|  |  |  |
| --- | --- | --- |
|  | | |
| МИНОБРНАУКИ РОССИИ | | |
| Федеральное государственное бюджетное  образовательное учреждение высшего образования  «МИРЭА — Российский технологический университет»  РТУ МИРЭА | | |
|  | | |
| Институт информационных технологий (ИТ) | | |
| Кафедра инструментального и прикладного программного обеспечения (ИППО) | | |
| **ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №3** | | |
| **по дисциплине**  «Хранилища данных и OLAP-технологии» | | |
|  | | |
| Выполнил студент группы ИКБО-02-15 | | Апальков П.Ю. |
| Принял | | Абаев Р. Г. |
| Работа выполнена | « » 2018 г. | *(подпись студента)* |
| Работа зачтена | « » 2018 г. | *(подпись руководителя)* |
| Москва 2018 | | |

# Содержание

[Задание 3](#_Toc531543377)

[Выполнение работы 3](#_Toc531543378)

[Установка OLAP-инструментов 3](#_Toc531543379)

[Установка Pentaho Community Edition 5.2 4](#_Toc531543380)

[Настройка MySQL для работы с Pentaho 4](#_Toc531543381)

[Настройка JDBC Security. 4](#_Toc531543382)

[Настройка Hibernate and Quartz 6](#_Toc531543383)

[Отмена автоматического запуска базы данных Hypersonic 10](#_Toc531543384)

[Запуск Pentaho Community Edition. 11](#_Toc531543385)

[Настройка соединения 12](#_Toc531543386)

[Заключение 14](#_Toc531543387)

# Задание

Установить все необходимые OLAP-инструменты

# Выполнение работы

## Установка OLAP-инструментов

Pentaho CE написан на Java и для его работы необходимо установить виртуальную машину Java 7 Development Kit.

1. Скачать JDK 7 с официального сайта Oracle

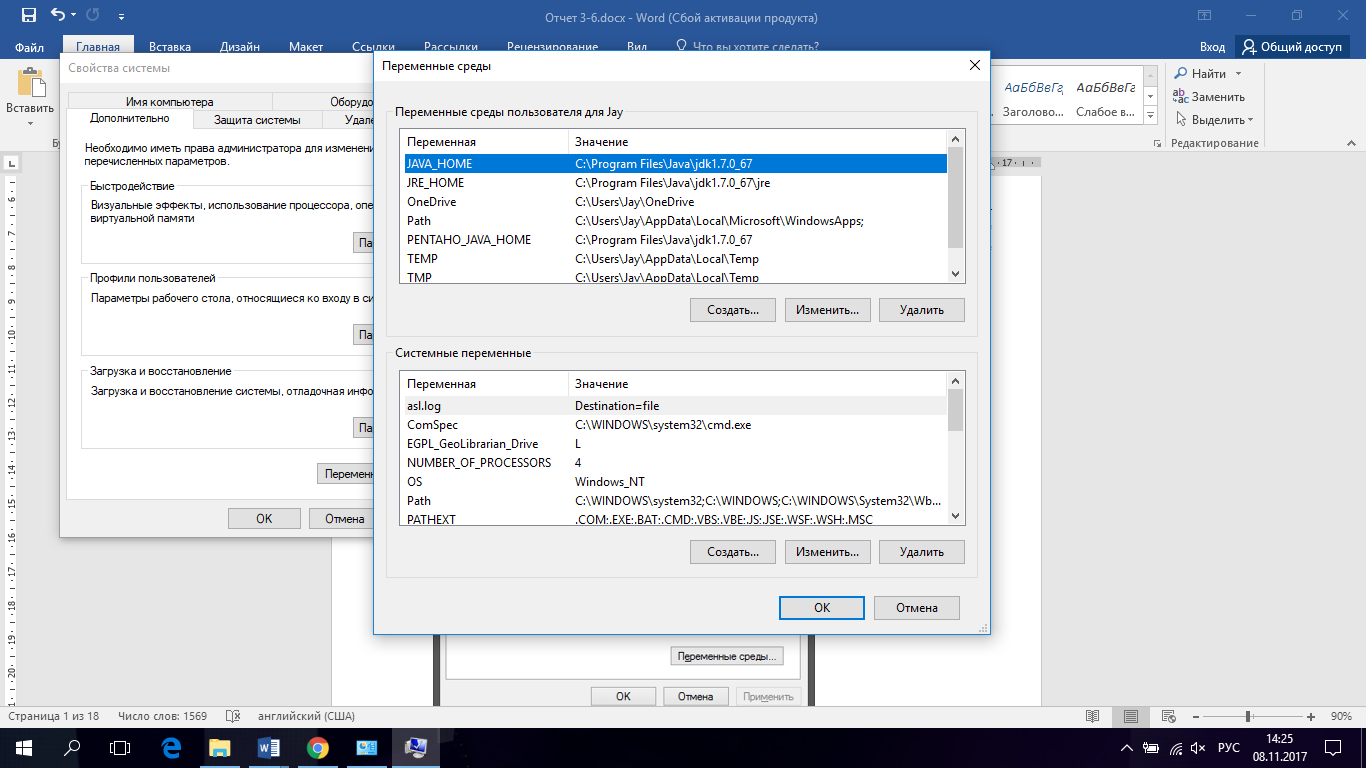
2. После установки необходимо установить системные переменные:

