

## Aufgabe 1

```

28     ADRID      int                auto_increment primary key,
29     FK_CID     int                not null,
30     ASTREET    varchar(50)        not null,
31     ACITY      varchar(50)        not null,
32     AFEDSTATE  enum ('NRW', 'NS', 'HE', 'BY', 'BW', 'RP', 'SH', 'SA', 'TH',
    'BR', 'HB', 'SL', 'MVP', 'BLN') not null,
33     ACITYCODE  varchar(10)        not null,
34     AOUNTRY    varchar(30)        not null,
35     ADATE      timestamp default current_timestamp() not null,
36     foreign key (FK_CID) references costumer(cid)
37 );
38 # 4. order
39 CREATE TABLE order1 (
40     OID int not null auto_increment primary key ,
41     ODELIVDATE date ,
42     OSTATUS enum('In Bearbeitung', 'Offen', 'Erledigt'),
43     FK_CID int,
44     BCOMMENT varchar(256),
45     OCHANGEDATE timestamp,
46     foreign key (FK_CID) references costumer(CID)
47 );
48
49 # 5. productcategory
50 create table productcategory
51 (
52     PCATID      int      auto_increment primary key,
53     PCATNAME    varchar(40)          not null,
54     PCATORIGIN  enum('Selbst gezüchtet', 'Zukauf')
55
56 );
57 # 6. Lieferant
58
59 create table supplier
60 (
61     SUPID      int      auto_increment primary key,
62     SUPNAME    VARCHAR(255)          not null,
63     SUPCONTACTNAME  VARCHAR(255)          not null,
64     SUPPHONE    VARCHAR(255)          not null,
65     SUPEMAIL    VARCHAR(255)          not null,
66     SUPWWW      VARCHAR(255)          not null
67 );
68 # 7. product
69 create table product
70 (
71     PRID      int      auto_increment primary key,
72     PRNAME    varchar(40)          not null,
73     PRAMOUNT  int                not null,
74     PPRICENETTO  FLOAT          not null,
75     PRIMODIFDATE    timestamp default current_timestamp() not null,
76     FK_PCATID  int                not null,
77     FK_SUPID   int                not null,
78     foreign key (FK_PCATID) references productcategory(PCATID),
79     foreign key (FK_SUPID) references supplier(SUPID)
80
81 );

```

```

82
83
84
85 # 8. service
86 create table SERVICE
87 (
88     SERVID int auto_increment primary key,
89     SERVNAME VARCHAR(255) not null,
90     SERVPRICE FLOAT not null
91 );
92 );
93
94 # 9. orderdetails
95 create table orderdetail
96 (
97     ODID int auto_increment primary key,
98     ODAMOUNT tinyint not null,
99     FK_OID int not null,
100    FK_PRID int ,
101    FK_SERVID int ,
102    ODATE timestamp default current_timestamp() not null,
103    foreign key (FK_OID) references order1(OID),
104    foreign key (FK_PRID) references product(prid),
105    foreign key (FK_SERVID) references service(SERVID)
106 );
107
108
109 # 10. employee
110 create table employee(
111     EID int auto_increment primary key,
112     EFIRSTNAME varchar(40) not null,
113     ELASTNAME varchar(40) not null,
114     ESALUTATION enum ('Hr.', 'Fr.', 'Div.') not null,
115     ESALARY FLOAT not null
116 );
117
118 # 11. Mitarbeiterservice
119 create table employeeService
120 (
121     ESID int auto_increment primary key,
122     FK_EID int ,
123     FK_SERVID int ,
124     foreign key (FK_EID) references employee(eid),
125     foreign key (FK_SERVID) references service(SERVID)
126 );
127
128 #12. Rechnung
129 create table invoice
130 (
131     INVID int auto_increment primary key,
132     FK_OID int not null,
133     INVDATE date ,
134     foreign key (FK_OID) references order1(oid)
135 );
136

