SQL Database for Beginners

Lab Exercise 11

Answer Key

1) Write a query creating a view projectstaffing returning staff details with all records in the projectperson table.

Answer:

Tip:

Before creating the view write and test the query:

```
select *
from projectperson pp, person p
where pp.pid=p.id
```

Now you can create the view from the above

```
create view projectstaffing as
select *
from projectperson pp, person p
where pp.pid=p.id
```

Finally test the view:

```
select * from projectstaffing
```

2) Write a query creating a table comments . The purpose of the table is to record notes by means of freetext associated with a modification timestamp.

Answer:

```
create table comments
(
  id int(11) not null,
  notes text,
  modified timestamp not null,
  primary key (id)
)
```

3) Write a query returning all records in the table comments.

Answer:

```
select * from comments
```

Note: The result of this query obviously is empty as no records have been inserted into the table.

4) Write a query adding a trigger to the table comments that sets the modified value to the current

timestamp on creation of any records.

Answer:

```
delimiter @@
create trigger commentcreated before insert on comments
for each row begin
   set new.modified=now();
end;
@@
delimiter ;
```

5) Write a query adding a record to the table comments and verify that the trigger is working correctly.

Answer:

```
insert into comments(notes)
values ('inserting a new note')
```

Post-test with:

```
select * from comments
```

This should show the new record including primary key and timestamp

6) Write a query adding a trigger to the table comments that sets the modified value to the current timestamp on each modification of any records.

Answer:

```
delimiter @@
create trigger commentmodified before update on comments
for each row begin
   set new.modified=now();
end;
@@
delimiter ;
```

7) Write a query modifying the previously added record in the table comments and verify that the trigger is working correctly.

Answer:

```
update comments
set notes='updating a note'
where id=1
```

Post-test with:

```
select * from comments
```

This should show the modified record including primary key and modified timestamp.

8) Write queries removing the table comments as well as the associated triggers.

Answer:

```
drop trigger commentcreated
drop trigger commentmodified
drop table comments
```

Note: The triggers might automatically be removed when dropping the table, however it's better to be explicit.

9) Write a query creating a stored procedure addproject. The purpose of the procedure is to add a new project to the project table using the submitted name and budget.

Answer:

```
delimiter @@
create procedure addproject(p_name varchar(255), p_budget double)
begin
  insert into project (label, budget)
  values(p_name, p_budget);
end;
@@
delimiter ;
```

Note: This is a very basic solution lacking any error handling, etc.

10) Write queries adding a new project "SEO" with a budget of 400 using the above stored procedure and verify that the stored procedure works correctly.

Answer:

```
call addproject('SEO', 400);
```

Post-test with:

```
select * from project
```

This should show the additional project.