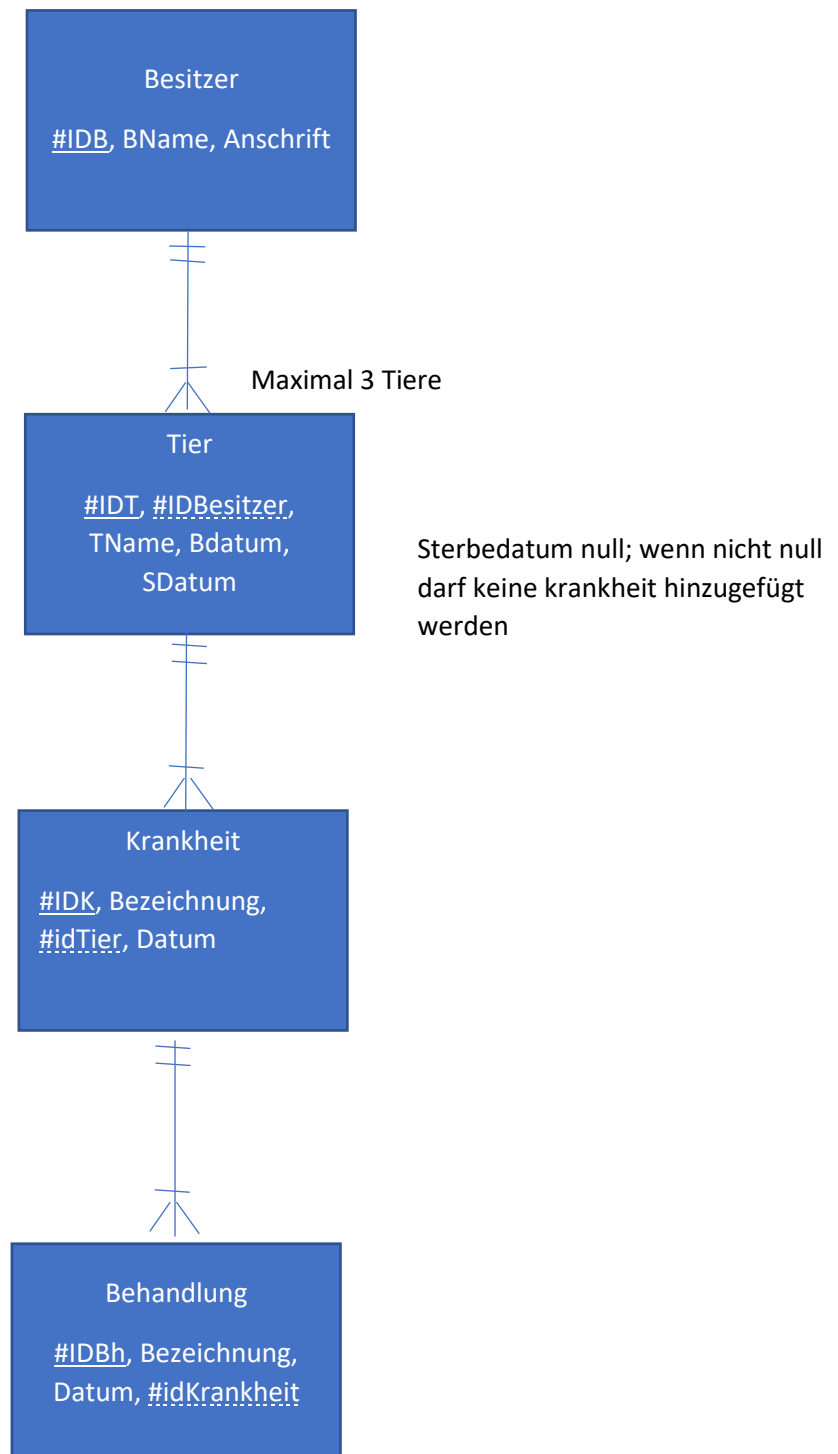


Physische Modell



Physische Modell:

```
drop table besitzer cascade constraints;  
drop table tier cascade constraints;  
drop table krankheit cascade constraints;  
drop table behandlung cascade constraints;  
drop sequence S_TIERARZT_IDSBE;  
drop sequence S_TIERARZT_IDSK;  
drop sequence S_TIERARZT_IDST;  
drop sequence S_TIERARZT_IDSB;
```

```
create table besitzer(  
    idB INTEGER,  
    Bname VARCHAR(50),  
    anschrift VARCHAR(100),  
  
    CONSTRAINT pk_idB PRIMARY KEY(idB)  
);
```

```
create table tier(  
    idT INTEGER,  
    idBesitzer INTEGER,  
    Tname VARCHAR(50),  
    Bdatum DATE,  
    Sdatum DATE,  
  
    CONSTRAINT pk_idT PRIMARY KEY (idT),  
    CONSTRAINT fk_idBesitzer FOREIGN KEY (idBesitzer) REFERENCES besitzer(idB)  
);
```



```
INSERT INTO besitzer VALUES(S_TIERARZT_IDSBE.nextVal, 'Herman', 'Villacherstraße 73a 9220 Velden');
```

```
INSERT INTO besitzer VALUES(S_TIERARZT_IDSBE.nextVal, 'Franz', 'Kieselweg 16 9500 Villach');
```

```
INSERT INTO besitzer VALUES(S_TIERARZT_IDSBE.nextVal, 'Klaus', 'Klagenfurterstraße 22 9500 Villach');
```

```
INSERT INTO besitzer VALUES(S_TIERARZT_IDSBE.nextVal, 'Friedl', 'Bäckerteichstraße 5a 9220 Velden');
```

```
INSERT INTO besitzer VALUES(S_TIERARZT_IDSBE.nextVal, 'Heinzi', 'Seitengasse 4 9500 Villach');
```

```
INSERT INTO besitzer VALUES(S_TIERARZT_IDSBE.nextVal, 'Heinzi', 'Seitengasse 4 9500 Villach');
```

```
select * from besitzer;
```

```
INSERT INTO tier VALUES(S_TIERARZT_IDST.nextVal, 6, 'Hansi (Hase)', to_date('06.05.2017', 'DD.MM.YYYY'), null);
```

```
INSERT INTO tier VALUES(S_TIERARZT_IDST.nextVal, 5, 'Struppi (Katze)', to_date('22.05.2012', 'DD.MM.YYYY'), null);
```

```
INSERT INTO tier VALUES(S_TIERARZT_IDST.nextVal, 4, 'Ginge (Papagei)', to_date('09.07.2008', 'DD.MM.YYYY'), null);
```

```
INSERT INTO tier VALUES(S_TIERARZT_IDST.nextVal, 3, 'Franziska (Maulwurf)', to_date('04.12.2016', 'DD.MM.YYYY'), null);
```

```
INSERT INTO tier VALUES(S_TIERARZT_IDST.nextVal, 2, 'Wolfi (Hund)', to_date('12.03.2010', 'DD.MM.YYYY'), to_date('20.06.2017', 'DD.MM.YYYY'));
```

```
INSERT INTO tier VALUES(S_TIERARZT_IDST.nextVal, 6, 'Rosi (Schildkröte)', to_date('07.02.2013', 'DD.MM.YYYY'), null);
```

```
INSERT INTO tier VALUES(S_TIERARZT_IDST.nextVal, 6, 'Franz_Bertholt (Giraffe)', to_date('17.05.2005', 'DD.MM.YYYY'), null);
```

```
