

# Aufgaben-Sequenz zur LWBAdventure-Datenbank

## RELATIONEN

NPCs (\$ NPCNr, NPCName) -- NPC: Non Playing Character

Dozent\_innen (! NPCNr, Lieblingsgetränk)

sonstigeNPCs (! NPCNr, Aufgabe)

Veranstaltungen (\$ VNr, VName, Kürzel, SWS, GebietNr, ! NPCNr, ! RaumNr)

Themengebiete (\$ GebietNr, GebietName)

Minigames (\$ GameNr, Game-Name, ! VNr)

Spieler\_innen (\$ SpNr, SpName, Schlüsselanzahl, ! RaumNr)

Räume (\$ RaumNr, Raumname, Ort, Funktion)

Unterricht (! VNr, ! NPCNr, ! RaumNr)

Spielstände (! GameNr, ! SpNr, Punkte, Note)

Standorte (! NPCNr, ! RaumNr)

Assistenz (! NPCNr, ! VeranstaltungsNr)

## ANFRAGEN

**1a. Welche Räume gibt es in der LWB-Adventure-World?**

Raeume

```
SELECT * FROM raeume;
```

**1b. Welche Aufgaben haben die sonstigen NPCs im LWB-Adventure?**

$\pi_{\text{Aufgabe}}(\text{sonstigeNPCs})$

```
SELECT aufgabe FROM sonstigenpcs;
```

**2a. Welche Lehrveranstaltungen haben 6 SWS?**

$\sigma_{\text{SWS} = 6}(\text{Veranstaltungen})$

```
SELECT * FROM veranstaltungen WHERE semester = 4;
```

**2b. Welche Lehrveranstaltungen gibt es im 4. Semester?**

$\sigma_{\text{RaumNr} = 4}(\text{Veranstaltungen})$

```
SELECT * FROM veranstaltungen WHERE raumnr = 4;
```

**2c. Welche Minigames gibt es im 4. Semester?**

$\sigma_{\text{RaumNr} = 4}(\text{Minigames} \bowtie \text{Veranstaltungen})$

```
SELECT * FROM minigames NATURAL JOIN veranstaltungen WHERE raumnr = 4;
```

**3a. Wie heißen die Spieler\_innen, die bisher das LWB-Adventure gespielt haben?**

$\pi_{\text{SpName}}(\text{Spieler\_innen} \bowtie \text{NPCs})$

SELECT spname FROM spieler\_innen NATURAL JOIN npcs;

**3b. Wie heißen die Dozenten im LWB-Adventure?**

$\pi_{\text{NPCName}}(\text{Dozent\_innen} \bowtie \text{NPCs})$

SELECT npcname FROM dozent\_innen NATURAL JOIN npcs;

**3c. Welche Aufgabe hat 'Heidi'?**

$\pi_{\text{Aufgabe}}(\text{sonstigeNPCs} \bowtie \text{NPCs})$

SELECT aufgabe FROM sonstigenpcs NATURAL JOIN npcs WHERE npcname = 'Heidi';

**4a. Welche Lehrveranstaltungen haben etwas mit 'Daten' (/ 'Programmierung') zu tun?**

$\sigma_{\text{VName}} = \text{'*Daten*'} (\text{Veranstaltungen})$

SELECT \* FROM veranstaltungen NATURAL JOIN themengebiete WHERE vname LIKE '%Daten%';

$\sigma_{\text{GebietName}} = \text{'*Daten*'} (\text{Veranstaltungen} \bowtie \text{Themengebiete})$

SELECT \* FROM veranstaltungen NATURAL JOIN themengebiete WHERE gebietname LIKE '%Daten%';

**4b. Welche Lehrveranstaltungen gehören zum Themengebiet 'Programmierung'?**

$\sigma_{\text{GebietName}} = \text{'Programmierung'} (\text{Veranstaltungen} \bowtie \text{Themengebiet})$

SELECT \* FROM veranstaltungen NATURAL JOIN themengebiete WHERE gebietname = 'Programmierung';

**5. Was ist das Lieblingsgetränk von Darth Schmidter?**

$\pi_{\text{Lieblingsgetränk}}(\sigma_{\text{NPCName}} = \text{'Darth Schmidter'} (\text{Dozent\_innen} \bowtie \text{NPCs}))$

SELECT Lieblingsgetraenk FROM dozent\_innen NATURAL JOIN npcs WHERE npcname = 'Darth Schmidter';

**6. Welche Lehrveranstaltungen finden nicht in der FU statt?**

$\pi_{\text{Vname, Ort}}((\text{Räume} \setminus \sigma_{\text{Ort}} = \text{'*FU*'} (\text{Räume})) \bowtie \text{Unterricht} \bowtie \text{Veranstaltungen})$