JAVA\_HOME = C:\Program Files (x86)\Java\jdk1.7.0\_71

JRE\_HOME = C:\Program Files (x86)\Java\jdk1.7.0\_71\jre

PENTAHO\_JAVA\_HOME = C:\Program Files (x86)\Java\jdk1.7.0\_71

Для установки системных переменных необходимо открыть свойства системы («Мой компьютер»-«Свойства») и выбрать опцию «Дополнительные параметры системы» и на вкладке «Дополнительно» открыть «Переменные среды» (Рис.1).

  
Рис.1 – Окно «Переменные среды»

## Установка Pentaho Community Edition 5.2

1. Скачать архив с Penataho с sourceforge

(<http://sourceforge.net/projects/pentaho/files/Business%20Intelligence%20Server/5.2/>)

2. Распаковать архив в удобную для работы директорию.

## Настройка MySQL для работы с Pentaho

Для корректной работы Pentaho необходимо создать следующие базы данных MySQL: *hibernate*, *quartz* и *jackrabbit* и пользователей *hibuser*, *pentaho\_user*, *jcr\_user* для этого нужно исполнить три .sql запроса, которые находятся в папке "...\biserver-ce\data\mysql5".

## Настройка JDBC Security.

Изменим файл «…pentaho-solutions\system\applicationContext-spring-security-hibernate.properties», оригинальный участок кода, который необходимо заменить представлен на листинге 1.

Листинг 1.

jdbc.driver=org.hsqldb.jdbcDriver

jdbc.url=jdbc:hsqldb:hsql://localhost:9001/hibernate

jdbc.username=hibuser

jdbc.password=password

hibernate.dialect=org.hibernate.dialect.HSQLDialect

Указанный выше текст следует заменить на текст из листинга 2.

Листинг 2.

jdbc.driver=com.mysql.jdbc.Driver

jdbc.url=jdbc:mysql://localhost:3306/hibernate

jdbc.username=hibuser

jdbc.password=password

hibernate.dialect=org.hibernate.dialect.MySQLDialect

Изменим файл «…pentaho-solutions\system\hibernate\hibernate-settings.xml», оригинальный участок кода, который необходимо заменить представлен на листинге 3.

Листинг 3.

<config-file>system/hibernate/hsql.hibernate.cfg.xml</config-file>

Указанный выше текст следует заменить на текст из листинга 4.

Листинг 4.

<config-file>system/hibernate/mysql5.hibernate.cfg.xml</config-file>

Изменим файл «…pentaho-solutions\system\quartz\quartz.properties».

Листинг 5.

org.quartz.jobStore.driverDelegateClass= org.quartz.impl.jdbcjobstore.PostgreSQLDelegate

Указанный выше текст следует заменить на текст из листинга 6.

Листинг 6.

org.quartz.jobStore.driverDelegateClass= org.quartz.impl.jdbcjobstore.StdJDBCDelegate

Изменим файл «…pentaho-solutions\system\applicationContext-spring-security-jdbc.xml», оригинальный участок кода, который необходимо заменить представлен на листинге 7.

Листинг 7.

