bindefault
  defCountry,'Newspaper.cCountryCode'
  ```
- *Нажать F5 для исполнения кода.*

Управление данными в таблицах

Прверка умолчания путем добавления строки со значением DEFAULT

Действие

- *В окне Query Analyzer набрать запрос*

```
INSERT INTO Newspaper
    VALUES('0008','Kansas Today','Kansas','Genral','Robin
Paul','1925 Shawnee Dr  ','Kansas City','Kansas','66106-
3025',DEFAULT,'(913)362-  9529','(913)362-9515')
```

- *Нажать F5 для исполнения кода.*

Работа с данными

- ***Язык манипулирования данных - Манипулирование данных представляет собой вставку, изменение и удаление данных. Для манипулирования данными применяется язык манипулирования данными (Data Manipulation Language) SQL-транзакций.***
- ***При работе с базой данных обычно выполняются три операции:***
  - ***Вставка строк***
  - ***Обновление строк***
  - ***Удаление строк***

Пример внесения сведений в Таблицу

Постановка задачи

Набор кандидатов выполнен через кадровые агентства. Название одного из новых агентств по кадрам – «Head Hunters» Сведениями о Head Hunters являются:

Атрибуты	Данные
Agency Code (код агентства)	0010
Name (название)	Head Hunters
Адрес	223 Hill Street
City (город)	Cleveland
State (штат)	Ohio
Почтовый индекс	44167-5943
Phone (телефон)	(440)345-8872
Fax (факс)	(440)345-8943
Charge (расход)	7
Total Paid (всего оплачено)	1000

Эти сведения должны быть сохранены в таблицу RecruitmentAgencies

Управление данными в таблицах

Решение

1. *Выбор таблицы для добавления информации.*
2. *Определение значений для вставки.*
3. *Вставка строк в таблицу.*
4. *Проверка вставки данных путем запроса таблицы.*

Управление данными в таблицах

Определить, в какую таблицу будет добавляться информация

Результат

- ***Информация должна быть добавлена в таблицу RecruitmentAgencies.***

Определение значений для вставки.

Результат

- ***В талицу RecruitmentAgencies должны быть вставлены значения:***
  cAgencyCode = '0010'
  cName = 'Head Hunters ',
  vAddress = '223 Hill Street ',
  cCity = 'Cleveland',
  cState = 'Ohio'
  cZip = '44167-5943',
  cPhone = '(440)345-8872',
  cFax = '(440)345-8943',
  siPercentageCharge = 7,
  mTotalPaid = 1000

Управление данными в таблицах

Вставка строк в таблицу.

Команда INSERT

- *Требуется добавлять данные в базу для работы с последней информацией об организации и выполненными ею транзакциями. Это делает командой INSERT*
- *Синтаксис*

```
INSERT [INTO] имя_таблицы [список_столбцов]
VALUES {DEFAULT список_значений|команда_select}
```

Действие

- *В окне Query Analyzer набрать:*

```
INSERT INTO RecruitmentAgencies
VALUES('0010','Head Hunters','223 Hill
Street','Cleveland', 'Ohio','44167-
5943','(440)345-8872','(440)345-
8943',7,1000)
```

- *Нажать F5 для выполнения запроса.*

Управление данными в таблицах

Сделать запрос в таблицу для проверки вставки данных

Действие

- ***В окне Query Analyzer набрать:***

SELECT * FROM RecruitmentAgencies

- ***Нажать F5 для выполнения запроса.***

Пример сохранение сведений из существующей таблицы в новую

Постановка задачи

Данные по внешним кандидатам с рейтингом восемь и выше копируются из таблицы ExternalCandidate в новую таблицу PreferredCandidate. Эта таблица еще не создана.

После ее создания с требуемыми правилами, необходимо добавить дополнительные строки по внешним кандидатам с рейтингом семь.

Управление данными в таблицах

Решение

1. Определить строки для вставки.

2. Создать новую таблицу с выбранными значениями.

3. Добавить строки в созданную таблицу.

4. Запросить таблицу для проверки добавления строк.

Определение строк для вставки.

Результат

- ***Из таблицы ExternalCandidate вставлены строки тех кандидатов, чей рейтинг восемь и выше.***

Создание новой таблицы с выбранными значениями

- *Команда SELECT INTO позволяет копировать содержимое одной таблицы в другую*
- *Синтаксис*

```
SELECT список_столбцов
INTO имя_новой_таблицы
FROM имя_таблицы
WHERE условие
```

Действие

- *В окне Query Analyzer набрать:*

```
SELECT * INTO PreferredCandidate
FROM ExternalCandidate
WHERE cRating >= 8
```

- *Нажать F5 для выполнения запроса.*

Управление данными в таблицах

Добавление строк в созданную таблицу

- *Команда INSERT INTO используется для добавления данных из одной таблицы в другую*
- *Синтаксис*

```
INSERT  [INTO] имя_таблицы1
SELECT имя(имена)_столбца
FROM имя_таблицы2
[WHERE условие]
```

Действие

- *В окне Query Analyzer набрать:*

