

In the name of God  
Database Lab  
Spring 2016

Iman TABRIZIAN

April 27, 2016

Date Performed: April 6, 2016  
Partners: Parham Alvani

## 1 Results and Conclusions

- a. 

```
create table payment_1 (  
    payment_number int ,  
    payment_date datetime,  
    payment_amount int ,  
    loan_number int ,  
    primary key (payment_number, loan_number),  
    foreign key (loan_number) references loan_1(loan_number)  
);
```
- b. 

```
CREATE PROCEDURE get_balance @name varchar(255),@balance int output,@account_number int  
select @name = account_1.balance,@account_number = account_1.account_number  
from depositor_1,account_1, customer_1  
where depositor_1.account_id = customer_1.customer_id and  
account_1.account_id = customer_1.customer_id and customer_1.customer_name = @name;
```
- c. 

```
create procedure get_payment @payment_number int, @loan_number int, @branch varchar(255)  
select @branch = loan_1.branch_name from loan_1, payment_1 where @payment_number = payment_1.payment_number and  
payment_1.loan_number = @loan_number;
```
- d. 

```
create procedure get_info @recep_id int as  
    WAITFOR DELAY '00:00:10';  
select * from customer_1 where customer_1.customer_id = @recep_id;
```
- e. 

```
create function get_max_intrest ()  
returns int  
as begin  
    declare @max int  
    select @max = account_1.account_number  
    from account_1  
    where account_1.account_number = (select max(account_number) from account_1);
```

```

from account_1 where max(account_1.intrest_rate) = account_1.intrest_rate;
return @max
end

f. create function get_department (@employee_id int)
returns varchar(255)
as begin
declare @var varchar(255)

select @var=employee_1.department_name from employee_1.employee_id;

return @var
end

g. create trigger create_log_insert_real on payment_1
after insert as
begin
print 'Inserted into payment'
end

h. alter trigger create_log_delete on payment_1
after delete as
begin
insert into payment_1_log values (deleted,'deleted');
end

i. create trigger update_name on branch_1
after update as
begin
if(update(branch_name))
begin
rollback
end
end

j. create trigger create_log_insert on payment_1
after update as
begin
print 'updated payment_1'
end

k. create table payment_1_log (
    payment_number int ,
    payment_date datetime,
    payment_amount int ,
    loan_number int ,
    log_type varchar(255)
);

```