CS1555 Recitation 3 Solution

Objective: To practice SQL DDL (schema evolution)

Notes:

1. You can use `describe <table_name>;` to validate attribute type changes.
2. You can use insert statements to validate attribute constraint changes.

Questions:

1. Create a table "t1" with a primary key attribute "a1" of type varchar2(10).
   (Before creating the table, please use drop table statement to avoid pre-existing tables with the same name)

```
drop table t1 cascade constraints;
create table t1 (
        a1 varchar2(10) primary key
);
```

2. Add a new column called "a2" of type varchar2(5) to "t1".

```
alter table t1 add a2 varchar2(5);
```

3. Modify the length of "a2" to be 10.

```
alter table t1 modify a2 varchar2(10);
```

4. Modify "a2" to be of type number(5);

```
alter table t1 modify a2 number(5);
```

5. Modify "a2" to be of type number(10, 5);

```
alter table t1 modify a2 number(10, 5);
```

6. Change the schema so that "a2" cannot be null.

```
alter table t1 modify a2 not null;
```

7. Change the schema so that "a2" can be null again.

```
alter table t1 modify a2 null;
```

8. Change the default value of "a2" to be 1.

```
alter table t1 modify a2 default 1;
```

9. Remove the default value of "a2".

```
alter table t1 modify a2 default null;
```

10. Change the schema so that "a2" must be unique.

```
alter table t1 add constraint t1_unqi_a2 unique (a2);
```

11. Change the schema so that "a2" doesn't need to be unique.

```
alter table t1 drop constraint t1_uniq_a2;
```

12. Add a range check to "t1" so that the value of "a2" must be greater or equal to 1 and less or equal to 10.

```
alter table t1 add constraint t1_a2_range check (a2 between 1 and 10);
```

13. Modify the range check so that the value of "a2" must be greater or equal to 1 and less or equal to 5.

```
alter table t1 drop constraint t1_a2_range;
alter table t1 add constraint t1_a2_range check (a2 between 1 and 5);
```

14. Add a check so that the value of "a2" must be in the set of {1,2,3,4,5}.

```
alter table t1 add constraint t1_a2_in check (a2 in (1,2,3,4,5));
```

15. Create a table "t2" with a primary key attribute "b1" of type number(10,5).
(Before creating the table, please use drop table statement to avoid pre-existing tables with the same name)

```
drop table t2 cascade constraints;
create table t2 (
        b1 varchar2(10) primary key
);
```

16. Add a foreign key constraint to "a2" so that "a2" refers to "b1" in table "t2".

```
alter table t1 add constraint t1_fk_t2 foreign key (a2) references t2(b1);
```

17. Try to insert a tuple into "t1" with values ("pitt01", 5).

```
insert into t1 (a1, a2) values ('pitt01', 1);
```

This will not work because the parent key in t2 cannot be found.

18. Try to drop table "t2".

```
drop table t2;
```

This will not work because attribute b1 in table t2 has a foreign key constraint, namely t1_fk_t2 referring to it.

19. Drop table "t2" with "cascade constraints" option.

```
drop table t2 cascade constraints;
```

This will work because the "cascade constraints" option will allow the drop statement also delete any constraints referring to any attributes in the dropped table.

20. Add a unique constraint on (a1, a2) and try to drop column "a2".

```
alter table t1 add constraint t1_uniq_a1_a2 unique (a1, a2);
alter table t1 drop column a2;
```

This will not work because the attribute a2 has a multi-column constraint, namely t1_uniq_a1_a2 referring to it.

21. Drop column "a2" with "cascade constraints" option.

```
alter table t1 drop column a2 cascade constraints;
```

This will work because the "cascade constraints" option will allow the drop statement also delete any constraints referring to the dropped attribute.