```

```

137 insert into account(ACCNAME,ACCPWD,ACCTYPE) values
138     ('kunde1', 'kunde1pwd', 1),
139     ('kunde2', 'kunde2pwd', 1),
140     ('kunde3', 'kunde3pwd', 1),
141     ('kunde4', 'kunde4pwd', 1),
142     ('kunde5', 'kunde5pwd', 1),
143     ('kunde6', 'kunde6pwd', 1),
144     ('kunde7', 'kunde7pwd', 1),
145     ('kunde8', 'kunde8pwd', 1),
146     ('kunde9', 'kunde9pwd', 1),
147     ('kunde10', 'kunde10pwd', 1),
148     ('admin1', 'admin1pwd', 2)
149 ;
150 insert into costumer(CFIRSTNAME,CLASTNAME,CSALUTATION
,CEMAIL,CPHONE,CBIRTHDATE,FK_ACCID) values
151     ('Mila',      'Müller',   'Fr.', 'milamueeller@icloud.com',
0521654654, '1971-12-08',1),
152     ('Emilia',   'Schmidt',  'Fr.', 'emiliaschmidt@verizon.net',
0521545245, '1972-08-13',2),
153     ('Leni',     'Schneider','Fr.', 'lenischneider@verizon.net',
052156464,  '1975-08-25',3),
154     ('Mira',     'Fischer',   'Fr.', 'mirafischer@aol.com',
0521685745, '1975-09-01',4),
155     ('Sophia',   'Weber',     'Fr.', 'sophiaweber@icloud.com',
0521654821, '1982-08-06',5),
156     ('Adam',     'Meyer',     'Hr.', 'adammeyer@msn.com',
0521654826, '1986-08-17',6),
157     ('Noah',     'Wagner',    'Hr.', 'noahwagner@icloud.com',
0521729287, '1987-07-28',7),
158     ('Felix',    'Becker',    'Hr.', 'felixbecker@outlook.com',
0521569654, '1993-09-21',8),
159     ('Leon',    'Schulz',    'Hr.', 'leonschulz@gmail.com',
0521875412, '1996-09-14',9),
160     ('Emil',     'Hoffmann',  'Hr.',
'emilhoffmann@sbcglobal.net',0521632548, '2006-10-04',10)
161 ;
162
163 insert into adress(FK_CID,ASTREET,ACITY,AFEDSTATE,ACITYCODE,ACOUNTRY)values
164     (1, 'Hauptstraße 1', 'Hamburg', 'NRW', '05996', 'Deutschland' ),
165     (2, 'Schulstraße 2', 'Dresden', 'NS', '60137', 'Deutschland'),
166     (3, 'Gartenstr. 3', 'München', 'HE', '74405', 'Deutschland'),
167     (4, 'Bahnhofstr. 4', 'Lübeck', 'BY', '09610', 'Deutschland'),
168     (5, 'Dorfstraße 5', 'Freiburg', 'RP', '88672', 'Deutschland'),
169     (6, 'Bergstraße 6', 'Berlin', 'SH', '36301', 'Deutschland'),
170     (7, 'Birkenweg 7', 'Potsdam', 'SA', '70427', 'Deutschland'),
171     (8, 'Lindenstr. 8', 'Trier', 'TH', '74057', 'Deutschland'),
172     (9, 'Kirchstraße 9', 'Lage', 'BR', '55899', 'Deutschland'),
173     (1, 'Waldstr. 1', 'Herford', 'HB', '61893', 'Deutschland');
174
175 insert into order1(odelivdate, ostatus, fk_cid, bcomment ) value
176     ('2022-06-30','offen',1,'Bitte morgens liefern!'),
177     ('2022-07-04','offen',1,''),
178     ('2022-09-14','Erledigt',3,''),
179     ('2022-10-11','Erledigt',4,''),
180     ('2022-11-29','In Bearbeitung',5,''),

