WORKSHOP SQL

Contents

[Project setup 2](#_Toc417658886)

[Data Definition Language 2](#_Toc417658887)

[Constraints 3](#_Toc417658888)

[Data Manipulation Language 4](#_Toc417658889)

[Retrieving Data from database 4](#_Toc417658890)

[Views 5](#_Toc417658891)

[Single row functions and group functions 5](#_Toc417658892)

[Liquibase generation DDLs 6](#_Toc417658893)

# Project setup

1. Open Intellij IDEA -> File -> New Project -> Maven Project -> Set the lines below -> Next -> Set project name and location -> Finish

*GroupId: ro.teamnet.zth*

*ArtifcatId: ZTH*

*Version: 1.0-SNAPSHOT*

1. Click on Database -> click on “+” icon -> Data Source -> Oracle and set the bellow data -> OK
   1. Host: 10.6.33.102
   2. Port: 1521
   3. Database: orcl
   4. User: xxx
   5. Password: xxx

# Data Definition Language

1. Create a table LOCATIONS with the following columns and data types:

LOCATION\_ID NUMBER PRIMARY KEY,

STREET\_ADDRESS VARCHAR2(40 BYTE),

POSTAL\_CODE VARCHAR2(12 BYTE),

CITY VARCHAR2(30 BYTE) NOT NULL,

STATE\_PROVINCE VARCHAR2(25 BYTE)

1. Create a table DEPARTMENTS with the following columns and data types:

DEPARTMENT\_ID NUMBER PRIMARY KEY,

DEPARTMENT\_NAME VARCHAR2(30 BYTE) NOT NULL,

LOCATION\_ID NUMBER

1. Create a table JOBS with the following columns and data types:

JOB\_ID NUMBER PRIMARY KEY,

JOB\_TITLE VARCHAR2(35 BYTE) NOT NULL,

MIN\_SALARY NUMBER(6),

MAX\_SALARY NUMBER(6)

1. Create a table EMPLOYEES with the following columns and data types:

EMPLOYEE\_ID NUMBER PRIMARY KEY,

FIRST\_NAME VARCHAR2(20 BYTE),

LAST\_NAME VARCHAR2(25 BYTE) NOT NULL,

EMAIL VARCHAR2(25 BYTE) NOT NULL,

PHONE\_NUMBER VARCHAR2(20 BYTE),

HIRE\_DATE DATE NOT NULL,

JOB\_ID VARCHAR2(10 BYTE) NOT NULL,

SALARY NUMBER(8,2),

COMMISSION\_PCT NUMBER(2,2),

MANAGER\_ID NUMBER,

DEPARTMENT\_ID NUMBER

1. Create 3 sequences like below:

CREATE SEQUENCE TAB\_DEPARTMENTS\_SEQ

START WITH 1

INCREMENT BY 1;

CREATE SEQUENCE TAB\_EMPLOYEES\_SEQ

START WITH 1

INCREMENT BY 1;

CREATE SEQUENCE ZTH\_SEQ

START WITH 406

INCREMENT BY 1;

# Constraints

1. Create the relations between the tables by alter the tables and add the FOREIGN KEYS:

FK\_EMPLOYEES\_DEPARTMENTS

FK\_EMPLOYEES\_JOBS

FK\_EMPLOYEES\_EMPL\_MANAGER

SYNTAX:

ALTER TABLE <TABLE1> ADD FOREIGN KEY (<FIELD\_TABLE1>)

REFERENCES <TABLE2> (<FIELD\_TABLE2>)

# Data Manipulation Language

1. Insert exercises:

Insert a new record in the table DEPARTMENTS:

insert into departments values

( TAB\_DEPARTMENTS\_SEQ.nextval,

'Administration',

1700

);

insert into jobs values

( 'AD\_PRES1'

, 'President'

, 20000

, 40000

);

INSERT INTO employees

VALUES (TAB\_EMPLOYEES\_SEQ.nextval,

'Steven',

'King',

'SKING',

'515.123.4567',

sysdate,

'AD\_PRES',

24000,

NULL,

NULL,

90);

1. Run a **ROLLBACK** command.
2. Insert exercises – Copy and Paste the content from file: *import\_values\_locations.sql* and *import\_values.sql* into a sql editor and run them all.
3. Run a **COMMIT** command.

