

# OraBench - Release Notes

---

build passing release v1.0.0 release date june github repo or version not found

---

## Version 1.1.0

Release Date: xx.xx.2021

### New Features

- Database driver **Oracle.jl** and programming language **Julia**

### Applied Software

Software	Type	Version	Remark
C++ (gcc)	Language	10.3.0	upgrade
Elixir	Language	1.12.1-otp-24	
Erlang	Language	24.0.5	upgrade
Go	Language	1.17	upgrade
godror	Driver	v0.9.0	
Java	Language	openjdk-16.0.2	upgrade
Julia	Language	v1.6.2	new
Kotlin	Language	1.5.30	upgrade
Oracle cx_Oracle	Driver	8.2.1	
Oracle Instant Client	Driver	19.11.0.0.0	
Oracle JDBC	Driver	21.1.0.0	
Oracle ODPI-C	Driver	4.2.1	
Oracle.jl	Driver	v0.3.1	new
oranif	Driver	0.2.3	
Python 3	Language	3.9.6	upgrade

### Open issues

- Go & godror: (see [here](#))

---

## Windows 10 Performance Snapshot

The finishing touch to the work on a new release is a test run with all databases under identical conditions on three different systems - Ubuntu 20.04 via VMware and WSL2, Windows 10. The measured time includes the total time required for the DDL effort (database, schema, user, 5 database tables) and the DML effort (insertion of 7011 rows). The hardware used includes an AMD Ryzen 9 5950X CPU with 128GB RAM. The tests run exclusively on the computer in each case. The detailed results can be found in the DBSeeder repository in the [resources/statistics](#) directory.

The following table shows the results of the Windows 10 run. If the database can run with both activated and deactivated constraints (foreign, primary and unique key), the table shows the better value and in the column **Improvement** the relative value to the worse run. For example, the MonetDB database is faster with inactive constraints by 21.2% compared to the run with activated constraints.

- **DBMS** - official DBMS name
  - **Type** - client version, embedded version or via trino
  - **ms** - total time of DDL and DML operations in milliseconds
  - **Constraints** - DML operations with active or inactive constraints (foreign, primary and unique key)
  - **Improvment** - improvement of total time if constraints are inactive
- 

## Detailed Open Issues

### Go & godror

- Issue: dropping and restoring the same index - SQL statement `DROP INDEX constraint_kxn_2 CASCADE` (see [here](#)).