```

```

181      ('2022-11-30','Erledigt',6,''),
182      ('2022-11-30','In Bearbeitung',6,'');
183
184
185
186 insert into productcategory(pcatname, pcatorigin) values
187
188      ('Schnittblume', 'Zukauf'),
189      ('Topfpflanzen', 'Selbst gezüchtet'),
190      ('Beetpflanzen', 'Selbst gezüchtet'),
191      ('Balkonpflanzen', 'Zukauf'),
192      ('Büsche', 'Zukauf')
193      ;
194
195 insert into supplier(supname, supcontactname, supphone, supemail, supwww)
196 values
197      ('Blumen Posch',                                'Elisa Posch', 065464138,
198      'bestellung@blumen-posch.de',                    'www.blumen-posch.de'),
199      ('Flower Gmbh Blumenhandel',                    'Ayla Bauer', 068565456,
200      'info@flower-blumenhandel.de',                    'www.flower-blumenhandel.de'),
201      ('DB Blumeneinzelhandels GmbH & Co. KG',        'Lea Koch', 058646545,
202      'order@db-blumeneinzelhandels.de',                'www.db-blumeneinzelhandels.de'),
203      ('Sperling Blumen Import',                      'Erik Schmitt', 065879634,
204      'bestellung@sperling.de',                        'www.sperling-blumen.de'),
205      ('Pflanzen-Discount',                            'Alex Braun', 078465345,
206      'info@pflanzen-discount.de',                    'www.pflanzen-discount.de')
207      ;
208
209
210
211
212
213
214
215
216
217
218
219 insert into product(prname, pramount, ppricenetto, fk_pcatid, fk_supid)
220 values
221      ('Gänseblümchen',                                99, 0.99,1,1),
222      ('Rosen',                                          10, 5.00,1,1),
223      ('Zamioculcas',                                   20, 2.00,2,2),
224      ('Bogenhanf',                                     5, 3.50,2,2),
225      ('Mexikanische Sonnenblume',                    10, 9.20,3,3),
226      ('Löwenohr',                                     10, 5.99,3,3),
227      ('Geranie ',                                     5, 6.60,4,4),
228      ('Petunie ',                                     10, 1.99,4,4),
229      ('Schmetterlingsflieder',                       1, 2.40,5,5),
230      ('Lavendelheide',                                2, 6.80,5,5)
231      ;
232
233
234
235
236
237
238
239 insert into service(servname, servprice) value
240      ('Trauerkranz',                                100.00),
241      ('gesteckte Arrangement',                      200.00),
242      ('bepflanzte Trauerschale',                    300.00),
243      ('Brautstrauß',                                400.00),
244      ('Königinnenstrauß',                          100.00),
245      ('Kutschengesteck',                            200.00),
246      ('Autogesteck',                                300.00),
247      ('Neuanlage Grabstätte',                      400.00),
248      ('Grabumrandung mit Heckenpflanzen', 100.00),

```

```

229      ('Bodendeckende Bepflanzung',      200.00)
230  ;
231
232  insert into orderdetail(odamount, fk_oid, fk_prid, fk_servid) value
233      (1,2,3,NULL),
234      (5,1,2,NULL),
235      (4,5,1,NULL),
236      (3,4,5,NULL),
237      (2,3,4,NULL),
238      (4,6,3,NULL),
239      (4,7,6,NULL)
240  ;
241
242  insert into employee(efirstname, elastname, esalutation, esalary) value
243      ('Lio', 'Krause', 'Hr.', 2000.00),
244      ('Emilio', 'Zimmermann', 'Hr.', 2300.00),
245      ('David', 'Hartmann', 'Hr.', 1900.00),
246      ('Ahmad', 'Schmitz', 'Hr.', 3000.00),
247      ('Tom', 'Hofmann', 'Div.', 3500.00);
248
249  insert into employeeservice(fk_eid, fk_servid) value
250      (1,2),
251      (1,3),
252      (2,2),
253      (3,3),
254      (4,4),
255      (5,5)
256  ;
257
258  insert into invoice(FK_OID, INVDATE) value
259      (1, '2022-06-30'),
260      (2, '2022-07-04'),
261      (3, '2022-09-14'),
262      (4, '2022-10-11'),
263      (5, '2022-11-29');

```

## Aufgabe 2

1. Führen Sie über das SQL-Kommando SELECT die folgenden SQL-Abfragen in der Tabelle product durch:

- Sortieren Sie die Datensätze mit einer SELECT-Anweisung nach Namen und Typ (ORDER BY - Klausel)!

```

1  select * from product order by PRNAME, FK_PCATID ; # wenn zuerst nach
   Name und dann nach Typ (macht weniger Sinn)
2  select * from product order by FK_PCATID, PRNAME ; # Zuerst Typ und
   dann nach Namen sortiert

```

- Selektieren Sie **nacheinander**? (!) die Namen der Produkte (Alias: Produktname) und Einzelpreis mit einem Einzelpreis zwischen Preis 1 bis Preis 2 (WHERE - Klausel mit logischer AND-Verknüpfung) und den Produktnummern zwischen unterer und oberer Grenze!

```
1 | select PRNAME as Produktname, PPRICENETTO from product where  
   PPRICENETTO between 3.5 and 6.5;
```

