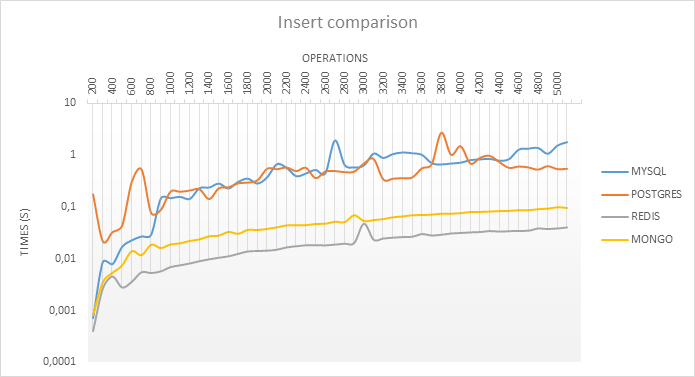
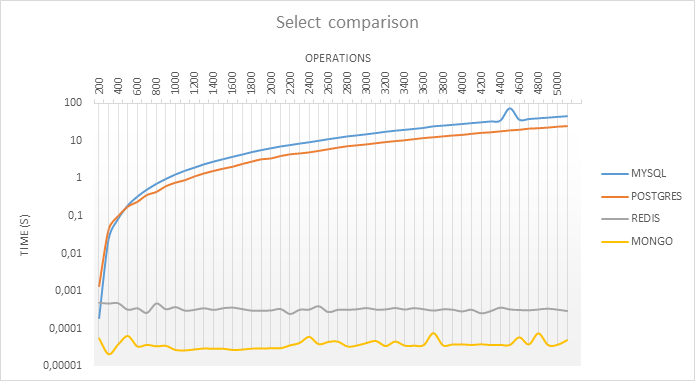
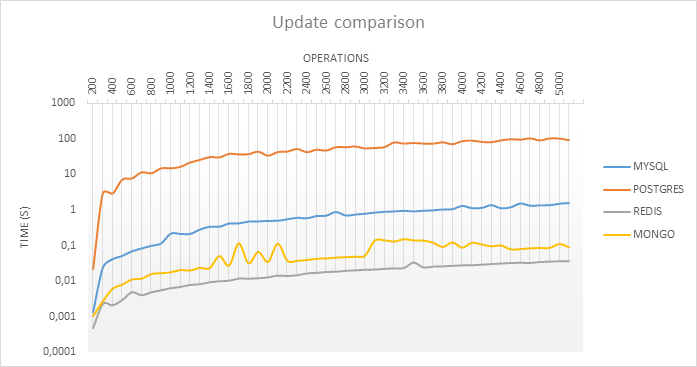
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | MySQL | Postgres | Redis | Mongo |
| Create | 3/4 | 3/4 | 1 | 2 |
| Read | 4 | 3 | 2 | 1 |
| Update | 3 | 4 | 1 | 2 |
| Delete | 4 | 3 | 1 | 2 |



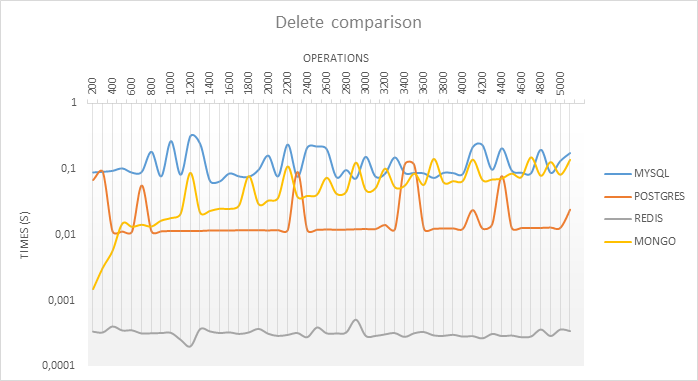
In the **create** category we can observe that Mongo and Redis are definitely better. At about 400–500 operations MySQL also looks good, but it decreases in performance after 600 records, with results similar to Postgres.



The **select** operation shows that SQL also has a bad time here. Performing it in MySQL and Postgres, time grows rapidly. For NoSQL it doesn’t matter how many records it needs to select, time is basically constant.



**update** is also much bigger for SQL databases, but the increase in time is similar for both databases. Time of operation for Redis and Mongo and MySQL is under 1 second for 5000 records. Only Postgres gives results over 10 seconds.



Here we have the biggest differences in results. The **delete** operation for Redis shows that having all records in memory allows for almost instant data removal. Time of removing records for SQL databases is stable. For Mongo we can see that the time of removal depends on the number of records.