*Wie aktiv waren die User: Wie viele unterschiedliche Aufgaben wurden bearbeitet: Angabe*

*von Durchschnitt, Minimum und Maximum*

SELECT min(TASK.AMOUNT) as MIN\_TASKS, max(TASK.AMOUNT)as MAX\_TASKS, round(avg(TASK.AMOUNT),2) as MEAN\_TASKS  
FROM (  
 SELECT count(d.TASK) as AMOUNT  
 FROM LOG\_DATA d  
 LEFT JOIN LOG\_ID i ON d.LOG\_ID = i.LOG\_ID  
 GROUP BY i.USER\_ID  
 ) TASK

*Was ist die Fehlerquote je User (Anzahl falsch gelöster Aufgaben / Anzahl aller gelöster*

*Aufgaben) dabei zählt jeder Versuch als gelöste Aufgabe*

SELECT correct\_amount.b\_user\_id as user\_id, (correct\_amount.b\_AMOUNT/sum\_amount.a\_AMOUNT) as Fehlerquote  
FROM (  
 SELECT d.CORRECT as b\_cor, i.USER\_ID as b\_user\_id, count(d.TASK) as b\_AMOUNT  
 FROM LOG\_DATA d  
 LEFT JOIN LOG\_ID i ON d.LOG\_ID = i.LOG\_ID  
 GROUP BY i.USER\_ID, d.CORRECT  
 ) correct\_amount  
LEFT JOIN (  
 SELECT i.USER\_ID as a\_user\_id, count(d.TASK) as a\_AMOUNT  
 FROM LOG\_DATA d  
 LEFT JOIN LOG\_ID i ON d.LOG\_ID = i.LOG\_ID  
 GROUP BY i.USER\_ID  
 ) sum\_amount  
ON correct\_amount.b\_user\_id = sum\_amount.a\_user\_id  
WHERE correct\_amount.b\_cor = 'False';

*Wie viele User haben wie viele Bücher ganz durchgearbeitet?*

SELECT books.user\_id, SUM(Case when test.test\_counter = books.counter then 1 else 0 end)  
FROM (  
 SELECT u.user\_id, count(u.task\_comp) as counter, u.book\_id as book  
 FROM (  
 SELECT DISTINCT double.user\_id as user\_id, double.task\_comp as task\_comp, double.book\_id  
 FROM (  
 SELECT l.USER\_ID as user\_id, d.TASK as task\_comp, TB.TASK as task\_bk, TB.BOOK\_ID as book\_id  
 FROM LOG\_DATA d  
 LEFT JOIN LOG\_ID l on d.LOG\_ID = l.LOG\_ID  
 RIGHT JOIN TASK\_BOOK TB on d.TASK = TB.TASK  
 WHERE d.CORRECT = 'True'  
 ) double  
 ) u  
 GROUP BY u.user\_id, u.book\_id  
 ) books  
LEFT JOIN (SELECT count(TASK) as test\_counter, BOOK\_ID  
FROM TASK\_BOOK  
GROUP BY BOOK\_ID) test on books.book = test.BOOK\_ID  
GROUP BY books.user\_id

*Welche Bücher wurden von wie vielen Usern ganz durchgearbeitet?*

SELECT books.book, SUM(Case when test.test\_counter = books.counter then 1 else 0 end)  
FROM (  
 SELECT u.user\_id, count(u.task\_comp) as counter, u.book\_id as book  
 FROM (  
 SELECT DISTINCT double.user\_id as user\_id, double.task\_comp as task\_comp, double.book\_id  
 FROM (  
 SELECT l.USER\_ID as user\_id, d.TASK as task\_comp, TB.TASK as task\_bk, TB.BOOK\_ID as book\_id  
 FROM LOG\_DATA d  
 LEFT JOIN LOG\_ID l on d.LOG\_ID = l.LOG\_ID  
 RIGHT JOIN TASK\_BOOK TB on d.TASK = TB.TASK  
 WHERE d.CORRECT = 'True'  
 ) double  
 ) u  
 GROUP BY u.user\_id, u.book\_id  
 ) books  
LEFT JOIN (SELECT count(TASK) as test\_counter, BOOK\_ID  
FROM TASK\_BOOK  
GROUP BY BOOK\_ID) test on books.book = test.BOOK\_ID  
GROUP BY books.book

*An welchem Tag wurden die meisten Tasks richtig gelöst?*

SELECT TASK, to\_char(DATE1,'DD-MM-YY'), Count(TASK)  
FROM LOG\_DATA  
GROUP BY TASK, to\_char(DATE1,'DD-MM-YY')  
FETCH FIRST 1 ROWS ONLY;

*Welcher war der schwerste Task?*

SELECT TASK, sum(TASK)  
FROM LOG\_DATA  
WHERE CORRECT = 'False'  
GROUP BY TASK  
ORDER BY SUM(TASK) DESC  
FETCH FIRST 1 ROWS ONLY;