Logging & Troubleshooting

Enabling Confluence logging

By default, PocketQuery is modest with logging in order to avoid flooding your log files. However, PocketQuery will become very talkative if you add some logging entries with level DEBUG at [Confluence Admin > Logging and Profiling]:

- `de.scandio.confluence.plugins.pocketquery`
- `de.scandio.atlassian.plugins.pocketquery`

Activate this, if you want to analyze the behaviour of PocketQuery. Information will be written to your `atlassian-confluence.log` file (if not configured otherwise in your instance).

Below some example log output:
run Running PocketQuery execution in of type [PocketQueryDefaultExecution]: queryName=SectionItems;
parameterString=(null)
ConfluencePocketQueryPlatformManager] isQueryAllowedInContext Checked if query SectionItems with spaces
attribute ' ' is allowed in current space PQ -- result: true
PocketQueryDefaultExecution] executeQueryAndTransformResult Executing query: queryName = SectionItems;
databaseName = PocketQuery; databaseUrl = jdbc:mysql://localhost:3306/pocketquery
PocketQueryDefaultExecution] mstart Start measuring execution time...
2019-10-22 08:36:11,471 DEBUG [http-nio-8090-exec-8] [managers.impl.delegates.SqlExternalDatabaseManager]
connectViaJdbc Trying to connect to JDBC database PocketQuery | jdbc:mysql://localhost:3306/pocketquery | com.
mysql.jdbc.Driver | root
2019-10-22 08:36:11,472 DEBUG [http-nio-8090-exec-8] [managers.impl.delegates.SqlExternalDatabaseManager]
loadDriver Trying to load database driver com.mysql.jdbc.Driver
2019-10-22 08:36:11,473 DEBUG [http-nio-8090-exec-8] [managers.impl.delegates.SqlExternalDatabaseManager]
loadDriver I could load the driver =)
PocketQueryDefaultExecution] executeQueryAndTransformResult Raw result retrieved: [{Section=Red, Item=Apples},
{Section=Red, Item=Cat}, {Section=Red, Item=Frog}, {Section=Blue, Item=Bike}, {Section=Blue, Item=Pig},
{Section=Green, Item=Slug}, {Section=Green, Item=Bird}]
PocketQueryDefaultExecution] mstart Start measuring execution time...
DelegatingExternalDatabaseManager] convert Nashorn engine detected on this Java platform. Using Nashorn
compiler engine.
executeConverter Loading JS libraries: {/pocketquery-core/lib/moment/moment.min.js}
executeConverter Loading: {/pocketquery-core/lib/moment/moment.min.js}
executeConverter Loading: {/pocketquery-core/core/js/nashorn/pocketquery-converter.js}