

SQL (STRUCTURED QUERY LANGUAGE)

1. Teil: SQL Foundation																										
Einleitung																										
DBMS	Database Management System																									
SQL-Datei	.sql																									
SQL-Programme	DB2 (IBM), Oracle, SQL Server (Microsoft), MySQL, SQLite, ...																									
Tabellen vs. Relationen	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: center;">Tabelle</td> <td style="text-align: center;">Spalten</td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;">Zeilen</td> <td></td> <td></td> <td></td> </tr> </table> <p style="text-align: center;">Dublikate erlaubt</p>				Tabelle	Spalten			Zeilen				<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: center;">Relation</td> <td style="text-align: center;">Attributen</td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;">Tupeln</td> <td></td> <td></td> <td></td> </tr> </table> <p style="text-align: center;">keine Duplikate -> PID</p>						Relation	Attributen			Tupeln			
Tabelle	Spalten																									
Zeilen																										
Relation	Attributen																									
Tupeln																										
Gross-/Kleinschreibung	nicht 'case-sensitiv'																									
Datentypen	BOOLEAN				Wahrheitswert																					
	SMALLINT, INTEGER, BIGINT				Exakte Numerische Typen																					
	DECIMAL				Fixkommazahlen																					
	REAL, DOUBLE, FLOAT				Gleitkommazahlen																					
	CHAR, VARCHAR, CLOB				Zeichenketten																					
	BLOB				Binary Large Object; binäre Zeichenkette																					
Operationen	DATE, TIME, INTERVAL				Datum ('yyyy-mm-dd'), Zeit, Zeitintervall																					
	+	-	*	/	Grundoperationen																					
	<	<=	=	>=	>	<>	Vergleiche																			
	%			\%			Wildcard - 0 bis beliebig viele Zeichen			Prozentzeichen																
	_						genau ein beliebiges Zeichen																			
	[]			[]^			zulässige Zeichenmengen			unzulässige Zeichenmengen																
boolsche Ausdrücke	and	or		not		logisch und		logisch oder		logisch nicht																
Aggregatfunktionen	count	min	max	sum	avg	anzahl	minimum	maximum	Summe	Durchschnitt																
Primärschlüssel (PID)	zur eindeutigen Identifizierung eines Tupels																									
Fremdschlüssel	Referenzen zu Primärschlüssel anderen Relationen																									
Schlüsselwörter und Reihenfolge	select Attributliste from Tabelle where Suchbedingung vor Gruppierung group by Attributliste having Suchbedingung nach Gruppierung order by Attributliste																									

Sprachkonstrukte der Data Manipulation Language (DML)

