**HANDS-ON TRANSACTION & EXCEPTION HANDLING**

**—------------------------------------------------------------------------------------------------------------------**

**Tables: Features, Sales, Stores**

|  |
| --- |
| **Transaction:**  **A Transaction is an action which can be performed to make one or more changes to the database. The transaction can take DML commands (Insert, Update and Delete)** |

**1. Create a transaction to update your Feature table and save it.**

**// Transaction to update a record**

begin transaction Update\_Features

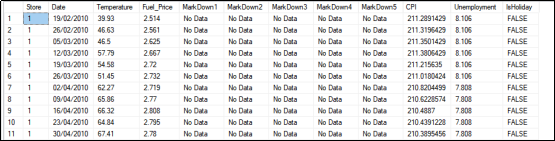
update Features

set MarkDown1='No Data',MarkDown2='No Data',MarkDown3='No Data',

MarkDown4='No Data', MarkDown5='No Data'

**// Commit Transaction (Saving the update)**

commit transaction Update\_Features



**2. Create a transaction to roll back the changes.**

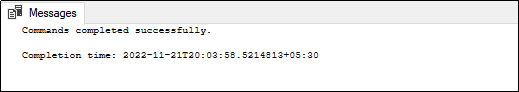
**// Transaction to Delete a record**

begin transaction Delete\_Stores

Delete Stores where Type='A'

**// Rollback Transaction (Undoing the deletion)**

rollback transaction Delete\_Stores



|  |
| --- |
| **SAVEPOINT:**           A Savepoint is like a marker. It saves a particular portion of a transaction without rolling the entire transaction. |

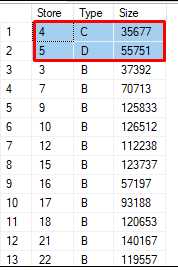
**4. Create a transaction to insert 2 records in the Stores table. Save the first record and roll back the last record.**

begin transaction Insert\_Stores

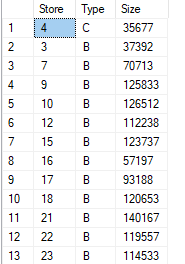
Insert into Stores values (4,'C',35677)

Save transaction First\_Record

Insert into Stores values (5,'D',55751)



rollback transaction First\_Record



|  |
| --- |
| **Exception Handling in Transaction:**  In the transaction, if the try blocks execute, the record will be automatically committed. If the catch blocks execute the record will be rolled back. |

**3. Create a transaction to divide the Weekly\_Sales column by zero. Use the exception-handling method.**

begin transaction Sales\_Division

begin try

Select (Weekly\_Sales)/0 from Sales

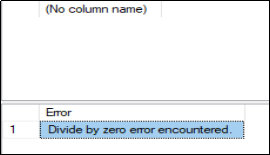
end try

begin catch

Select ERROR\_MESSAGE()as Error

end catch

**// The Transaction will be rolled back since we cannot divide Weekly\_Sales by 0.**



**4. Create a transaction to divide the Weekly\_Sales column by five. Use the exception-handling method.**

**begin transaction Sales\_Division**

**begin try**

**Select \*,(Weekly\_Sales)/5 as New\_Weekly\_Sales from Sales**

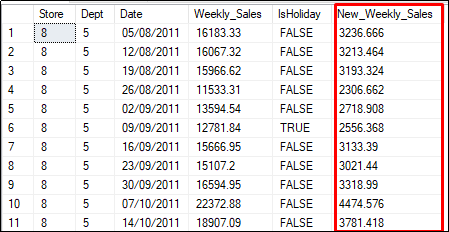
**end try**

**begin catch**

**Select ERROR\_MESSAGE()as Error**

**end catch**

**// The Transaction will be committed since we can divide a Weekly\_Sales by 5.**

****