





## 2. Обработка ошибок в транзакциях

Если ошибка делает невозможным выполнение транзакции, SQL SERVER автоматически выполняет откат и освобождает ресурсы, удерживаемые транзакцией. Если к примеру сетевое соедин. клиента с сервером разорвано, выполнение отката всех несуб. транзакций для этого соедин. В случае ошибки во время выполнения инструкции сервер по умолчанию выполняет откат ~~всех~~ той инструкции которая привела к ошибке. Это поведение можно изменить таким образом, что в случае ошибки инструкции будет произведен откат только этой инструкции.

На случай возникновения ошибок код должен содержать исправляющее действие: COMMIT или ROLLBACK. Эффективным средством для обработки ошибок, включая ошибки транзакций, является конструкция TRY..CATCH.

## 3. R (A, B, C, D, E, F, G, H, I, J)

$$S = \{ABD \rightarrow E, AB \rightarrow G, B \rightarrow F, C \rightarrow J, CJ \rightarrow I, G \rightarrow H\}$$

Множество ФЗ явл. неприводимым тогда, когда выполняется след. св-ва:

1. Для любой ФЗ  $X \rightarrow Y$ , Y - один элемент
2. Ни одну ФЗ нельзя удалить без изменения замыкания.
3. Ни один атрибут не может быть удален из генер. замыкания без изм. замыкания



1. Выполняется

2.  $S = \{$
1.  $ABD \rightarrow E,$
  2.  $AB \rightarrow G,$
  3.  $B \rightarrow F,$
  4.  $C \rightarrow J,$
  5.  $CJ \rightarrow I,$
  6.  $G \rightarrow H \}$

$C \rightarrow J \Rightarrow C \rightarrow CJ$  транзитивно  $CJ \Rightarrow C \rightarrow I$ , значит

5 на  $C \rightarrow I$

1.  $ABD \rightarrow E$
2.  $AB \rightarrow G$
3.  $B \rightarrow F$
4.  $C \rightarrow J$
5.  $C \rightarrow I$
6.  $G \rightarrow H$

Также можем вывести  $CJ \rightarrow I$ : 5 и 4  $C \rightarrow I, J \Rightarrow CJ \rightarrow I, J \Rightarrow CJ \rightarrow I$

Т.к. мы смогли удалить атрибут из детерминанта без изм. замыкания, значит данное множество не является ключом.

Мне (3)