```
INSERT INTO PreferredCandidate
SELECT * FROM ExternalCandidate
WHERE cRating = 7
```

- *Нажать F5 для выполнения запроса.*

Управление данными в таблицах

Запросить таблицу для проверки добавления столбцов

Действие

- *В окне Query Analyzer набрать:*

```
SELECT * FROM PreferredCandidate
```

- *Нажать F5 для выполнения запроса.*

Пример обновления таблицы

Постановка задачи

Результат испытаний внешнего кандидата Джей Шаффер (cCandidateCode 000049) повысился на два пункта в связи с обнаруженной ошибкой.

Управление данными в таблицах

Решение

1. *Определить атрибут, который будет обновлен.*
2. *Определить обновляемые значения.*
3. *Обновить строки.*
4. *Запрсить таблицы.*

Управление данными в таблицах

Определение атрибута, который будет обновлен.

Результат

- ***В таблице ExternalCandidate должен быть обновлен атрибут siTestScore.***

Определение обновляемых значений.

Результат

- ***Оценка испытания внешнего кандидата Jane Schaffer должна увеличится на 2 пункта. cCandidateCode для Jane – 000049.***

Управление данными в таблицах

Обновление строк.

- *Команда UPDATE используется для изменения данных в базе.*
- *Синтаксис*

```
UPDATE имя_таблицы
SET имя_столбца = значение [,имя_столбца = значение]
[FROM имя_таблицы]
[WHERE условие]
```

Действие

- *В окне Query Analyzer набрать:*

```
UPDATE ExternalCandidate
SET siTestScore=siTestScore+2
WHERE cCandidateCode='000049'
```

- *Нажать F5 для выполнения запроса.*

Управление данными в таблицах

Запрос таблиц.

Действие

- *В окне Query Analyzer набрать:*

```
SELECT * from ExternalCandidate
WHERE cCandidateCode = '000049'
```

- *Нажать F5 для выполнения запроса.*

Пример удаления данных

Постановка задачи

Таблица ExternalCandidate содержит данные по всем внешним кандидатам, были отвергнуты при вступительном испытании. Некоторым сведениям, присутствующим в таблице, уже более двух лет. Они занимают пространство на жестком диске, для которого нашлось бы иное применение. Эти данные уже не понадобятся. Необходимо гарантировать, что эти старые данные удалены из таблицы ExternalCandidate.

Управление данными в таблицах

Решение

1. *Определить строки, которые надо удалить.*
2. *Удалить строки(у).*
3. *Запросить таблицу.*

Управление данными в таблицах

Определение строк, которые надо удалить.

Результат

- ***Строки, содержащие данные о кандидатах, взявших вступительное испытание более чем два года назад.***

Управление данными в таблицах

Удаление строк(и).

- *Для удаления строк из таблици используется команда DELETE*
- *Синтаксис*

```
DELETE [FROM] имя_таблицы
[FROM таблиц(ы)]
[WHERE условие]
```

Действие

- *В окне Query Analyzer набрать:*

```
DELETE FROM ExternalCandidate
WHERE dTestDate < dateadd(yy,-2,getdate())
```

- *Нажать F5 для выполнения запроса.*

Управление данными в таблицах

Запросить таблицу.

Действие

- *В окне Query Analyzer набрать:*

```
SELECT * FROM ExternalCandidate
WHERE dTestDate < dateadd(yy,-2,getdate())
```

- *Нажать F5 для выполнения*

Сокращение таблицы

- Команда TRUNCATE TABLE
  - используется для удаления строк из таблицы
  - Идентична (по функциональности) команде DELETE
  - Работает быстрее команды DELETE
  - Не запускает действие
  - Синтаксис

```
TRUNCATE TABLE имя_таблицы
```

Выводы

Из этого урока Вы узнали, что:

- Правила и умолчания, это объекты, которые задаются для столбцов или пользовательских типов данных для обозначения значений ограничений и умолчаний соответственно
- Правило создается при помощи команды CREATE RULE, а при помощи процедуры *sp_bindrule* назначается для столбца или пользовательского типа данных.
- Правило снимается при помощи процедуры sp_unbindrule.
- Умолчание создается при помощи команды CREATE DEFAULT, а при помощи процедуры *sp_binddefault* назначается для столбца или пользовательского типа данных.
- Умолчание снимается при помощи процедуры sp_ unbindefault.
- Данные изменяются в базе для поддержания их в актуальном состоянии.
- Для вставки строк в таблицу используется команда INSERT.
- Командой SELECT INTO копируется содержимое одной таблицы в другую.

Выводы (продолжение)

- Команда SELECT с выражением INTO может применяться для сохранения результирующего множества в новую таблицу, без какого либо процесса описания данных.
- SQL Server предоставляет команду обновления строки UPDATE для изменения значений внутри таблиц.
- Удалить строку из таблицы можно при помощи команды DELETE.
- Для удаления всех строк из таблицы используйте команду TRUNCATE TABLE.