INSERT INTO tier VALUES(S_TIERARZT_IDST.nextVal, 6, 'Franz_Bertholt (Giraffe)', to_date('17.05.2005', 'DD.MM.YYYY'), null);
```

```
select* from tier;
```

```
INSERT INTO krankheit VALUES(S_TIERARZT_IDSK.nextval, 'Hodenkrebs', 6, to_date('20.06.2017', 'DD.MM.YYYY'));
```

```
INSERT INTO krankheit VALUES(S_TIERARZT_IDSK.nextval, 'Nierenprellung', 5, to_date('17.05.2017', 'DD.MM.YYYY'));
```

```
INSERT INTO krankheit VALUES(S_TIERARZT_IDSK.nextval, 'Gehirnerschütterung', 4,
to_date('03.01.2017', 'DD.MM.YYYY'));

INSERT INTO krankheit VALUES(S_TIERARZT_IDSK.nextval, 'Läuse', 3, to_date('22.04.2017',
'DD.MM.YYYY'));

INSERT INTO krankheit VALUES(S_TIERARZT_IDSK.nextval, 'Beinbruch', 2, to_date('05.05.2017',
'DD.MM.YYYY'));

INSERT INTO krankheit VALUES(S_TIERARZT_IDSK.nextval, 'Gräberallergie', 6, to_date('20.06.2017',
'DD.MM.YYYY'));

INSERT INTO krankheit VALUES(S_TIERARZT_IDSK.nextval, 'Hodenkrebs', 6, to_date('20.06.2017',
'DD.MM.YYYY'));

insert into krankheit values (S_TIERARZT_IDSK.nextVal, 'keine Krankheit', 6, to_date('27.06.2017',
'DD.MM.YYYY'));
```

```
select * from krankheit;
```

```
INSERT INTO behandlung VALUES(S_TIERARZT_IDSB.nextval, 'Hodenkrebs-Einschläferung',
to_date('20.06.2017', 'DD.MM.YYYY'), 2);

INSERT INTO behandlung VALUES(S_TIERARZT_IDSB.nextval, 'Nierenprellung-Kontrolle',
to_date('30.05.2017', 'DD.MM.YYYY'), 3);

INSERT INTO behandlung VALUES(S_TIERARZT_IDSB.nextval, 'Gehirnerschütterung-Spritze',
to_date('03.01.2017', 'DD.MM.YYYY'), 4);

INSERT INTO behandlung VALUES(S_TIERARZT_IDSB.nextval, 'Gehirnerschütterung-Spritze',
to_date('10.01.2017', 'DD.MM.YYYY'), 5);

INSERT INTO behandlung VALUES(S_TIERARZT_IDSB.nextval, 'Läuse-Spritze', to_date('22.04.2017',
'DD.MM.YYYY'), 5);

INSERT INTO behandlung VALUES(S_TIERARZT_IDSB.nextval, 'Beinbruch-Eingipsen',
to_date('05.02.2017', 'DD.MM.YYYY'), 6);

INSERT INTO behandlung VALUES(S_TIERARZT_IDSB.nextval, 'Beinbruch-Entgipsen',
to_date('06.03.2017', 'DD.MM.YYYY'), 6);

INSERT INTO behandlung VALUES(S_TIERARZT_IDSB.nextval, 'Hodenkrebs-Einschläferung',
to_date('20.06.2017', 'DD.MM.YYYY'), 2);

INSERT INTO behandlung VALUES(S_TIERARZT_IDSB.nextval, 'Hodenkrebs-Einschläferung',
to_date('20.06.2017', 'DD.MM.YYYY'), 2);
```

```
select tname "Tiername", krankheit.datum "Krankheit-Datum", krankheit.bezeichnung "Krankheit",
behandlung.datum "Behandlung-datum",

behandlung.bezeichnung "Behandlung", bdatum "Geburtsdatum", sdatum "Sterbedatum", bname
"Besitzername", anschrift
```