# The fully qualified Java class name of the JDBC driver to be used

datasource.driver.classname=org.hsqldb.jdbcDriver

# The connection URL to be passed to our JDBC driver to establish a connection

datasource.url=jdbc:hsqldb:hsql://localhost:9002/userdb

# The connection username to be passed to our JDBC driver to establish a connection

datasource.username=sa

# The connection password to be passed to our JDBC driver to establish a connection

datasource.password=

# The SQL query that will be used to validate connections from this pool before returning them to the caller.

# This query must be an SELECT statement that returns at least one row.

# HSQLDB: SELECT 1 FROM INFORMATION\_SCHEMA.SYSTEM\_USERS

# MySQL, H2, MS-SQL, POSTGRESQL, SQLite: SELECT 1

# ORACLE: SELECT 1 FROM DUAL

datasource.validation.query=SELECT 1 FROM INFORMATION\_SCHEMA. SYSTEM\_USERS

Указанный выше текст следует заменить на текст из листинга 8.

Листинг 8.

datasource.driver.classname=com.mysql.jdbc.Driver

datasource.url=jdbc:mysql://localhost:3306/hibernate

datasource.username=hibuser

datasource.password=password

datasource.validation.query=SELECT 1

## Настройка Hibernate and Quartz

Далее необходимо настроить Hibernate and Quartz для того, чтобы использовать базы данных hibernate и quartz, созданные ранее. Для этого заменим в файле «…tomcat\webapps\pentaho\META-INF\context.xml» код из листинга 9 на код в листинге 10.

Листинг 9.

<Resource name="jdbc/Hibernate" auth="Container" type="javax.sql. DataSource"

factory="org.apache.commons.dbcp.BasicDataSourceFactory" maxActive="20" maxIdle="5"

maxWait="10000" username="hibuser" password="password"

driverClassName="org.hsqldb.jdbcDriver" url="jdbc:hsqldb:hsql://localhost/hibernate" validationQuery="select count(\*) from INFORMATION\_SCHEMA. SYSTEM\_SEQUENCES" />

<Resource name="jdbc/Quartz" auth="Container" type="javax.sql.DataSource"

factory="org.apache.commons.dbcp.BasicDataSourceFactory" maxActive="20" maxIdle="5"

maxWait="10000" username="pentaho\_user" password="password"

driverClassName="org.hsqldb.jdbcDriver" url="jdbc:hsqldb:hsql://localhost/quartz" validationQuery="select count(\*) from INFORMATION\_SCHEMA. SYSTEM\_SEQUENCES"/>

Листинг 10.

<Resource name="jdbc/Hibernate" auth="Container" type="javax.sql.DataSource"

factory="org.apache.commons.dbcp.BasicDataSourceFactory" maxActive="20" maxIdle="5"

maxWait="10000" username="hibuser" password="password"

driverClassName="com.mysql.jdbc.Driver" url="jdbc:mysql://localhost:3306/hibernate" validationQuery="select 1" />

<Resource name="jdbc/Quartz" auth="Container" type="javax.sql.DataSource"

factory="org.apache.commons.dbcp.BasicDataSourceFactory" maxActive="20" maxIdle="5"

maxWait="10000" username="pentaho\_user" password="password"

driverClassName="com.mysql.jdbc.Driver" url="jdbc:mysql://localhost:3306/quartz" validationQuery="select 1"/>

При этом надо убедиться, что был удален файл, представляющий копию данного: «…tomcat\conf\Catalina\localhost\pentaho.xml» или в него были внесены изменения.

В файле «…pentaho-solutions\system\jackrabbit\repository.xml» закомментируем исходный код из листинга 11 и раскомментируем код из листинга 12.

Листинг 11.

<FileSystem class="org.apache.jackrabbit.core.fs.local.LocalFileSystem">

<param name="path" value="${rep.home}/repository"/>

</FileSystem>

Листинг 12.

<FileSystem class="org.apache.jackrabbit.core.fs.db.DbFileSystem">

<param name="driver" value="com.mysql.jdbc.Driver"/>

<param name="url" value="jdbc:mysql://localhost:3306/jackrabbit"/>

<param name="user" value="jcr\_user"/>

<param name="password" value="password"/>

<param name="schema" value="mysql"/>

<param name="schemaObjectPrefix" value="fs\_repos\_"/>

</FileSystem>

Закомментируем исходный код части DataStore из листинга 13 и раскомментируем код листинга 14.

Листинг 13.

<DataStore class="org.apache.jackrabbit.core.data.FileDataStore"/>

Листинг 14.