select	Allgemein select * from Personen select name, city from Personen Tupel erscheinen in beliebiger Reihenfolge! Liefert immer Tabellen (PID gilt nicht mehr)	name, city	Attributliste mit Komma trennen	
		*	* = alle Attribute	
		Personen	Relation	
	Dublikate entfernen select distinct Geburtstag from Personen		identische Tupel werden nur einmal angezeigt	
	Attributname umbenennen select Name as 'last name', Geburtstag as birthday from Personen	birthday	Neuer Name	
		'last name'	Hochkomma da auseinander geschrieben	
	Sortieren select * from Personen order by Name asc select * from Personen order by Name desc s...Personen order by Name desc, Name asc	asc desc ... , ...	ascending - aufsteigen descending - absteigend mehrfach sortieren	
	Filtern (Suchbedingung) s...Personen where Geburtstag > '2006-01-21' s...Personen where PID between 1 and 5	>, % and	Operatoren boolescher Ausdruck	
	Suchen nach Mustern s...Personen where Name like '%e%' s...Personen where Name not like '%e%'	'2006-01-21' 1, 5 '%e%'	Datentyp Alle die ein 'e' enthalten	
	Abfrage ob vorhanden s.. where Name in ('Peter', 'Anna') s.. where Name not in ('Peter', 'Anna')	not ... ('Peter', 'Anna')	nicht ... Liste mit Namen	
	Abgeleitete Spalten (Attributliste mit Ausdrücken) select PID * 100 as 'PID neu' from Personen	* 100	Operator Datentyp	
	Aggregationen (Gruppieren) select count(ID) as Anzahl from Artikel select count(ID) as Anzahl, min(Preis) as Minimum, sum(Preis) as Summe from Artikel	count(..), min(..), sum(..), avg(..)	Aggregatfunktion	
	Aggregationen nur auf Gruppen select Lieferant, count(ID) as Anzahl from Artikel group by Lieferant			
	mit Filter s..ikel group by Lieferant having count(ID) >=3	having	nach having muss eine Aggregatfunktion kommen	
	Tabellenverknüpfungen (Joins) select * from Personen inner join Bilder select Name, Aufnahmeort, BID from Personen inner join Bilder on Personen.PID = Bilder.PID s...Personen left join Bilder on ...	inner join on ... left join	jede Kombination mit Verknüpfung auch leere anzeigen	
	Unterabfrage (Subqueries, Subselect) where PID Operationen (select PID from Bilder) where PID [not] in (select PID from Bilder) where PID [not] exists (select PID from Bilder) where Operationen all (select PID from Bilder) where Operationen any (select PID from Bilder)	vergleich enthält existiert für alle für min. 1	subquery muss liefern: 1 Zeile, 1 Spalte x Zeilen, 1 Spalte Datensätze oder keine x Zeilen, 1 Spalte x Zeilen, 1 Spalte	
	als korrelierte Abfragen select outercolumns from outertable where outercolumnvalue in (select innercolumn from innertable where innercolumn = outercolumn)		Unterabfrage beinhaltet Attribute von äusserer Relation	
	insert	Zeilen einfügen insert into Personen (Name, PID, Geburtstag) values ('Johannes', 99, '2009-01-15')		eine Zeile einfügen
		insert into table (coll, col2, ...) select (vall, val2, val3, ...) from ...		mehrere Zeilen einfügen
	update	update table set column = expr where condition		werte einer Spalte ändern
update table set column = expression {, column = expression} [where searchcondition]			mehreren Spalten	
delete	delete from table		alle Tupel löschen	
	delete from table [where searchcondition]		löschen von Tupeln	

Sprachkonstrukte der Data Definition Language (DDL)

create	create table	Anlegen einer Relation (mit Primär- und Fremdschlüssel)
	create temporary table	Anlegen einer temporären Relation
alter	alter table	Ändern einer Relation
drop	drop table	Löschen einer Relation
grant		Rechte für Zugriff einstellen
Stored Procedures	create procedure Prozedurname (Parameterliste) SQL-Anweisungen	komplexe SQL Anweisungen kapseln

2. Teil: SQL Call-Level Interface

SQL und JAVA: JDBC (Java Database Connectivity)

1. Verbindung aufbauen	// Connection herstellen Connection con = getConnection(DBTyp, JDBC-URL, User, Password);	
2. Aufbau und Ausführung	// Statement herstellen, um ein Tabelle lesen zu können Statement s = con.createStatement(); // Ein Statement an die Datenbank schicken ResultSet rs = s.executeQuery("select * from title_authors");	
3. Zeilenweises Auslesen	// Das ResultSet kann nun ausgelesen werden. // In JDBC beginnt das erste Element mit 1. String title_id; String au_id; int au_order; double royalty_share; while (rs.next()) { title_id = rs.getString(1); au_id = rs.getString(2); au_order = rs.getInt(3); royalty_share = rs.getDouble(4); }	
4. Freigabe aller Ressourcen	// Die Ressourcen werden mit close wieder freigegeben rs.close(); s.close(); con.close();	
	<pre> static Connection getConnection(int type, String jdbcurl, String user, String pw) { //Zuerst muss der Treiber ueber Reflection geladen werden Class c; switch (type) { case 0: //SQL-Server c = Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver"); break; case 1: //MySQL c = Class.forName("com.mysql.jdbc.Driver"); break; case 2: //JDBC-ODBC Bridge c = Class.forName("sun.jdbc.odbc.JdbcOdbcDriver"); break; } //nun wird ein Datenbank-spezifisches Connection-Objekt erzeugt. return DriverManager.getConnection(jdbcurl, user, pw); } </pre>	
Verwendete Klassen	Connection	für das Verbinden mit SQL-Datenbank
	Statement	für das Lesen von SQL-Relation
	PreparedStatement	für das Vorkompilieren und Parametrieren von SQL-Anweisungen
	ResultSet	für SQL-Anweisungen
	ResultSetMetaData	für Meta-Daten (Spalten, Datentypen, ...)
Verwendetet Methoden	Statement.execute()	SQL-Anweisung schicken (no return-value)
	Statement.executeUpdate()	return value = Anzahl geänderter Einträge