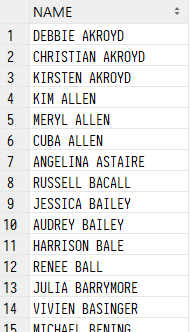
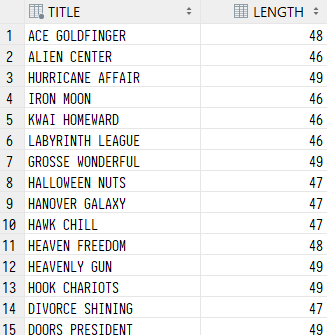
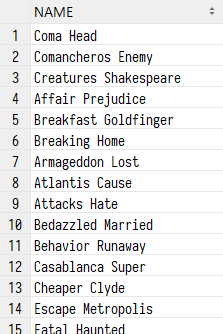
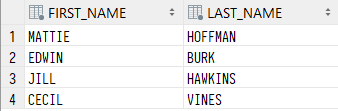
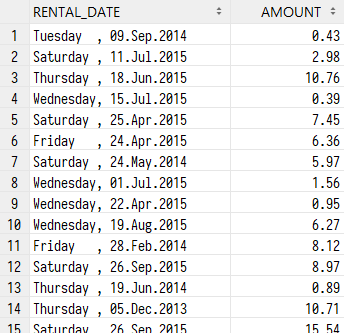
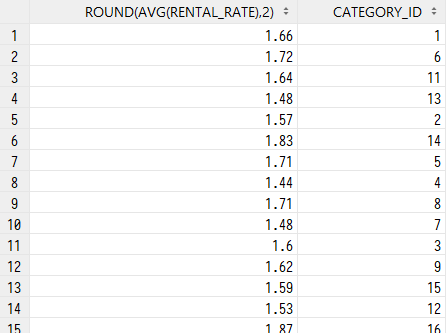
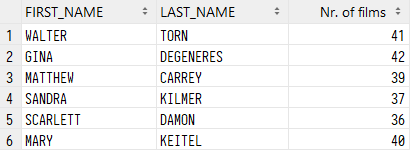
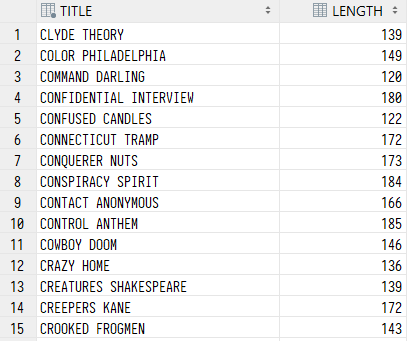
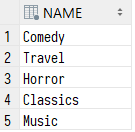
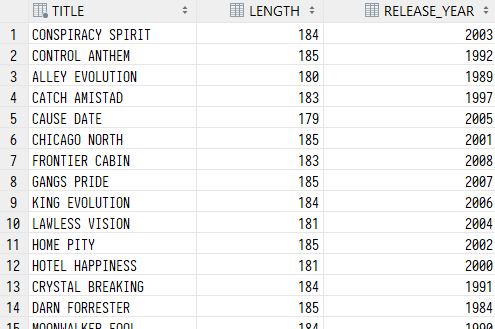
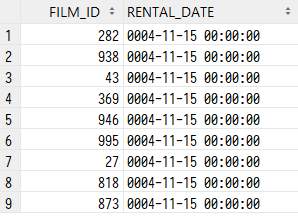
Ausarbeitung Übung 02

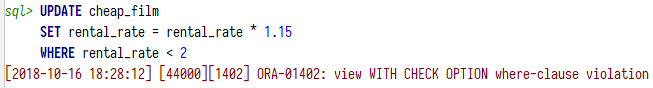
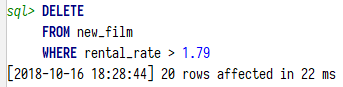
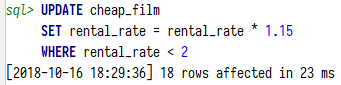
# Grundlagen

*-- 1.1***SELECT first\_name** || **' '** || **last\_name AS** "Name"  
**FROM** actor  
**ORDER BY last\_name ASC**;  
  
*-- 1.2***SELECT  
 title**,  
 **length  
FROM** film  
**WHERE length** < 50;  
  
*-- 1.3***SELECT** *INITCAP*(**title**) **AS** "Name"  
**FROM** film  
**WHERE title LIKE '\_\_\_A%'**;  
  
*-- 1.4***SELECT title  
FROM** film  
**WHERE original\_language\_id IS NOT NULL**;  
  
*-- 1.5***SELECT** *COUNT*(*\**) **AS** "Nr. of rented films"  
**FROM** rental  
**WHERE rental\_date** < *TO\_DATE*(**'31.12.2015'**, **'DD.MM.YYYY'**) **AND  
 rental\_date** > *TO\_DATE*(**'01.01.2015'**, **'DD.MM.YYYY'**);  
  
*-- 1.6***SELECT inventory\_id  
FROM** inventory  
**MINUS  
SELECT inventory\_id  
FROM** rental;  
  