<DataStore class="org.apache.jackrabbit.core.data.db.DbDataStore">

<param name="url" value="jdbc:mysql://localhost:3306/jackrabbit"/>

<param name="user" value="jcr\_user"/>

<param name="password" value="password"/>

<param name="databaseType" value="mysql"/>

<param name="driver" value="com.mysql.jdbc.Driver"/>

<param name="minRecordLength" value="1024"/>

<param name="maxConnections" value="3"/>

<param name="copyWhenReading" value="true"/>

<param name="tablePrefix" value=""/>

<param name="schemaObjectPrefix" value="ds\_repos\_"/>

</DataStore>

В части «FileSystem Workspace» закомментируем исходный код из листинга 15 и раскомментируем код листинга 16.

Листинг 15.

<FileSystem class="org.apache.jackrabbit.core.fs.local.LocalFileSystem">

<param name="path" value="${wsp.home}"/>

</FileSystem>

Листинг 16.

<FileSystem class="org.apache.jackrabbit.core.fs.db.DbFileSystem">

<param name="driver" value="com.mysql.jdbc.Driver"/>

<param name="url" value="jdbc:mysql://localhost:3306/jackrabbit"/>

<param name="user" value="jcr\_user"/>

<param name="password" value="password"/>

<param name="schema" value="mysql"/>

<param name="schemaObjectPrefix" value="fs\_ws\_"/>

</FileSystem>

В части «PersistenceManager» закомментируем код, представленный на листинге 17 и раскомментируем код из листинга 18.

Листинг 17.

<PersistenceManager class="org.apache.jackrabbit.core.persistence.pool.H2PersistenceManager">

<param name="url" value="jdbc:h2:${wsp.home}/db"/>

<param name="schemaObjectPrefix" value="${wsp.name}\_"/>

</PersistenceManager>

Листинг 18.

<PersistenceManager class="org.apache.jackrabbit.core.persistence.bundle.MySqlPersistenceManager"><param name="driver" value="com.mysql.jdbc.Driver"/>

<param name="url" value="jdbc:mysql://localhost:3306/jackrabbit"/>

<param name="user" value="jcr\_user" />

<param name="password" value="password" />

<param name="schema" value="mysql"/>

<param name="schemaObjectPrefix" value="${wsp.name}\_pm\_ws\_"/>

</PersistenceManager>

В части «FileSystem Versioning» закомментируем код, представленный в листинге 19, раскомментируем код из листинга 20.

Листинг 19.

<FileSystem class="org.apache.jackrabbit.core.fs.local.LocalFileSystem">

<param name="path" value="${rep.home}/version" />

</FileSystem>

Листинг 20.

<FileSystem class="org.apache.jackrabbit.core.fs.db.DbFileSystem">

<param name="driver" value="com.mysql.jdbc.Driver"/>

<param name="url" value="jdbc:mysql://localhost:3306/jackrabbit"/>

<param name="user" value="jcr\_user"/>

<param name="password" value="password"/>

<param name="schema" value="mysql"/>

<param name="schemaObjectPrefix" value="fs\_ver\_"/>

</FileSystem>

В части «PersistenceManager Versioning» закомментируем код, представленный в листинге 21 и раскомментируем код, представленный в листинге 22.

Листинг 21.

<PersistenceManager class="org.apache.jackrabbit.core.persistence.pool.H2PersistenceManager">

<param name="url" value="jdbc:h2:${rep.home}/version/db"/>

<param name="schemaObjectPrefix" value="version\_"/>

</PersistenceManager>

Листинг 22.

<PersistenceManager class="org.apache.jackrabbit.core.persistence.bundle.MySqlPersistenceManager"><param name="driver" value="com.mysql.jdbc.Driver"/>

<param name="url" value="jdbc:mysql://localhost:3306/jackrabbit"/>

<param name="user" value="jcr\_user" />

<param name="password" value="password" />

<param name="schema" value="mysql"/>

<param name="schemaObjectPrefix" value="pm\_ver\_"/>

</PersistenceManager>

## Отмена автоматического запуска базы данных Hypersonic

Для отмены автоматического запуска БД Hypersonic необходимо внести некоторые изменения в файл « … tomcat\webapps\pentaho\WEB-INF\web.xml»

Первая часть этого файла, которую необходимо удалить, представлена в листинге 23.

Листинг 23.

<!-- [BEGIN HSQLDB DATABASES] -->

<context-param>

<param-name>hsqldb-databases</param-name>

<param-value>sampledata@../../data/hsqldb/sampledata,

hibernate@../../data/hsqldb/hibernate,

quartz@../../data/hsqldb/quartz</param-value>

</context-param>

<!-- [END HSQLDB DATABASES] -->

Вторая часть этого файла, которую необходимо удалить, представлена в листинге 24.