2. Führen Sie die folgenden SQL-Abfragen als Verbundabfrage über die Tabellen order und customer durch:

1. Ermitteln Sie Anzahl der Bestellungen eines Kunden (Alias: Anzahl Bestellungen für einen Kunden) (COUNT - Funktion) und geben Sie diese und den Namen des Kunden (Alias: Kunde) aus!

```
1 | select count(FK_CID) as Anzahl1 , concat(CLASTNAME, ', ', CFIRSTNAME) as  
   Kunde from order1, costumer where CID=3 and FK_CID=CID;
```

2. Ermitteln Sie die Namen aller Kunden (Alias: Kundenname) und die Anzahl der Bestellungen für jeden Kunden (Alias: Anzahl) und Gruppieren Sie die Ergebnisse nach der Anzahl (COUNT und GROUP BY - Klausel)!

```
1 | select concat(CLASTNAME, ', ', CFIRSTNAME) as Kunde, count(FK_CID) as  
   Anzahl1 from order1, costumer where FK_CID=CID group by FK_CID;
```

3. Ermitteln Sie den Kundennamen (Alias: Kunde) und die Anzahl der Bestellungen für ein konkretes Produkt (via FK\_PRID) für diesen Kunden (Alias: Anzahl) (COUNT und GROUP BY, INNER JOIN - Klausel)!

```
1 | select concat(CLASTNAME, ', ', CFIRSTNAME) as Kunde,  
2 |       count(FK_CID) as Anzahl1  
3 |       from order1  
4 |       inner join costumer on order1.FK_CID = costumer.CID  
5 |       inner join orderdetail o on order1.OID = o.FK_OID  
6 |       where FK_PRID=3  
7 |       group by FK_CID;
```

3. Führen Sie die folgenden SQL-Abfragen ggf. als Verbundabfrage über die Tabellen order, customer, orderdetail und product durch:

1. Ermitteln Sie die Produktnamen (Alias: Produkt) für alle Produkttypen (Alias: Produkttyp) aller Einträge in der Tabelle product.

```
1 | select PRNAME as Produkt from product;
```

2. Ermitteln Sie die Summe aller Produkte in der Tabelle product.

```
1 | select count(PRNAME) from product;
```

3. Ermitteln Sie den Kundennamen (Alias: Kunde),  
die bestellten Produkte (Alias: Produktname),  
das Lieferdatum und  
den jeweiligen Bestellpreis für einen bestimmten Kunden über alle Bestellungseinträge.

```
1 select CLASTNAME as Kunde,  
2         PRNAME as Produkt,  
3         ODELIVDATE as Lieferdatum,  
4         PPRICENETTO as Preis  
5 from order1  
6 inner join costumer c on order1.FK_CID = c.CID  
7 inner join orderdetail o on order1.OID = o.FK_OID  
8 inner join product p on o.FK_PRID = p.PRID  
9 where cid=6
```

4. Welche Produkte wurden noch nicht bestellt?

```
1 select PRNAME as Produkt  
2 from order1  
3 inner join orderdetail o on order1.OID = o.FK_OID  
4 right join product p on o.FK_PRID = p.PRID  
5 where OID is null;
```

5. Ermitteln Sie die Anzahl an Bestellungen für jedes Produkt!

```
1 select PRNAME as Produkt,  
2         sum(ODAMOUNT) as Anzahl  
3 from product  
4 left join orderdetail o on product.PRID = o.FK_PRID  
5 group by Prid;
```

---

Fragen von Grünwoldt

Alle Kunden ausgeben mit Service den sie gebucht haben

```
1 select CLASTNAME,  
2         SERVNAME  
3 from order1  
4 inner join orderdetail o on order1.OID = o.FK_OID  
5 left join costumer c on order1.FK_CID = c.CID  
6 left join service s on o.FK_SERVID = s.SERVID  
7 where FK_SERVID is not null ;
```



Wie oft alle Kunden einen bestimmten Service gekauft haben

```
1 select SERVNAME,  
2      sum(ODAMOUNT)  
3      from order1  
4      inner join orderdetail o on order1.OID = o.FK_OID  
5      left join service s on o.FK_SERVID=s.SERVID  
6      where FK_SERVID is not null  
7      group by FK_SERVID;
```