**Umar Khan  
Purpose:**The purpose of the lab was to test different techniques of entering large sums of data into a database to observe the time taken by each technique to determine which technique serves us best in entering large data. Each technique have their own complexity.  
The Assignment extension was to do the same work with SQL Stored Procedures and see the difference and change regarding time taken and complexity.

**What We Have Done:**What I have done is that I have produced a table that correlates between different techniques and the time taken to enter the same amount of data in Milliseconds time, I have taken screenshots for each method as well as the source code has been attached with this document. There is one extra java class by the name of TableDropper that performs the renewal of the table to make sure the table is fresh when entering data into the database to make sure that the external factors like such don’t effect the time data.

|  |  |
| --- | --- |
| **Method** | **Time** |
| **AutoCommit = False** | |
| Statement(Non-Batched, Auto= False) | **1292 ms** |
| Statement(Batched, Auto= False) | **896 ms** |
| Prepared Statement(Non-Batched, Auto= False) | **948 ms** |
| Prepared Statement(Batched, Auto= False) | **274 ms** |
| Stored Procedure(AutoCommit = False) | **735 ms** |
| **AutoCommit = True** | |
| Statement(Non-Batched, Auto= True) | **1374 ms** |
| Statement(Batched, Auto= True) | **905 ms** |
| Prepared Statement(Non-Batched, Auto= True) | **1045 ms** |
| Prepared Statement(Batched, Auto= True) | **438 ms** |
| Stored Procedure(AutoCommit = True) | **898 ms** |

**Outcomes:**Prepared Statement in general is better than Statement while entering large sums of data but the complexity of coding in Prepared Statement is a bit higher than using the simple Statement, The variations in the AutoCommit and Batched are as such:  
1. AutoCommit = True and AutoCommit = False variation is such that it is a tradeoff between Performance, Flexibility and Safety but seeing in terms of performance AutoCommit = False is better than AutoCommit = True in Entering large sums of data.  
2. Batched Execution are better than Non-Batched Execution and the impact in my experience and my test runs were more than the AutoCommit’s Impact  
3. Batched coupled with Prepared Statements produce an enormous impact on time reduction in terms of Transaction for both AutoCommit = True and AutoCommit = False.

**What you should do:**1. AutoCommit should be set according to preference if you’re an experienced coder only then would I recommend AutoCommit = False Otherwise the default AutoCommit = True does not have that much of a performance impact and offers more safety.  
(Note: Remember if you’re setting AutoCommit = False, it’s a bad practice to Commit() all the executed statements at the end of all your code and will generally perform even worse than AutoCommit = True).  
2. For entering large sums of data it is better to do Batched Execution as it offers better performance.  
3. Prepared Statements are better than statements if you can handle the way they are used by Setting each element (I call this added complexity to code) and if they are coupled with Batched Execution they produce significant improvement in Performance.

**Code:  
Stored Procedure:**[CREATE](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/create-procedure.html) [PROCEDURE](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/create-procedure.html) data\_insert

([IN](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/comparison-operators.html#function_in) varName [varchar](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/string-types.html)(25), [IN](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/comparison-operators.html#function_in) varSemester [int](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/numeric-types.html), varAddress [varchar](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/string-types.html)(50))

BEGIN

[insert](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/insert.html) into Students(name, semester, address)

[values](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/miscellaneous-functions.html#function_values) (varName, varSemester,varAddress);

END