Листинг 24.

<!-- [BEGIN HSQLDB STARTER] -->

<listener>

<listener-class>

org.pentaho.platform.web.http.context.HsqldbStartupListener

</listener-class>

</listener>

<!-- [END HSQLDB STARTER] -->

Далее необходимо обновить Java библиотеку для работы с MySQL. После этого файл «mysql-connector-java-5.1.30.jar» необходимо поместить в «…tomcat\lib» и удалить имеющуюся там устаревшую библиотеку «mysql-connector-java-5.1.17.jar»

## Запуск Pentaho Community Edition.

Перед непосредственным запуском Pentaho Community Edition необходимо запустить MySQL для работы с базой данных.

Для запуска веб-сервера Pentaho Community Edition необходимо выполнить скрипт «start-pentaho.bat» находящийся в корневой директории программы. После этого требуется подождать окончания процесса запуска, который может занять до нескольких минут.

Зайдем в браузер по адресу localhost:8080. Если сервер успел запуститься, то мы попадем на страницу логина (Рис.2). Нажимаем «Login as Evaluator» и затем на кнопку «Go» в разделе «Admin».

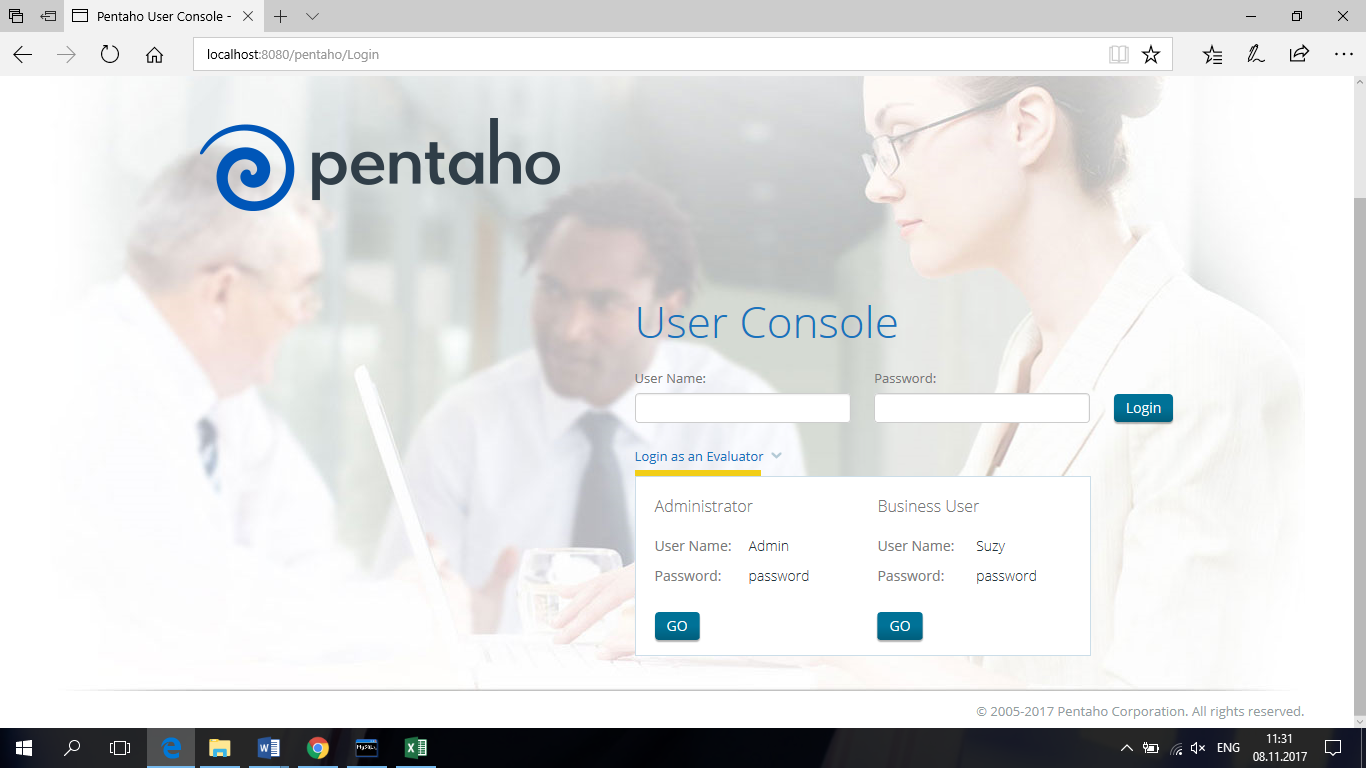


Рис. 2 – Страница логина Pentaho.

