

Wir suchen Unterstützung für das Wintersemester 2024/25

- Algorithmen und Datenstrukturen
- Data Science 2
- Einführung in Deep Learning
- Programmierpraktikum 2
- Programmierung
- Rechnerarchitektur

Mehr Informationen und Kontakt:

<https://www.cs.hhu.de/jobs>





Übung 6

Datenbanken: Eine Einführung

■ Tabellen erstellen

```
create table table_name (  
    column1 datatype constraint,  
    column2 datatype constraint,  
    ....  
    constraint1,  
    constraint2,  
    ...);
```

Constraints

- not null
- default <Wert>
- auto_increment
- unique
- primary key
- foreign key

Datentypen

Datentyp	Bedeutung
smallint integer, int	ganzzahlige Werte
decimal(p,q) numeric(p,q)	Festkommazahlen mit Genauigkeit <i>p</i> und <i>q</i> Nachkommastellen
float(p) real double precision	Gleitkommazahlen
character(n), char(n) character varying(n), varchar(n)	Zeichenketten fester Länge <i>n</i> Zeichenketten variabler Länge <i>n</i>
boolean	Wahrheitswerte (true, false)
bit(n) bit varying(n)	Bitfolgen fester Länge <i>n</i> Bitfolgen variablen Länge <i>n</i>
date time timestamp interval	Datum Zeit Zeitstempel Zeitintervall

■ Tabelle löschen

```
drop table table_name;
```

■ Tabelle ändern

```
alter table table_name  
  modification1,  
  modification2,  
  ...;
```

Modifikationen zB:

- add column *spaltendefinition*
 - Spaltendefinition = spaltenname datentyp constraints
- drop column *spaltenname*
- alter column *spaltenname* ...
 - neuerDatentyp
 - set default defaultWert
- rename column *altername* to *neuename*
- add **constraint** oder add constraint
 - add unique (spaltenname)
 - add constraint BedName unique (spaltenname)
- drop **constraint** oder drop constraint
 - drop unique (spaltenname)
 - drop constraint BedName

■ **select-from-where**

`select` [distinct] *Spalten*
from *Tabellen*
where *Bedingungen*;

■ **Beispiel:** R(A,B) S(A,C)

`select distinct R.A, B`
`from R, S`
`where R.A = S.A`
`and C <> 5;`

entspricht $\pi_{R.A, B}(\sigma_{R.A=S.A \wedge C \neq 5}(R \times S))$