

### Datentyp

integer: 1,2,3  
 serial: fortlaufende Ganzzahl  
 float: 1.23  
 numeric: 1, 1.23  
 varchar: 'Hallo Welt!'  
 date: '2001-01-21'  
 time: '11:55:30'  
 timestamp: '2014-10-02 10:39:26.332'  
 current\_timestamp: '2014-10-02 10:39:26.332'  
 boolean: true, false

### Einschraenkung (Constraint)

PRIMARY KEY ... Primärschlüssel  
 UNIQUE ... Einzigartiger Wert  
 NOT NULL ... Eingabe muss erfolgen  
 CHECK ... Definierter Wertebereich  
 DEFAULT ... Wenn nichts eingetragen  
 REFERENCES ... Verweis auf  
 Primärschlüssel

### Bedingung

Logisch: =,<>,! =,<>,<=,>=  
 Grenzen: BETWEEN *minimum* AND *maximum*  
 Aus Menge: IN (*wert*, ...)  
 Wortstück: LIKE '*stringteil*'

```
... WHERE alter = 35
... WHERE alter BETWEEN 35 AND 45;
... WHERE alter IN (35,45,55);
... WHERE name LIKE '%ei%';
... WHERE name LIKE '%er';
```

### Aggregatfunktionen

AVG() ... Mittelwert der Spalte  
 COUNT() ... Anzahl der Tupel  
 MAX() ... Maximaler Wert der Spalte  
 MIN() ... Minimaler Wert der Spalte  
 STDDEV() ... Standardabw der Spalte  
 SUM() ... Summer der Spalte  
 VARIANCE() ... Varianz der Spalte  
 Als WHERE-Bedingung ist keine  
 Aggregatfunktion möglich.

```
SELECT MAX(einkommen)
FROM person;
```

### Verknuempfungoperatoren

AND  
 OR

### Tabelle erstellen

```
CREATE TABLE tabellenname (
  attributname datentyp constraint,
  ...
);

CREATE TABLE personen (
  id serial UNIQUE PRIMARY KEY,
  name varchar NOT NULL,
  alter integer NOT NULL CHECK (alter
  IN(25,35,45,55)),
  einkommen float NOT NULL
  CHECK(einkommen
  BETWEEN 1000.00 AND 3000.00),
  zeitstempel timestamp DEFAULT
  current_timestamp,
  auto integer REFERENCES
  kfz(nummer)
);
```

### Tabelle loeschen

```
DROP TABLE tabellenname;

DROP TABLE personen;
```

### Tabelle aus bestehender erstellen

```
CREATE TABLE tabellenname2
AS (
  abfrage
FROM tabellenname1
);
```

```
CREATE TABLE personen_neu
AS (
  SELECT * FROM personen
);
```

### Datensatz erstellen

```
INSERT INTO tabellenname (
  attributname,
  ...
) VALUES (
  wert,
  ...
);
```

```
INSERT INTO personen (
  name,
  alter,
  einkommen,
  auto
) VALUES (
  'Hans Gruber',
  35,
  2487.24,
  'G383AS'
);
```

### Spalte in Tabelle erstellen

```
ALTER TABLE tabellenname
ADD attributname datentyp;

ALTER TABLE personen
ADD ledig BOOLEAN;
```

### Gesamten Tabelleninhalt loeschen

```
DELETE FROM tabellenname

DELETE FROM personen;
```



### Bestimmte Tabelleninhalte loeschen

```
DELETE FROM tabellenname
WHERE bedingung
```

```
DELETE FROM personen
WHERE alter=35;
```

### Tabelleninhalte editieren

```
UPDATE tabellenname
SET attributname;
```

```
UPDATE personen
SET alter = 55
WHERE
name = 'Hans Gruber';
```

### Gesamten Tabelleninhalt abfragen

```
SELECT * FROM tabellenname
```

```
SELECT * FROM personen;
```

### Bestimmte Tabelleninhalte abfragen

```
SELECT attributname
FROM tabellenname
WHERE bedingung;
```

```
SELECT name , alter
FROM personen;
```

```
SELECT name, alter
FROM personen
WHERE einkommen <= 2000.00
OR
WHERE einkommen >=3000;
```

### Verschachtelte Abfragen

```
SELECT attributname
FROM tabellenname
WHERE attributname bedingung (
SELECT ...
);
```

```
SELECT *
FROM personen
WHERE einkommen = (
SELECT MAX(einkommen)
FROM personen
);
```

### Mehrere Tabellen Abfragen

```
SELECT tabellenname1.attributname1
FROM tabellenname1, tabellenname2
WHERE tabellenname1.attributname1 =
tabellenname2.attributname2*;
```

```
SELECT personen.name
FROM personen, vorgaben
WHERE personen.einkommen <
vorgaben.mindestlohn;
```

### Variablenname deklarieren

```
... FROM tabellenname as variablenname, ...
```

```
SELECT a.name
FROM personen as a, vorgaben as b
WHERE a.einkommen <
b.mindestlohn;
```

### Synthetische Tabelle erstellen (View)

```
CREATE VIEW viewname AS abfrage
```

```
CREATE VIEW reiche_personen
AS
SELECT name
FROM personen
WHERE einkommen >100000.00;
```