SELECT vname, raumnr AS semester, ort FROM raeume NATURAL JOIN unterricht NATURAL JOIN veranstaltungen WHERE ort NOT LIKE '%FU%';

**7. Welche Dozenten sind in der LWB nur leitend tätig und machen keine Assistenz?**

$\pi_{\text{NPCName}}(\text{NPCs} \bowtie (\pi_{\text{NPCNr}}(\text{Dozent\_innen}) \setminus \pi_{\text{NPCNr}}(\text{Assistenz})))$

SELECT npcname FROM npcs NATURAL JOIN (SELECT npcnr FROM dozent\_innen EXCEPT SELECT npcnr FROM assistenz) AS <alias>;

**Anfragen, die nur mit erweiterter relationaler Algebra beschrieben werden können:**

**8. Wieviele Mini-Games gibt es in der LWB-Adventure-World? (Ausgaben-Titel: AnzahlMiniGames)**

$\gamma_{\text{AnzahlMiniGames}} \leftarrow \text{COUNT}(*)(\text{Minigames})$

`SELECT COUNT(*) AS AnzahlMiniGames FROM mini-games;`

**9. Wieviele SWS müssen in der LWB insgesamt absolviert werden? (Ausgaben-Titel: GesamtanzahlSWS)**

$\rho_{\text{GesamtanzahlSWS}} \leftarrow \text{SUM}(\text{sWS})(\gamma_{\text{SUM}(\text{sWS})}(\text{Veranstaltungen}))$

$\gamma_{\text{GesamtanzahlSWS}} \leftarrow \text{SUM}(\text{sWS})(\text{Veranstaltungen})$

`SELECT SUM(sws) AS GesamtanzahlSWS FROM veranstaltungen;`

**10. Wie heißt die Veranstaltung mit den meisten SWS?**

$\pi_{\text{Vname}}(\sigma_{\text{sWS} = \text{MAX}(\text{sWS})}(\gamma_{\text{MAX}(\text{sWS})}(\text{Veranstaltungen})))$

`SELECT vname FROM veranstaltungen WHERE sws = (SELECT MAX(sws) FROM veranstaltungen);`

**11. Gesucht sind die Namen, Semester (entspricht RaumNr!) und SWS aller Veranstaltungen von Winnie the K absteigend sortiert nach SWS-Anzahl!**

$\tau_{\text{sWS}} \downarrow (\pi_{\text{Vname}, \text{sWS}, \text{Semester}}(\rho_{\text{Semester} \leftarrow \text{Raumnr}}(\sigma_{\text{NPCName} = \text{'Winnie the K'}}(\text{Veranstaltungen} \bowtie \text{Unterricht} \bowtie \text{NPCs}))))$

`SELECT vname, sws, raumnr AS semester FROM veranstaltungen NATURAL JOIN unterricht NATURAL JOIN npc WHERE npcname = 'Winnie the K' ORDER BY sws DESC;`

**12. Wieviele Veranstaltungen gibt es pro Standort?**

$\tau_{\text{AnzahlVeranstaltungen}}(\gamma_{\text{Ort}, \text{AnzahlVeranstaltungen}} \leftarrow \text{COUNT}(*)(\text{Räume} \bowtie \text{Unterricht}))$

`SELECT ort, COUNT(*) AS AnzahlVeranstaltungen FROM raeume NATURAL JOIN unterricht GROUP BY ort ORDER BY COUNT(*);`

**13. Welche Spieler\_innen haben einen Gesamt-Notendurchschnitt, der nicht zwischen 2.0 und 4.0 liegt? (Sortierung nach Gesamt-Notendurchschnitt absteigend)**

$\tau_{\text{Schnitt}}(\text{Spieler\_innen} \bowtie (\sigma_{\text{Schnitt} < 2.0 \wedge \text{Schnitt} > 4.0}(\gamma_{\text{Schnitt} \leftarrow \text{AVG}(\text{Note})}(\text{Spielstand}))))$

$Q1 \leftarrow \gamma_{\text{Schnitt} \leftarrow \text{AVG}(\text{Note})}(\text{Spielstand})$

$Q2 \leftarrow \sigma_{\text{Schnitt} < 2.0 \wedge \text{Schnitt} > 4.0}(Q1)$

$Q3 \leftarrow \tau_{\text{Schnitt}}(\text{Spieler\_innen} \bowtie Q2)$

`SELECT * FROM spieler_innen NATURAL JOIN spielstand WHERE note = (SELECT AVG(note) FROM spielstand WHERE AVG(note) NOT BETWEEN 2.0 AND 4.0);`