



**Datenbankentwicklung  
und  
Datenbankdokumentation**

## Agenda

- Joins
- Aggregatfunktionen
- Tricks beim Suchfiltern
- Umkreissuche
- Referenzielle Integrität
- Celco's Worm Tree
- DB-Dokumentation mit TYPO3

## Inner Joins

```
SELECT  c.companyname,  
        p.firstname,  
        p.lastname  
FROM    company c INNER JOIN persons p  
ON      c.company_id = p.company_id
```

## Inner Joins, alternative Schreibweise

```
SELECT  c.companyname,  
        p.firstname,  
        p.lastname  
FROM    company c,  
        persons p  
WHERE   c.company_id = p.company_id
```

## Outer Join

```
SELECT  c.companyname,  
        p.firstname,  
        p.lastname  
FROM    company c LEFT OUTER JOIN persons p  
ON      c.company_id = p.company_id
```

## Self Join

```
SELECT  c1.companyname,  
        c2.companyname AS possiblepartner  
FROM    company c1 LEFT OUTER JOIN company c2  
ON      c1.specialisation = c2.specialisation
```

## Union Join

```
SELECT  c.companyname
FROM    company c
WHERE   company_id = 1

UNION

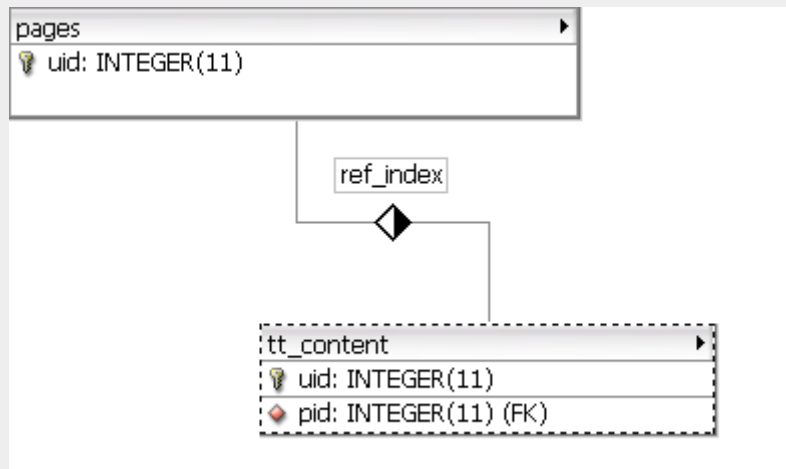
SELECT  c.companyname
FROM    company c
WHERE   company_id = 2
```

## Aggregatfunktionen

- COUNT
- SUM
- AVG
- MAX
- MIN

```
SELECT    count(c.uid) AS anzahl,  
          p.uid  
FROM      pages p LEFT OUTER JOIN tt_content c  
ON        p.uid = c.pid  
GROUP BY p.uid
```

## Referenzielle Integrität (InnoDB)



```

CREATE TABLE pages (
  uid INTEGER(11) NOT NULL,
  pid INTEGER(11) NOT NULL,
  PRIMARY KEY(uid)
);
  
```

```

CREATE TABLE tt_content (
  uid INTEGER(11) NOT NULL,
  pid INTEGER(11) NOT NULL,

  PRIMARY KEY(uid),
  FOREIGN KEY(pid)
    REFERENCES pages(uid)
    ON DELETE CASCADE
    ON UPDATE NO ACTION
);
  
```

## Suchergebnisse Filtern

```
WHERE title LIKE '%untitled%'
```

```
WHERE CONCAT(title,subtitle,nav_title) LIKE '%untitled%'
```

```
WHERE title LIKE '%untitle_'
```

```
WHERE uid IN (6,3)
```

```
WHERE CONCAT(',',usergroup,',') LIKE ',6,%'
```

```
WHERE deleted + hidden = 0
```

```
WHERE startdate BETWEEN 18485240 AND 18293402
```

