

a)

1. Datenbank-Modelle:

a) Nennen Sie vier Datenbank-Modelle.

- hierarchisch (+ sehr schnelle Abfragen; + schneller Zugriff auf zusammengehörende Daten; - nur Baumstruktur, keine anderen Verknüpfungen)
- objektorientiert
- NoSQL (+ Rechenleistung kann einfach verteilt werden; - Redundanz der Daten)

b) Welches Modell steht hinter MongoDB?

NoSQL

2. Index:

a) Erklären Sie, was ein Index ist.

Eine sortierte Ansammlung von Zeigern auf eine (oder mehrere) Attribute einer Tabelle.

b) Wofür wird ein Index eingesetzt?

Um die Suche und das Sortieren von gewissen Attributen zu beschleunigen. JOIN-Abfragen können auch beschleunigt werden.

c) Was sind die Nachteile eines Indexes?

- Es benötigt zusätzlichen Speicherplatz
- INSERT, UPDATE und DELETE werden langsamer

d) Erstellen Sie für die Tabelle *tblTest*(ID, Nachname, Vorname, Gebdat...) einen Index auf den Nachnamen.`CREATE INDEX i_nachname ON tblTest(Nachname);`e) Lassen Sie sich alle Indizes der Tabelle *tblTest* anzeigen.`SHOW INDEX FROM tblTest;`

f) Löschen Sie den in Aufgabe d erstellten Index wieder.

`DROP INDEX i_nachname ON tblTest;`

g) Die Abfrage

`select Vorname from tblTest where Nachname = 'xxx'`

(xxx ist variabel) ist sehr häufig, aber langsam. Setzen Sie einen sinnvollen Index.

`CREATE INDEX i_nachname ON tblTest(Nachname);`**3. Benutzer und Rechte:**a) Erstellen Sie den Benutzer *fmuster*, der nur auf demselben Computer auf MySQL zugreifen kann, auf dem die Datenbank auch installiert ist. Das Passwort soll 'GIBM2018' sein.`CREATE USER 'fmuster'@'localhost' IDENTIFIED BY 'GIBM2018';`*Beachten Sie, dass der Benutzer fmuster noch keine Rechte hat.*b) Geben Sie dem Benutzer *fmuster* das Recht, Daten der Tabelle *tblIMA* in der Datenbank *dbPersonal* anzusehen, einzufügen und zu ändern.`GRANT SELECT, INSERT, UPDATE ON dbPersonal.tblIMA TO 'fmuster'@'localhost';`

- c) Erstellen Sie den Benutzer *hansmueller* (Passwort *qwert*), der von jedem beliebigen Computer auf die Datenbank zugreifen darf. Geben Sie dem Benutzer gleichzeitig die vollen Rechte für die Datenbank *db_test*.
`CREATE USER 'hansmueller'@'%' IDENTIFIED BY 'qwert';`
`GRANT ALL PRIVILEGES ON db_test.* TO 'hansmueller'@'%';`
- d) Legen Sie einen Benutzer *admin* (Passwort *secure_pwd_7*) an, der nur lokal auf die Datenbank zugreifen darf. Geben Sie dem Benutzer die vollen Rechte für alle DBs. Der Benutzer darf auch andern Benutzern seine Rechte (oder einen Teil davon) weitergeben.
`CREATE USER 'admin'@'localhost' IDENTIFIED BY 'secure_pwd_7';`
`GRANT ALL ON *.* TO 'admin'@'localhost' WITH GRANT OPTION;`
- e) Geben Sie dem obigen Benutzer *admin* noch die Möglichkeit, vom Computer mit der IP-Adresse 192.168.1.17 zuzugreifen. Die Rechte sollen gleich sein wie oben beschrieben.
`CREATE USER 'admin'@'192.168.1.17' IDENTIFIED BY 'secure_pwd_7';`
`GRANT ALL ON *.* TO 'admin'@'192.168.1.17' WITH GRANT OPTION;`
Beachten Sie, dass ein neuer Benutzer mit demselben Namen, aber einem anderen Host angelegt werden muss!
- f) Lassen Sie sich alle existierenden Benutzer anzeigen.
`SELECT * FROM mysql.user;`
(interessant für die Fragestellung sind die Attribute *User* und *Host*)
- g) Lassen Sie sich alle Berechtigungen des Netzwerk-Benutzers *testuser* anzeigen.
`SHOW GRANTS FOR 'testuser'@'%';`
- h) Entziehen Sie dem vorher erstellten Benutzer *fmuster* das Recht, Daten der Tabelle *tblMA* in der Datenbank *dbPersonal* zu ändern.
`REVOKE UPDATE ON dbPersonal.tblMA FROM 'fmuster'@'localhost';`
- Löschen Sie den Benutzer *fmuster*, der vom lokalen Computer auf die Datenbank zugegriffen hat.
`DROP USER 'fmuster'@'localhost';`

4. Transaktionen:

- a) Was ist eine Transaktion?
Eine Folge von Arbeitsschritten, die als logische Einheit betrachtet wird. Es werden entweder alle Arbeitsschritte ausgeführt oder keiner.
- b) Wie lautet der Befehl um eine Transaktion zu starten?
`START TRANSACTION`
- c) Was bewirkt ein COMMIT und was ein ROLLBACK am Ende einer Transaktion?
`COMMIT`: die gemachten Änderungen werden dauerhaft gespeichert
`ROLLBACK`: die gemachten Änderungen werden verworfen