

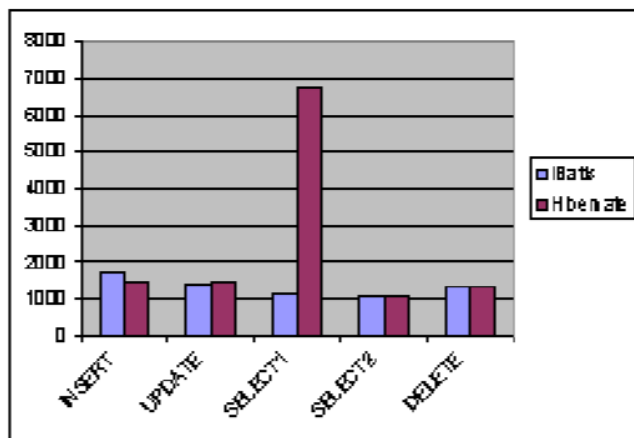
Testy i ich wyniki znalezione w sieci:

Zapis				Odczyt			
	JDBC	MyBatis	Hibernate		JDBC	MyBatis	Hibernate
próba 1	32 ms	87 ms	121 ms	próba 1	6 ms	11 ms	100 ms
próba 2	18 ms	24 ms	47 ms	próba 2	0 ms	3 ms	5 ms
próba 3	13 ms	15 ms	38 ms	próba 3	0 ms	2 ms	5 ms
próba 4	10 ms	16 ms	34 ms	próba 4	1 ms	2 ms	3 ms
próba 5	10 ms	11 ms	42 ms	próba 5	0 ms	1 ms	3 ms
Średnio:				Średnio:			
<ul style="list-style-type: none"> <li>MyBatis wydajniejsze od Hibernate o <b>46%</b></li> <li>JDBC wydajniejsze od Hibernate o <b>71%</b></li> <li>JDBC wydajniejsze od MyBatis o <b>46%</b></li> </ul>				<ul style="list-style-type: none"> <li>MyBatis wydajniejsze od Hibernate o <b>84%</b></li> <li>JDBC wydajniejsze od Hibernate o <b>94%</b></li> <li>JDBC wydajniejsze od MyBatis o <b>63%</b></li> </ul>			

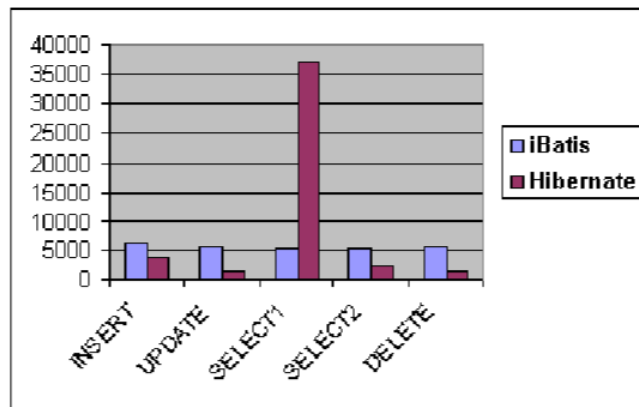
<https://bykowski.pl/mybatis-lekka-i-wydana-alternatywa-dla-hibernate/>

	A	B	C	D
1	function	session factory	insert (2 rows)	total
2	jdbc	181	48	229
3	mybatis	188	255	443
4	hibernate	764	137	901
5				
6				
7	function	session factory	select(58 rows)	total
8	jdbc	180	11	191
9	mybatis	187	226	413
10	hibernate	755	185	940
11	hibernate Sql Query	761	109	870
12			349 rows	
13	jdbc	180	26	206
14	mybatis	188	258	446
15	hibernate	757	226	983
16	hibernate Sql Query	762	126	888

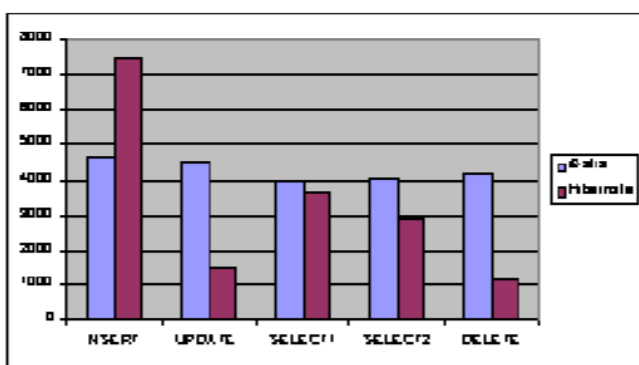
<https://www.programmersought.com/article/38921710283/>



**Figure 7: 1 Thread 5000 records 10 iterations**



**Figure 8: 5 Threads 5000 records 10 iterations**



**Figure 9: 50 Threads 5000 records 10 iterations**

<https://arxiv.org/ftp/arxiv/papers/0710/0710.1404.pdf>

Testy do przeprowadzenia:

- INSERT
- UPDATE
- SELECT (z określoną liczbą rekordów)
- SELECT (ze złączeniami)
- powtórny SELECT (testowanie pamięci cache)
- DELETE