Project 1 Work Log					
Team Members:					
Cameron Hill					
Taylor Hunt					
Chris Langham					
		Date:	Description of Work:	Version #	Comments: (bugs to be fixed, problems, etc.)
Taylor, Chris, & Cameron		1/28/2014	Began work on design document.	1	
Taylor & Cameron		1/29/2014	More work on design document (mainly section 2).	1.1	
Taylor		1/30/2014	Added coding style document. Worked on application part of section 3 on design doc.	1.2	
Cameron		1/30/2014	Worked on section 3 relation part.	1.3	
Chris		1/30/2014	Worked on database part of design doc.	1.4	
Taylor		1/31/2014	Updated some stuff on the application part of section 3.	1.5	
Taylor		2/2/2014	Worked on section 4 of the design doc.	1.6	
Cameron		2/2/2014	Worked on section 3 parser and relation.	1.7	
Taylor		2/3/2014	Polished design doc and added some to section 3.	1.8	
Chris		2/3/2014	Worked on database part of design doc.	1.9	
Cameron		2/3/2014	Edited and polished through design doc.	2 (ready to submit)	
Cameron		2/4/2014	Added preliminary header documents - defined classes/enums	1-	
Cameron		2/6/2014	Designed and preliminary programming of conditions	1	Found need for pointers - most reference variables removed
Cameron		2/7/2014	Debugging of conditions - selection operation	1.1	
Cameron		2/8/2014	More polishing conditions. Also set union operation	1.2, 1.0	
Taylor		2/8/2014	Difference of sets, cross product, and a couple other functions	1.1	
Cameron		2/8/2014	Polished differences		For all relational algebra functions, you have to go through and set result.attrubtes to oldRelation.attributes or whatever. ALSO, the functions right now take strings, which is what I had changed it to, but they need to be able to take relations as well. The reason is that the database cell could also send it one of the local relations that the user creates. So sending a string means it is a relation held in the Database object. Sending a relation means it was in the DatabaseCell object (which right is from the test main)
Cameron/Chris/Taylor		2/10/2014	Sprint Session: Finishing up deliverable 2 - engine code w/ unit test		

Cameron/Chris		2/11/2014	Broke up parser into functions. Coded through main logic flow	1	
Cameron		2/13/2014	Implementation of basic parser functions	1.1	
Chris		2/13/2014	Work on setting up unit test for parser	1.1	
Taylor		2/13/2014	Standardization of unit test with Chris	1.11	
Taylor		2/14/2014	Worked on some parser functions and formatting. Completed union, difference, cross product, and delete from parser functions.	1.12	
Taylor		2/15/2014	More work on parser functions including insertInto	1.13	
Cameron		2/15/2014	Continued work on parser. Update to ConditionList	1.2, 2.0	
Cameron		2/16/2014	Finalizing work on parser to start thorough tests	1.3	
Taylor		2/16/2014	Lots of formatting in the parser class and worked on createTable for the parser	1.31	
Cameron		2/17/2014	Major debugging work	2	
Taylor		2/18/2014	Coded the framework for the application class.	2.1	
Taylor		2/19/2014	Did some work on the application	2.2	
Taylor		2/20/2014	Continued defining functions in the application class.	2.3	
ALL		2/20/2014	Work on defining problems in application and assigning work for last push	2.4	
Cameron		2/21/2014	Minor Debugging Work	2.42	
Chris		2/21/2014	Defined some of the update function applications	2.5	
Taylor		2/21/2014	Finished the remove functions in the application.	2.6	
Cameron		2/22/2014	Polished over parser. Improved functionality for final product	2.7	
Taylor		2/22/2014	Lots of debugging the application. Did some work on the menus and bad input handling.	2.8	
Taylor		2/23/2014	More debugging the application. Added the winningTeams relation and a few functions to go along with it. Fixed a couple bugs also.	2.9	
Taylor		2/24/2014	Made some updates to the design document. Added some functions to the application. Lots of formatting and commenting. Some overall project testing of the code	2.92	
Cameron		2/23/2014	Final Bug Fixxes. LOTS of debugging	2.93	
Cameron		2/24/2014	Report Formatting. BTHO Project 1?	2.94	
Chris		2/24/2014	Hardcore testing. Fixing logic of code. Try to create every possible scenario	3	