

WIEDERHOLUNG

TRANSAKTION

ACID

TRANSAKTIONSVERWALTUNG

- KONSISTENZSICHERUNG
- RECOVERY
- MEHRBENUTZERSYNCHRONISATION

ANOMALIEN BEI UNKONTROLLIERTEM MEHRBENUTZERBETRIEB

LOST UPDATE

Zeit	Transaktion 1	Transaktion 2

1	Lese Betrag	-
2	-	Lese Betrag
3	Erhöhe Betrag um 1	-
4	-	Erhöhe Betrag um 1
5	Commit	-
6	-	Commit

UNCOMMITTED DEPENDENCY (DIRTY READ)

Zeit	Transaktion 1	Transaktion 2

1	Lese Betrag	-
2	Erhöhe Betrag um 1	-
3	-	Lese Betrag
4	Rollback	-

INCONSISTENCY READ (NONREPEATABLE READ)

Zeit	Transaktion 1	Transaktion 2

1	Lese Haben_Konto	-
2	-	Ändere Soll_Konto
3	-	Ändere Haben_Konto
4	-	Commit
5	Lese Soll_Konto	-
6	Commit	-

PHANTOMDATEN

Zeit	Transaktion 1	Transaktion 2

1	Lese Kontostand	-
2	-	Ändere Kontostand
4	-	Commit
5	Lese Kontostand	-
6	Commit	-

LOCKS

READ LOCKS (SHARED)

WRITE LOCKS (EXCLUSIVE)

ZWEI-PHASEN- SPERRPROTOKOLL

LOS GEHT'S

\o/

CREATE VIEW

```
CREATE VIEW  
    SELECT - Anweisung;
```

```
REPLACE VIEW  
    SELECT - Anweisung;
```

```
CREATE OR REPLACE VIEW view_name AS SELECT - Anweisung;
```

BEISPIEL:

```
CREATE VIEW PERSONAL_IT1 AS  
  SELECT *  
  FROM PERSONAL WHERE GEH_STUFE='it1';
```

```
SELECT *  
FROM PERSONAL_IT1;
```

SCHACHTELN VON VIEWS:

```
CREATE VIEW PERSONAL_IT1_AOK AS  
SELECT *  
FROM PERSONAL_IT1  
WHERE KRANKENKASSE = 'aok';
```

SPALTEN UMBENENNEN:

```
CREATE VIEW BOHRMASCHINEN (NR, NWERT, ZWERT)
AS
SELECT MNR, NEUWERT, ZEITWERT
FROM MASCHINE
WHERE NAME='Bohrmaschine';
```

```
SELECT *
FROM BOHRMASCHINEN WHERE ZWERT > 16000;
```

```
DROP VIEW view_name;
```

```
DROP VIEW PERSONAL_IT1;
```

ES WERDEN KEINE DATEN IN RELATIONEN GELOSCHT.

AUFGABEN

1. ERSTELLEN SIE EINEN VIEW FÜR ALLE MITARBEITER, DIE IN EINER ABTEILUNG ARBEITEN, DIE MIT 'P' ANFÄNGT
2. ERSTELLEN SIE EINEN VIEW FÜR MITARBEITER UND DIE SUMME IHRER PRÄMIEN.
3. ERSTELLEN SIE EINEN VIEW FÜR MITARBEITER, DIE KINDER HABEN. UND DANN EINE VIEW FÜR MITARBEITER, DIE MINDESTENS ZWEI KINDER HABEN.

NICHT BEZOGEN AUF VIEWS:

- I. FÜGEN SIE JEDEM MITARBEITER EINE PRAMIE VON 500.- HINZU, DER EIN KIND HAT.
 - IA. FÜGEN SIE JEDEM MITARBEITER EINE PRÄMIE VON 500.- PRO KIND HINZU.