После авторизации вы попадете на контрольную панель Pentaho (Рис.3).



Рис.3. - Контрольная панель Pentaho

Для дальнейшей работы потребуется плагин Saiku Analytics. Для установки плагинов в Pentaho существует специальный Marketplace, для доступа к нему нужно нажать Home и во всплывающем меню выбрать Marketplace.

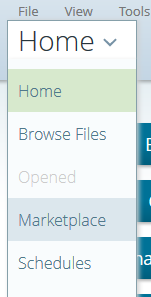


Рис. 4. - Меню Home.

После загрузки страницы Marketplace нажимаем в списке Saiku Analytics - Install. После успешной установки на экран выведется окно на Рис.5.

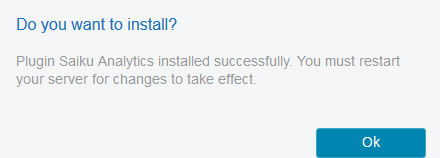


Рис. 5 – Окно установки.

## Настройка соединения

Для настройки соединения необходимо нажать «Manage Data Sources» в главном меню (Рис.6). Затем, в открывшемся окне нажимаем «New Data Source» (Рис. 7).

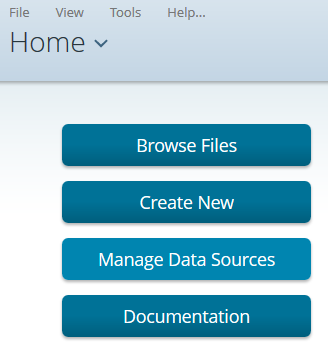


Рис.6. – Главное меню.

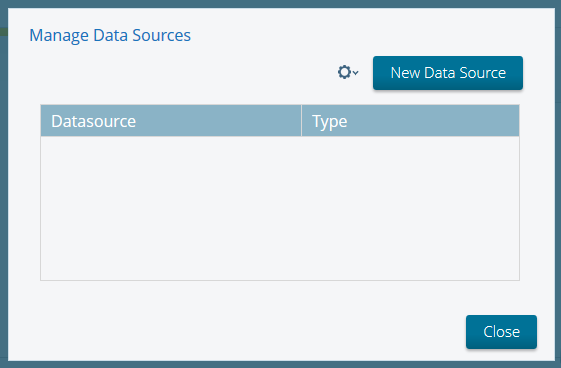


Рис. 7. – Окно Manage Data Sources

В поле «Data Source Name» выбираем имя, и в поле «Data Source Type» указываем «Database Table(s)» (Рис.8).

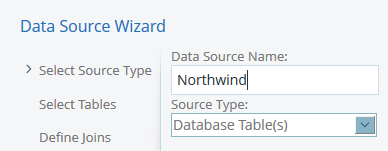


Рис. 8 – Пример заполнения полей в окне Data Source Wizard

После этого настройте новое соединение в окне Database Connection, вызвать которое можно, нажав “+” в открывшемся подразделе. Заполнить поля следует следующим образом (рис.9):

**Connection Name:** произвольное имя на латинице

**Database Type:** MySQL

**Host Name:** localhost

**Database Name:** название базы данных, в которой хранится хранилище **User Name:** root (или иное имя пользователя)

**Password:** ввести свой пароль.

Результат тестирования представлен на рис.10.

