

## ora\_bench - Auxiliary Scripts.

### 1 run\_bench\_all\_drivers

This script executes the following driver specific sub-scripts:

- `src_c/scripts/run_bench_odpi`
- `src_elixir/scripts/run_bench_oranif`
- `src_erlang/scripts/run_bench_oranif`
- `src_go/scripts/run_bench_godror`
- `src_java/scripts/run_bench_jdbc`
- `src_kotlin/scripts/run_bench_jdbc`
- `src_python/scripts/run_bench_cx_oracle`
- `scripts/run_finalise_benchmark`

The possible exclusion of drivers made before is taken into account.

### 2 run\_collect\_and\_compile

Depending on the desired drivers, the script collects the libraries necessary for compilation and then compiles the ora\_bench software in the appropriate programming languages:

- C -- creation of the C specific property file -- cleaning existing binaries -- compilation with `make`
- Elixir -- creation of the Elixir specific property file -- providing `hex` -- cleaning existing dependencies -- load dependencies -- compilation with `mix`
- Erlang -- creation of the Erlang specific property file -- compilation with `rebar3`
- Go -- providing the `godror` driver
- Python -- creation of the Python specific property file

### 3 run\_create\_bulk\_file

This script is used to create a bulk file.

### 4 run\_db\_setup

This script downloads the Docker image with the requested database version from Docker Hub and prepares it for further processing:

- stops a currently running Docker container `ora_bench_db` and deletes it together with its image
- creates the Docker network `ora_bench_net`
- Docker image with the requested database version is downloaded from Docker Hub
- the new Docker container `ora_bench_db` is started as part of the network `ora_bench_net`
- the database schema `SCOTT` is set up in the database with the necessary rights and the corresponding database table

## 5 `run_finalise_benchmark`

In this script, OraBench.java is used to reset the following configuration parameters to the value 'n/a':

- `benchmark.comment`
- `benchmark.database`
- `benchmark.driver`
- `benchmark.host.name`
- `benchmark.id`
- `benchmark.language`
- `benchmark.number.cores`
- `benchmark.os`
- `benchmark.user.name`
- `connection.service`
- `sql.create`

## 6 `run_properties_standard`

This script is executed for one of the supported databases with standard properties. The script performs the following tasks:

- `run_collect_and_compile` - collects the libraries necessary for compilation and then compiles the ora\_bench software,
- `run_db_setup` - downloads the Docker image from Docker Hub and prepares its database for further processing and
- `run_bench_all_drivers` - executes the driver specific sub-scripts.

## 7 `run_properties_variations`

This script is executed for one of the supported databases. The script performs the following tasks:

- `run_collect_and_compile` - collects the libraries necessary for compilation and then compiles the ora\_bench software,
- `run_db_setup` - downloads the Docker image from Docker Hub and prepares its database for further processing and
- `run_bench_all_drivers` - is running for each of the following parameter combinations:

| <b>batch.size</b> | <b>core.multiplier</b> | <b>transaction.size</b> |
|-------------------|------------------------|-------------------------|
| default value     | default value          | default value           |
| default value     | 1                      | default value           |
| 0                 | default value          | default value           |
| 0                 | default value          | 0                       |
| 0                 | 1                      | default value           |
| 0                 | 1                      | 0                       |

## 8 `run_show_environment`

Logs the ora\_bench specific environment variables and the installed software versions.