from besitzer

inner join tier on idBesitzer = idB

inner join krankheit on idT = idTier

inner join behandlung on idK = idKrankheit;

TRIGGER

Trigger 1

```
CREATE OR REPLACE TRIGGER T_TIERARZT_INSERT_TIER
BEFORE INSERT ON TIER
FOR EACH ROW
DECLARE Tcount NUMBER := 0;
BEGIN
    SYS.DBMS_OUTPUT.PUT_LINE('hallo');
    SELECT COUNT(*) INTO Tcount FROM besitzer
    INNER JOIN tier ON besitzer.IDB = tier.IDBESITZER
    WHERE besitzer.idb = :new.idBesitzer;

    if tcount > 3 then
        raise_application_error(-20010, 'Dieser Besitzer hat schon 3 Tiere!!');
    end if;
END;
```

Trigger 2

```
CREATE OR REPLACE TRIGGER T_TIERARZT_BEHANDLUNG2
BEFORE INSERT OR UPDATE ON BEHANDLUNG
for each row
declare Bcount Number := 0;
BEGIN
    select count(*) into bcount from behandlung
```

```
inner join krankheit on behandlung.IDKRANKHEIT = krankheit.IDK
inner join tier on krankheit.IDTIER = tier.IDT
where tier.sdatum < :new.DATUM;

if bcount > 0 then
    raise_application_error(-20010, 'Des Tier ist nicht mehr am been!!');
end if;
END;
```

Trigger 3

```
CREATE OR REPLACE TRIGGER T_TIERARZT_BEHANDLUNG
BEFORE INSERT OR UPDATE ON KRANKHEIT
for each row
declare Bcount Number := 0;
BEGIN
    select count(*) into bcount from krankheit
    inner join tier on krankheit.IDTIER = tier.IDT
    where tier.sdatum < :new.DATUM;

    if bcount > 0 then
        raise_application_error(-20010, 'Des Tier ist nicht mehr am been!!');
    end if;
END;
```

Trigger 4

```
create or replace TRIGGER T_TIERAZT_BEHANDLUNGBEITOT
BEFORE UPDATE ON TIER
FOR EACH ROW
declare Banzahl number := 0;
BEGIN
    select count(*) into banzahl from tier
```

```
inner join krankheit on tier.IDT = krankheit.IDTIER
```

```
inner join behandlung on krankheit.IDK = behandlung.IDKRANKHEIT
```

```
where krankheit.DATUM > tier.SDATUM or behandlung.DATUM > tier.SDATUM;
```

```
if Banzahl > 0 then
```

```
    raise_application_error(-20010, 'Dieses Tier hat noch Behandlungen!!');
```

```
end if;
```

```
END;
```