Übung Datenbanken und Informationssysteme 1 2024w

Lecturer:
Josef Küng, Johannes Schrott, Wolfram Wöß

SQL Part 2 **Submission 05.12.2024, 13:00**

Technical Guidelines

The course information system is implemented in an Oracle database.

Oracle SQL Developer

The client software "Oracle SQL Developer" is currently available for Windows, macOS, and Linux. The "Oracle SQL Developer" (e.g., Windows 64-bit with JDK included) has to be downloaded from the Oracle website and installed on your PC. In Windows, after unzipping the downloaded file, you can immediately start "sqldeveloper.exe" without any further installation procedure. (https://www.oracle.com/tools/downloads/sqldev-downloads.html)

Create a new database connection (green +-symbol at the upper left side of the window) and connect to the database:
Name (connection): choose a name for the connection by your own
Benutzername (user): infosys
Kennwort (password): infosys
Hostname (host): infosys.faw.jku.at
Port: 1521
Service-Name: infosys

After you are successfully connected to the database, you can create and execute SQL statements in the "Query Builder" frame.

Course Information System

The JKU stores data about courses of the SS 2030 in an information system with the following four relations: course, lecturer, appointment, and room.

CourseId is structured as follows: the first 3 digits correspond to the institute number and the first 4 digits correspond to the department number. The institutes with the number 311, 312, and 321 comprise the entire area of "Computer Science". Course types are VO (Vorlesung / lecture), UE (Übung / exercise), SE (Seminar / seminar), and PR (Praktikum / practical course).

The relation "appointment" is based on the calendar day. Thus, for each appointment a course takes place, there is one entry in the table.

Relation name	Attribute	Type	Comment
Course	courseId	varchar2(6)	e.g., 351011
	title	varchar2(50)	course title
	hours	number(2)	weekly hours
	type	char(2)	VO, UE, SE, ...
Lecturer	lecturerId	varchar2(4)	
	name	varchar2(50)	
Room	roomId	varchar2(8)	
	name	varchar2(30)	room name
	capacity	number(4)	number of persons
	building	varchar2(20)	building name
Appointment	courseId	varchar2(6)	course number
	lecturerId	varchar2(4)	
	app_date	date	calendar day
	start_hour	number(2)	
	start_minute	number(2)	
	end_hour	number(2)	
	end_minute	number(2)	
	roomId	varchar2(8)	

Exercises

Create and execute the following SQL statements. You have to submit the SQL statement as well as the result set (output) including the number of rows in the result set. Please consider that the layout of the output should be easy to read (one line for one row).

6.9. Create a list of all lectures (courseId, title, weekly hours) held by the FAW institute (courseId begins with "3127") that have less than the average number of weekly hours of all courses. Sort the result set in ascending order by course title. (5 points)

6.10. Create a list that compares buildings by total number of rooms. The result should include buildings (name) and the total number of rooms in the building, in descending order by number of rooms. Limit the list to buildings with at least 9 rooms. (4 points)

6.11. Create a list of all courses (courseId, title, type, start time, end time, lecturer, room name, capacity) that are in progress on March 11th, 2030 at 10:30 in the "TNF-Turm" building and end at 11:45 at the latest. Lecturer "N. N." should not be included in the list. The courses should be sorted by capacity in descending order. (5 points)

6.12. Create a list of all "Computer Science" exercises (lecturerId, courseId, title, type, date, start time, end time, duration) with the longest (maximum) appointment duration in minutes. (6 points)

6.13. Create a list of the morning courses that start first (date, start time, end time, courseId, title, type, weekly hours) in room "T 911" for each day in the period 06-10 May, 2030 and sort the results by date. (5 points)

6.14. Create a list of all courses held by Wöß Wolfram starting at 13:45 or 15:30 with the corresponding number of appointments (courseId, title, start time, end time, number of appointments), ordered by start time. (4 points)

Create a view based on that query with the name "CStatistics". Attention: This operation cannot be executed in the Oracle database since you do not have the "create view" privilege. Destructive student(s) attacked the database server and therefore student privileges had to be limited to "select". (2 points)
Delete the view "CStatistics". (1 point)

6.15. Create a list of all course appointments held by "Wöß Wolfram" in March, 2030 including the lecturer's name, date, courseId, title, type, and roomId. The output format is defined in the following paragraph. (2 points)
Assume that your query result is persisted in the database as table "CourseAppointments" with the columns lecturer, date, courseNo, courseTitle, type, date, roomId.
CourseAppointments ({lecturer, date, course, title, type, roomId}, {courseId → title type}).

a) What is the primary key of table "CourseAppointments"? (1 point)
b) In which normal form is table "CourseAppointments"? (1 point)
c) Rename the course "Übung Informationssysteme 1" to "Übung Datenbanken und Informationssysteme 1". This operation cannot be executed in the Oracle database (see comments above in 6.14). (2 points)
d) What are the consequences of that update concerning anomalies and consistency of the entire course database. (2 points)