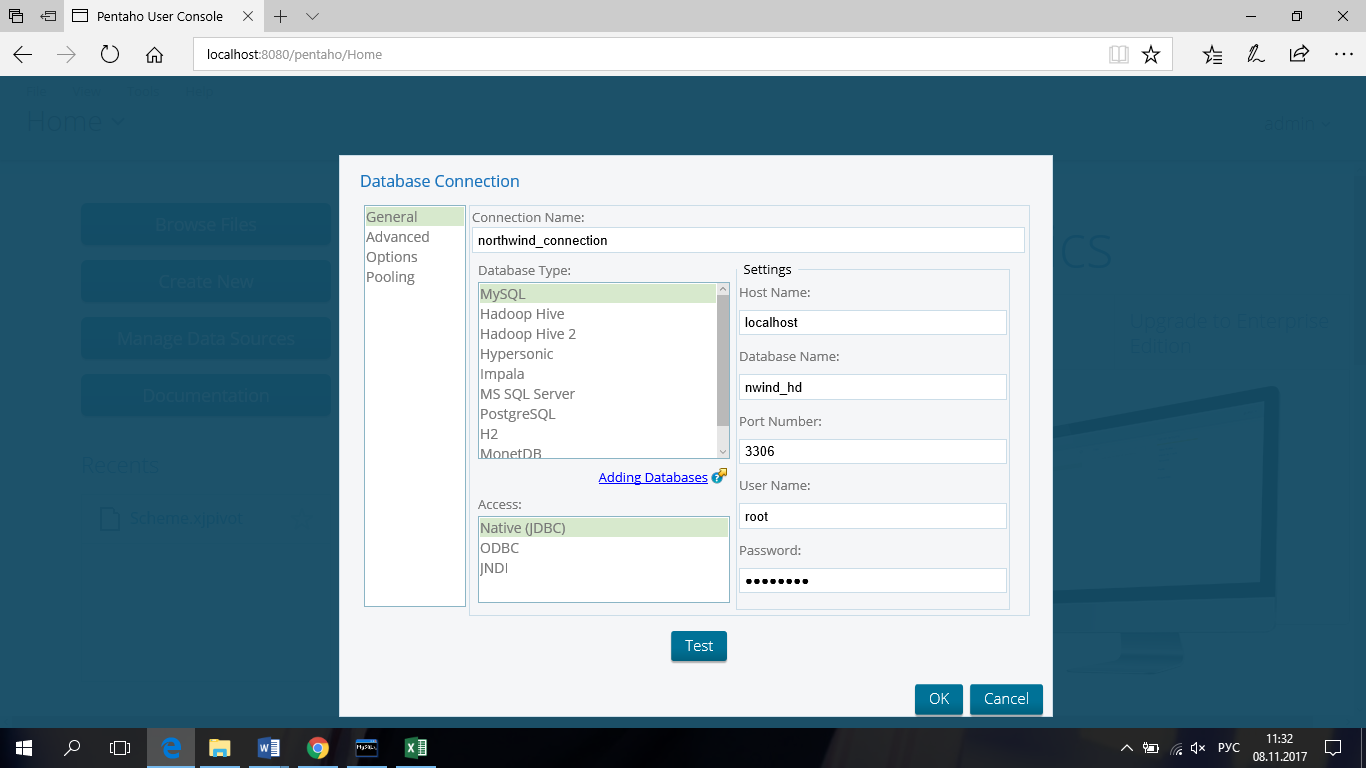


Рис. 9. – Пример заполнения полей в окне Database Connection

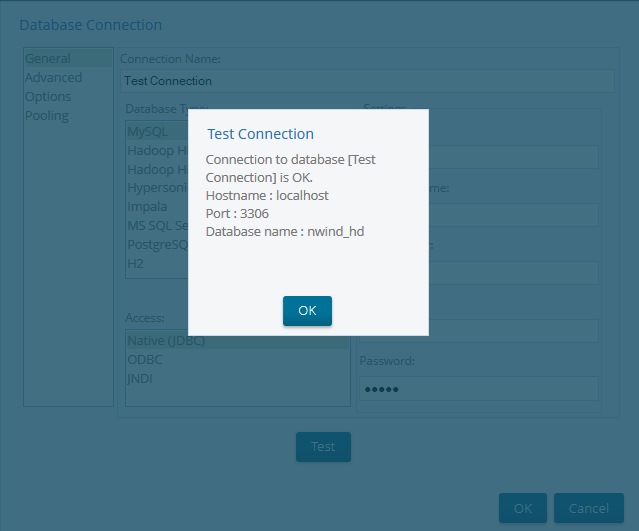


Рис. 10. – Тестирование соединения

После заполнения всех этих полей, нажимаем «ОК».

# Заключение

В ходе выполнения данной работы рассмотрен процесс инсталляции комплекса программ для работы OLAP хранилища, которые включают в себя Pentaho, MySQL и JDK7. Так же был рассмотрен процесс конфигурации данного набора программного обеспечения для их взаимодействия.