SQL Data Definition Language: Exercises

1. Which of the following table definitions are valid? Where invalid, explain why.

```sql
create table Stuff1 (
    name text primary key,
    number int,
    rating float not null
);
```

```sql
create table Stuff2 (
    name varchar(25) primary key,
    number int primary key,
    rating float
);
```

```sql
create table Stuff3 (
    name text primary key,
    number int unique default 0,
    rating float
);
```

```sql
create table Stuff4 (
    name char(30) unique,
    number int unique,
    rating real
);
```

2. Suppose we have defined this table:

```sql
create table Fluff (
    this int,
    that int,
    other text unique,
    primary key (this, that)
);
```

Which of the following is valid? (Consider each as if it were being applied to any empty instance of the table.) For each that is invalid, identify the problem.

```sql
insert into Fluff values (1, 2, 'my'), (1, 2, 'night');
insert into Fluff values (11, 22, 'twinkle'), (33, 44, 'twinkle');
insert into Fluff values (100, 5, 'night'), (100, 10, 'my');
insert into Fluff values (null, null, 'oh');
insert into Fluff values (5, null, 'uh');
insert into Fluff values (null, 20, 'a'), (null, 21, 'b');
insert into Fluff values (80, 81, null);
insert into Fluff values (90, 91, null), (92, 93, null);
```
3. Again, suppose we have defined this table:

```sql
create table Fluff (
    this int,
    that int,
    other text unique,
    primary key (this, that)
);
```

Which of these table definitions is valid, given the definition of table Fluff? Where invalid, explain why.

```sql
create table Nonsense1 (  
a int,
b int,
foreign key (b) references Fluff(this)
);
```

```sql
create table Nonsense2 (  
a int,
b text references Fluff(other)
);
```

```sql
create table Nonsense3 (  
a int,
b int,
c int,
foreign key (b, c) references Fluff
);
```

```sql
create table Nonsense4 (  
a int references Fluff(blah),
b int
);
```

4. Can you think of any other ways that an attempt to define a foreign key could fail?