JDBC

AUFGABEN

- DATENBANKVERBINDUNGEN AUFBAUEN UND VERWALTEN.
- SQL-ANFRAGEN AN DIE DATENBANK WEITERLEITEN UND DIE
- ERGEBNISSE IN EINE FÜR JAVA NUTZBARE FORM UMZUWANDELN

JDBC IST TEIL DES JDK

`java.sql` **UND** `javax.sql`

JEDE DATENBANK HAT EIGENE TREIBER

```
package java.sql;
```

```
/**  
 * The interface that every driver class must implement.  
 *  
 * ...  
 */  
public interface Driver {  
    ...  
}
```

```
public class oracle.jdbc.driver.OracleDriver implements java.sql.Driver {  
    ...  
}
```

```
package java.sql;
```

```
/**
```

```
 * <P>The basic service for managing a set of JDBC drivers.<br>
```

```
 *
```

```
 * ...
```

```
 *
```

```
 */
```

```
public class DriverManager {
```

```
final Connection connection = DriverManager.getConnection(  
    "jdbc:oracle:thin:@ora14.informatik.haw-hamburg.de:1521:inf14", username, password);
```


java.sql.Connection

```
connection.setAutoCommit(autoCommit);
```

```
connection.commit();
```

```
connection.rollback();
```

```
connection.setReadOnly(readOnly);
```

```
connection.setTransactionIsolation(level);
```

```
connection.setSavepoint(name)  
connection.rollback(savepoint);  
connection.releaseSavepoint(savepoint);
```

```
Statement createStatement();
```


java.sql.Statement

```
connection.createStatement().executeQuery(sql);
```

```
connection.createStatement().executeUpdate(sql)
```

JDO

```
<table name="PERSONAL">
  <column name="PNR" type="INTEGER" required="true" primaryKey="true"
    autoIncrement="true"/>
  <column name="VORNAME" type="VARCHAR" size="50" required="true"/>
  <column name="NACHNAME" type="VARCHAR" size="50" required="true"/>
  <column name="GEH_STUFE" type="VARCHAR" size="5"/>
  <column name="ABT_NR" type="VARCHAR" size="5"/>
  <column name="KRANKENKASSE" type="VARCHAR" size="3"/>
  <foreign-key foreignTable="GEHALT">
    <reference local="GEH_STUFE" foreign="GEH_STUFE"/>
  </foreign-key>
  <foreign-key foreignTable="ABTEILUNG">
    <reference local="ABT_NR" foreign="ABT_NR"/>
  </foreign-key>
</table>
```

CODE-BEISPIEL

ORACLE JDBC-TREIBER

[HTTP://WWW.ORACLE.COM/TECHNETWORK/DATABASE/
ENTERPRISE-EDITION/JDBC-112010-090769.HTML](http://www.oracle.com/technetwork/database/enterprise-edition/jdbc-112010-090769.html)

AUFGABE 1

**FÜGEN SIE DIE RESTLICHEN FELDER DER KLASSE `Personal`
HINZU UND ERGÄNZEN SIE DIE `toString` - FUNKTION**

AUFGABE 2

ERSTELLEN SIE DIE STRUKTUR FÜR PERSONAL ANALOG FÜR DIE
TABELLE MASCHINE:
– LEGEN SIE EINE NEUE KLASSE AN
– GEBEN SIE ALLE MASCHINEN MIT ALLEN ATTRIBUTEN AUF DER
KOMMANDOZEILE AUS

AUFGABE 3

FUGEN SIE EINE NEUE ABTEILUNG HINZU

AUFGABE 4

FUGEN SIE EINE NEUE MASCHINE UND EINEN NEUEN MITARBEITER HINZU. BAUEN SIE DIESE FUNKTION SO, DASS WENN EIN INSERT FEHLSCHLÄGT, KEINE DER BEIDEN EINTRÄGE IN DER DATENBANK STEHEN (SIEHE `rollback`)

NÄCHSTE VORLESUNG

7.1.2016

A stylized, flat-design illustration of Rainbow Dash from the animated series My Little Pony: Friendship is Magic. She is depicted in a dynamic, flying pose, angled towards the right. Her body is a light blue color, and her wings are spread wide. Her most prominent feature is her long, flowing mane and tail, which are composed of multiple overlapping bands of color in the rainbow spectrum: red, orange, yellow, green, blue, and purple. Her eyes are large and expressive, with black outlines and pinkish-purple irises. The background is a solid, dark grey-blue color. Overlaid on the image is the German text 'DAS WAR'S FÜR HEUTE' in a large, white, sans-serif font. The text is split across two lines: 'DAS WAR'S FÜR' on the top line and 'HEUTE' on the bottom line. The letters are bold and have a slight shadow, making them stand out against the background and the character.

DAS WAR'S FÜR
HEUTE