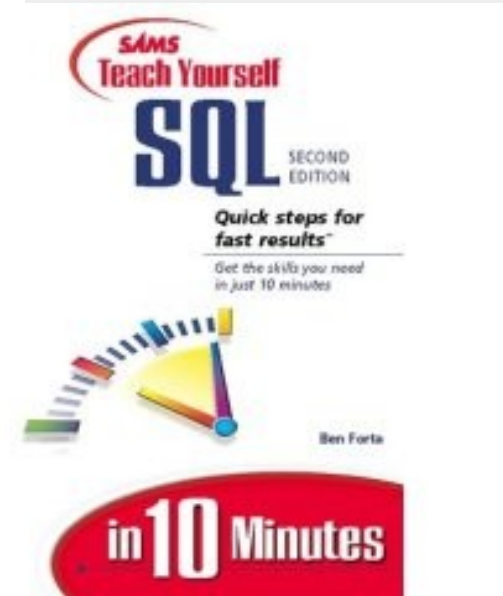
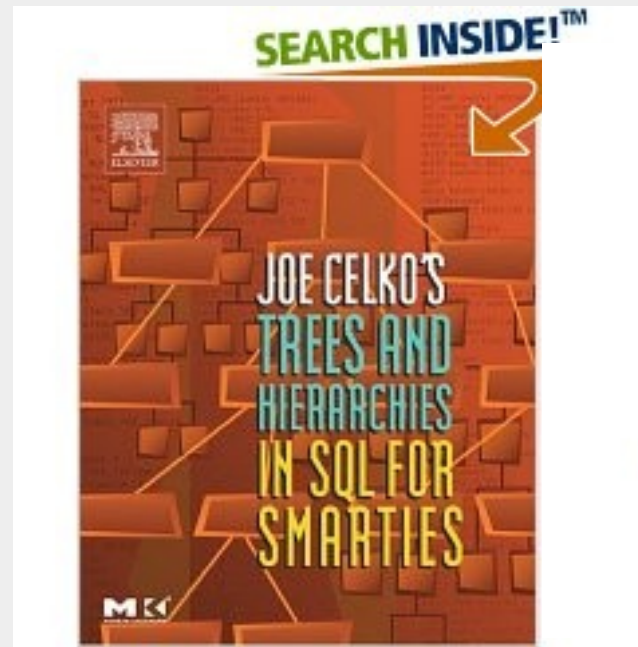
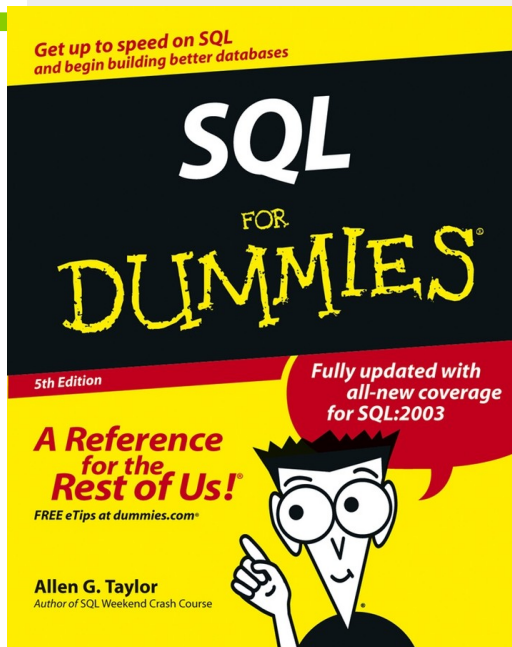
## PLZ-Abfrage

```
SELECT          a.name,  
                ROUND (SQRT (POW ( (u2.utm_lon-u.utm_lon) ,2) +  
                POW ( (u2.utm_lat-u.utm_lat) ,2)) /1000) AS entfernung  
FROM            umkreissuche u  
INNER JOIN     umkreissuche u2  
ON             SQRT (POW ( (u2.utm_lon-u.utm_lon) ,2) +  
                POW ( (u2.utm_lat-u.utm_lat) ,2)) < 10000  
AND            u.zipdisplay_postcode = '81667'  
INNER JOIN     fe_users a  
ON            a.zip = u2.zipdisplay_postcode  
  
ORDER BY      entfernung ASC
```

