

Zweites DBA Praktikum

Aufgabe 1

Erstellen Sie eine relationale Datenbank mit mindestens 12 Tabellen und füllen Sie Ihre Tabellen mit Datensätzen (mind. 10 Datensätze in Elterntabellen und mind. fünf Datensätze pro Kindtabelle)!

```
1  drop database floristik;
2  create database floristik;
3  use floristik;
4  # account
5      create table account
6  (
7      ACCID    int auto_increment           primary key,
8      ACCNAME  varchar(255)                 not null,
9      ACCPWD   varchar(255)                 not null,
10     ACCTYPE  int                           not null
11 );
12 # costumer
13 create table costumer(
14     CID        int      auto_increment      primary key,
15     CFIRSTNAME varchar(40)                  not null,
16     CLASTNAME  varchar(40)                  not null,
17     CSALUTATION enum ('Hr.', 'Fr.', 'Div.') not null,
18     CEMAIL     varchar(100)                 not null,
19     CPHONE     varchar(50)                  not null,
20     CBIRTHDATE date                        not null,
21     FK_ACCID   int                          not null,
22     foreign key (FK_ACCID) references account(accid)
23 );
24
25 # adress
26 create table adress
27 (
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',
33     'BR', 'HB', 'SL', 'MVP', 'BLN') not null,
34     ACITYCODE  varchar(10)                  not null,
35     AOUNTRY    varchar(30)                  not null,
36     ADATE      timestamp default current_timestamp() not null,
37     foreign key (FK_CID) references costumer(cid)
38 );
39 # order
40 CREATE TABLE order1 (
41     OID int not null auto_increment primary key ,
42     ODELIVDATE date ,
43     OSTATUS enum('In Bearbeitung', 'Offen', 'Erledigt'),
44     FK_CID int,
```

```

44         BCOMMENT varchar(256),
45         OCHANGEDATE timestamp,
46         foreign key (FK_CID) references costumer(CID)
47     );
48
49     # 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     #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
69     # product
70     create table product
71     (
72         PRID      int      auto_increment primary key,
73         PRNAME varchar(40)                not null,
74         PRAMOUNT int                    not null,
75         PPRICENETTO FLOAT                not null,
76         PRIMODIFDATE timestamp default current_timestamp() not null,
77         FK_PCATID int                    not null,
78         FK_SUPID  int                    not null,
79         foreign key (FK_PCATID) references productcategory(PCATID),
80         foreign key (FK_SUPID) references supplier(SUPID)
81     );
82
83
84
85     # 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     # 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                not null,
101     FK_SERVID int                not null,
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 #Mitarbeiter
109 # 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 #MitarbeiterService
119 create table employeeservice
120 (
121     ESID      int auto_increment primary key,
122     FK_EID    int                not null,
123     FK_SERVID int                not null,
124     foreign key (FK_EID) references employee(eid),
125     foreign key (FK_SERVID) references service(SERVID)
126 );
127
128 #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)
values

```

```

196      ('Blumen Posch',                                'Elisa Posch', 065464138,
197      'bestellung@blumen-posch.de',                  'www.blumen-posch.de'),
198      ('Flower Gmbh Blumenhandel',                    'Ayla Bauer', 068565456,
199      'info@flower-blumenhandel.de',                  'www.flower-blumenhandel.de'),
200      ('DB Blumeneinzelhandels GmbH & Co. KG',        'Lea Koch', 058646545,
201      'order@db-blumeneinzelhandels.de',              'www.db-blumeneinzelhandels.de'),
202      ('Sperling Blumen Import',                      'Erik Schmitt', 065879634,
203      'bestellung@sperling.de',                      'www.sperling-blumen.de'),
204      ('Pflanzen-Discount',                            'Alex Braun', 078465345,
205      'info@pflanzen-discount.de',                    'www.pflanzen-discount.de')
206  ;
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
240
241
242  insert into service(servname, servprice) value
243      ('Trauerkranz',                                100.00),
244      ('gesteckte Arrangement',                      200.00),
245      ('bepflanzte Trauerschale',                    300.00),
246      ('Brautstrauß',                                400.00),
247      ('Königinnenstrauß',                          100.00),
248      ('Kutschengesteck',                            200.00),
249      ('Autogesteck',                                300.00),
250      ('Neuanlage Grabstätte',                      400.00),
251      ('Grabumrandung mit Heckenpflanzen',          100.00),
252      ('Bodendeckende Bepflanzung',                200.00)
253  ;
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999

```

```

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 PRID,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:

- 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        count(FK_PRID) as Anzahl1
3        #count(FK_PRID)*ODAMOUNT as Anzahl1??
4 from product
5 inner join orderdetail o on product.PRID = o.FK_PRID
6 group by FK_PRID;

```