# Retrieving Data from database

1. Play a little bit with the data – SELECT clauses
2. Return all employees
3. Return all departments
4. Return all jobs
5. Return First Name and Last Name for all employees
6. Return all Employees from department 50
7. Increase salary by 30% for all employees in department 50 (UPDATE Statement)
8. Remove the employee with EMPLOYEE\_ID 101 (DELETE Statement)
9. Return all Employees with job IT\_PROG, ordered by their First Name.
10. Same as 2, but also return the department name in the SELECT clause
11. Using an Alias

select count(employee\_id) from employees emp where emp.JOB\_ID = 'IT\_PROG'

1. Return all Employees from Seattle (location\_id = 1700);

# Views

1. Creating a View with employee\_id, first\_name and department\_name

# Single row functions and group functions

1. Run the following SQL: SELECT sysdate from dual;
2. Format sysdate with **TO\_CHAR** function.

select to\_char(sysdate, 'dd-MM-yyyy') from dual

1. Return a date from a varchar2 field

select to\_date ('25-11-2014', 'dd-MM-yyyy') from dual

1. Return all employees First Name, with Upper case and Email with Lower case
2. Return all employees First Name with the prefix “First Name: ” (use concat function). After this, do the same using || operator
3. Use the Count function: return the total number of employees.
4. Return the number of employees with job IT\_PROG
5. Return the number of employees for each department (count + group by)
6. Return the total salary for all employees in department 50 (use the SUM function)
7. Return max, min salary from employees in department 50.

# Liquibase generation DDLs

1. Get jdbc driver for Oracle from master branch in Java Repository
2. Open a command prompt, go in the directory where the ojdbc6.jar stands and run the following command:

mvn install:install-file -DgroupId=com.oracle -DartifactId=ojdbc6 -Dversion=11.2.0 -Dpackaging=jar -Dfile=ojdbc6.jar -DgeneratePom=true

1. Add the oracle driver dependency inside the pom.xml:

<dependency>

<groupId>com.oracle</groupId>

<artifactId>ojdbc6</artifactId>

<version>11.2.0</version>

</dependency>

1. Add liquibase plugin in pom.xml:

<plugins>

<plugin>

<groupId>org.liquibase</groupId>

<artifactId>liquibase-maven-plugin</artifactId>

<version>3.0.5</version>

<configuration>

<changeLogFile> src/main/resources/db/changelog/db.changelog-master.xml </changeLogFile>

<driver>oracle.jdbc.driver.OracleDriver</driver>

<url>jdbc:oracle:thin:@10.6.33.102:1521:orcl</url>

<username>username</username>

<password>password</password>

</configuration>

<executions>

<execution>

<phase>process-resources</phase>

<goals>

<goal>update</goal>

</goals>

</execution>

</executions>

</plugin>

</plugins>

1. Modify username and password tag from in the above plugin configuration
2. Create a *db* folder in resources; in *db* folder create a *changelog* folder
3. In *changelog* folder create 2 files: *db.changelog-master.xml and db.changelog-1.0.xml* with the following structure:

<?xml version="1.0" encoding="UTF-8"?>

<databaseChangeLog

xmlns="http://www.liquibase.org/xml/ns/dbchangelog/1.9"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://www.liquibase.org/xml/ns/dbchangelog/1.9

<http://www.liquibase.org/xml/ns/dbchangelog/dbchangelog-1.9.xsd>">

</databaseChangeLog>

1. In  *db.changelog-1.0.xml* create a changeSet for creating a table *PERSONS* with the following fields:

PERS\_ID NUMBER(6) PRIMARY KEY,

FIRST\_NAME VARCHAR2(40 BYTE),

LAST\_NAME VARCHAR2(40 BYTE),

EMAIL VARCHAR2(30 BYTE) NOT NULL,

PHONE\_NUMBER VARCHAR2(25 BYTE),

ADDRESS\_ID NUMBER(6) NOT NULL,

1. Include *db.changelog-1.0.xml* in *db.changelog-master.xml*
2. Run the compile task in Maven
3. Create *db.changelog-2.0.xml* in which you will create a changeset with ADDRESS table which will have the following fields:

ADR\_ID NUMBER PRIMARY KEY,

STREET VARCHAR2(50 BYTE),

NO NUMBER(6),

CITY VARCHAR2(30 BYTE) NOT NULL,

1. In same file *db.changelog-2.0.xml* create another changeset with a constaint for ADDRESS\_ID from PERSONS table
2. Include *db.changelog-2.0.xml* in *db.changelog-master.xml*
3. Run the compile task in Maven