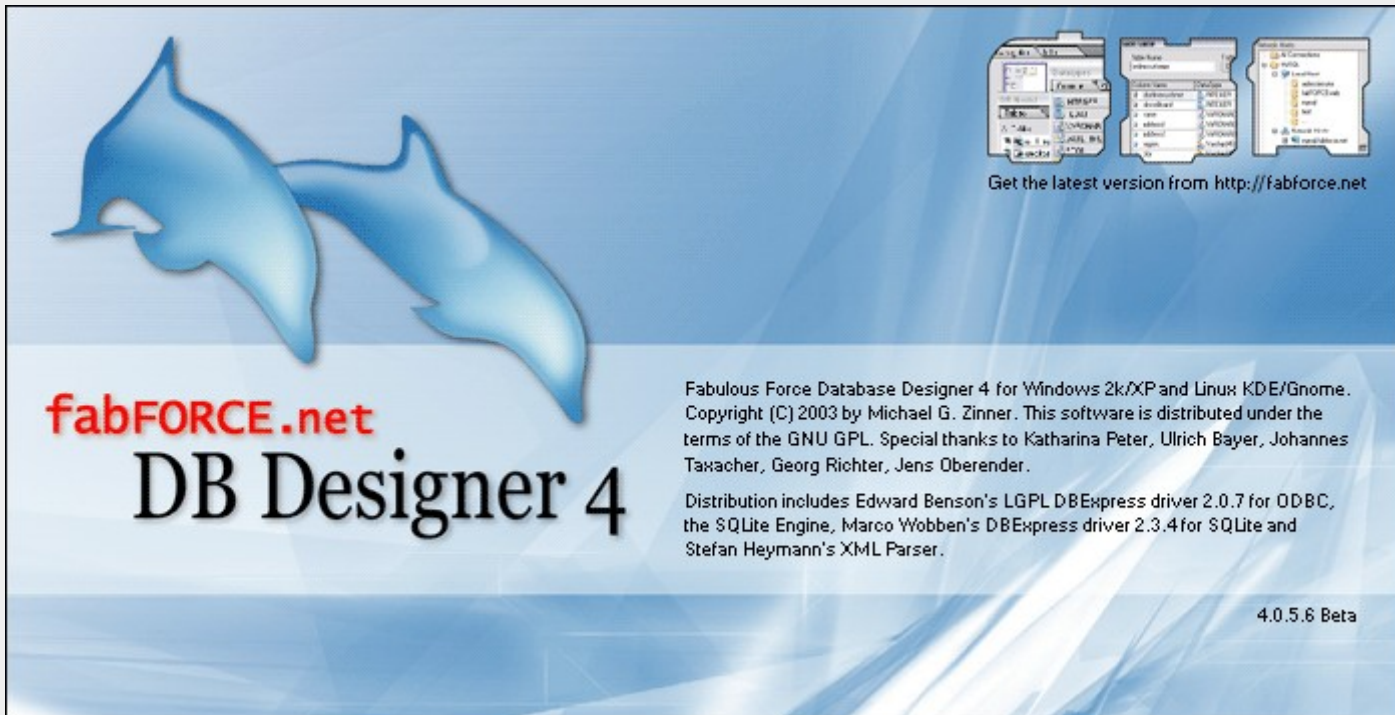
## Rekursion vs. Worm Tree (Celco)

```
function recTree($parent) {  
    SELECT uid  
    FROM pages  
    WHERE pid = $parent  
  
    while ($res) {  
        recTree($uid)  
    }  
    return $tree  
}
```

```
SELECT    p.uid,  
          p.lvl,  
          p.lft,  
          p.rgt  
  
FROM      pages p  
  
ORDER BY  p.lft
```



## DBDesigner



Get the latest version from <http://fabforce.net>

**fabFORCE.net**  
**DB Designer 4**

Fabulous Force Database Designer 4 for Windows 2k/XP and Linux KDE/Gnome.  
Copyright (C) 2003 by Michael G. Zinner. This software is distributed under the terms of the GNU GPL. Special thanks to Katharina Peter, Ulrich Bayer, Johannes Taxacher, Georg Richter, Jens Oberender.

Distribution includes Edward Benson's LGPL DBExpress driver 2.0.7 for ODBC, the SQLite Engine, Marco Wobben's DBExpress driver 2.3.4 for SQLite and Stefan Heymann's XML Parser.

4.0.5.6 Beta

Ext dev eval

TYPO3 



**Datenbankentwicklung  
und  
Datenbankdokumentation**