*-- 1.7***SELECT  
 first\_name**,  
 **last\_name  
FROM** customer  
 **INNER JOIN** address **USING** (**address\_id**)  
 **INNER JOIN** city **USING** (**city\_id**)  
**WHERE city IN** (**'Newcastle'**, **'Linz'**, **'London'**);  
  
*-- 1.8***SELECT** *to\_char*(**rental\_date**, **'Day, DD.Mon.YYYY'**) **AS** rental\_date,  
 **amount  
FROM** payment p  
 **INNER JOIN** rental **r USING** (**rental\_id**)  
**WHERE** p.**customer\_id** = 420;  


# Gruppierungen und Unterabfragen

*-- 2.1***SELECT** *ROUND*(*AVG*(**rental\_rate**), 2),  
 **category\_id  
FROM** film  
 **INNER JOIN** film\_category **USING** (**film\_id**)  
**GROUP BY category\_id**;  
  
*-- 2.2***SELECT title  
FROM** film  
**WHERE length** > (**SELECT length  
 FROM** film  
 **WHERE film\_id** = 50) **AND  
 replacement\_cost** > (**SELECT replacement\_cost  
 FROM** film  
 **WHERE film\_id** = 101);  
  
*-- 2.3***SELECT title  
FROM** film  
 **INNER JOIN** film\_category **USING** (**film\_id**)  
**WHERE length** < 60 **AND  
 category\_id IN** (**SELECT category\_id  
 FROM** film  
 **INNER JOIN** film\_category **USING** (**film\_id**)  
 **WHERE film\_id IN** (10, 20, 30));  
  
*-- 2.4***SELECT  
 first\_name**,  
 **last\_name**, *Count*(*\**) **AS** "Nr. of films"  
**FROM** actor  
 **INNER JOIN** film\_actor **USING** (**actor\_id**)  
**WHERE actor\_id IN** (**SELECT actor\_id  
 FROM** film\_actor  
 **GROUP BY actor\_id  
 HAVING** *COUNT*(*\**) > 35)  
**GROUP BY actor\_id**, **first\_name**, **last\_name**;  
  
*-- 2.5***SELECT  
 title**,  
 **length  
FROM** film  
**WHERE length** > (**SELECT** *AVG*(**length**)  
 **FROM** film);  
  
*-- 2.6***SELECT UNIQUE name  
FROM** film\_category  
 **INNER JOIN category USING** (**category\_id**)  
**WHERE category\_id IN** (**SELECT category\_id  
 FROM** film\_category  
 **GROUP BY category\_id  
 HAVING** *COUNT*(**film\_id**) < 60);  
  
*-- 2.7***SELECT  
 title**,  
 **length**,  
 **release\_year  
FROM** film f1  
**WHERE length** >= **ALL** (**SELECT length  
 FROM** film f2  
 **WHERE** f2.**release\_year** = f1.**release\_year**);  
  
*-- 2.8***SELECT  
 film\_id**,  
 *to\_date*(**rental\_date**, **'YYYY-MM-DD'**) **AS** rental\_date  
**FROM** (**SELECT** *\** **FROM** rental  
 **INNER JOIN** inventory **USING** (**inventory\_id**)  
 **INNER JOIN** film **USING** (**film\_id**)  
 **ORDER BY rental\_date DESC**)  
**FETCH FIRST** 9 **ROWS ONLY**;  


# Insert, Update und Delete *-- 3.1* **CREATE TABLE** new\_film **AS** (**SELECT** *\** **FROM** film **WHERE release\_year** >= (**SELECT** *MAX*(**release\_year**) **FROM** film)); *-- 3.2* **INSERT INTO** new\_film (**film\_id**, **title**, **language\_id**, **rental\_duration**, **rental\_rate**, **replacement\_cost**, **release\_year**, **last\_update**) **VALUES** (1001, **'Jason Bourne'**, 1, 5, 1.75, 16.99, 2016, (**SELECT** *sysdate* **FROM** *dual*)); *-- 3.3* **UPDATE** new\_film **SET rental\_rate** = **rental\_rate** \* 1.15 **WHERE rental\_rate** < 2; *-- 3.4* **CREATE OR REPLACE VIEW** cheap\_film **AS SELECT title**, **description**, **rental\_rate**, **length FROM** new\_film **WHERE rental\_rate** <= 2 **WITH CHECK OPTION**; *-- 3.5: COMMIT and ROLLBACK does not affect the session since -- data-dictionary manipulations (3.4) are followed by an automatic commit -- 3.6* **UPDATE** cheap\_film **SET** rental\_rate = rental\_rate \* 1.15 **WHERE** rental\_rate < 2;

  
*-- Does not work since some films' rental\_rate would  
-- exceed the upper boundary (2)  
  
-- 3.7***DELETE  
FROM** new\_film  
**WHERE rental\_rate** > 1.79;  
  
*-- 3.8 (1.79 \* 1.15) > 2, Since those entries are deleted in 3.7, all other values (x \* 1.15)*

*-- 3.9***DROP TABLE** new\_film;

  
**DROP VIEW** cheap\_film;

