AMERICAN INTERNATIONAL UNIVERSITY - BANGLADESH
Faculty of Science and Information Technology

Project Name: HR Management System

Course Name: Advanced Database Management System

Course Instructor: Juena Ahmed Noshin

Section: D

Semester: 2017-2018, Summer

Group Members:

1. SAKIB,AHMED SHAHRIAR (16-32170-2)

2. IMAM,RUBAIYA (16-31753-1)

3. SIDDIQUI,IFTEKHAR HASAN (16-31815-1)

4. TUHIN,MAMUN UR RASHID (14-27353-2)

Submission Date: 31st August, 2018

Introduction:

Human resource management systems' database is designed to manage payroll, attendance, leave management and staff personal information.

The aim of this project is to design such database which allows the admin to streamline their human resource tasks and manage their employee in a more effective and efficient way.

The system will ensure effective utilization and maximum development of human resource, generate and maintain human resource records and allow proper interactions and timely access to accurate information to those who require the information.

Project Proposal:

This database will be able to manage employee information by automating core HR, benefits and payroll processes for increased efficiency and productivity. It will also reduce the time of computation taken between process by timely generating the necessary reports and statistics. This database will reduce redundant data and error scope by easily creating accurate reporting and analysis. This database will ensure the security of employee information. The aim of creating such a database is to reduce routine administration and promote a paperless environment.

It merges human resources as a discipline and, in particular, it's basic HR activities and processes with the information technology field. The linkage of its financial and human resource modules through one database is the most important distinction to the individually and proprietarily developed predecessors.

Human Resource Information Systems provide a means of acquiring, storing, analyzing and distributing information to various stakeholders. HRIS enable improvement in traditional processes and enhance strategic decision-making. The wave of technological advancement has revolutionized each and every space of life today, and HR in its entirety was not left untouched.

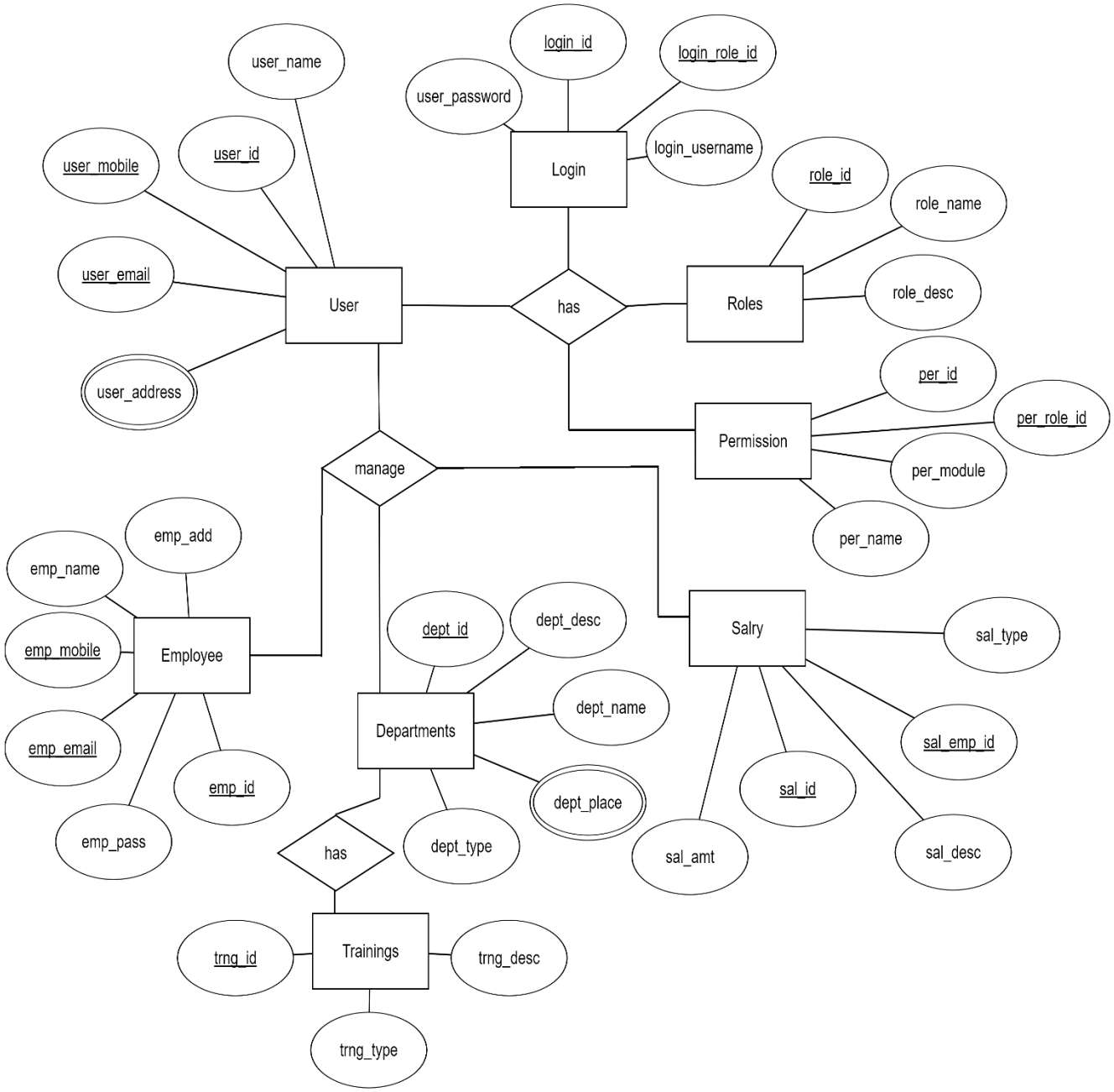
Early systems were narrow in scope, typically focused on a single task, such as improving the payroll process or tracking employees' work hours. This system covers maximum spectrum of tasks associated with Human Resources departments, including tracking & improving process efficiency, managing organizational hierarchy, and simplifying financial transactions of all types

Scenario Description:

This **Human Resource Management System** database is a form of database that combines a number of processes to ensure the easy management of human resources, business processes and data. Human Resources Software will be used by businesses to combine a number of necessary HR functions, such as storing employee data, managing payrolls, recruitment processes, benefits administration and keeping track of attendance records. It ensures everyday Human Resources processes are manageable and easy to access.

In this system there are two types of users, admin and employee. Admin will have maximum privileges regarding to the database. An employee can login to the system by using their given employee ID and their password. The admin can assign employees to specific roles, departments and also with various trainings. The admin can edit, create, delete or update all the information related to trainings, departments, roles, salaries and employee information. The trainings are under consideration of each department. An employee can show all the informations related to them and also get notified for their assigned trainings. Admin can also add new trainings, departments, roles and employee to the database.

ER Diagram


user_name
login_id
login_role_id
user_password
user_id
Login
login_username
user_mobile
role_id
role_name
user_email
User
has
Roles
role_desc
user_address
per_id
per_role_id
Permission
manage
per_module
per_name
emp_add
emp_name
dept_id
dept_desc
Salry
sal_type
emp_mobile
Employee
Departments
dept_name
emp_email
sal_emp_id
emp_id
sal_id
dept_place
emp_pass
has
dept_type
sal_amt
sal_desc
trng_id
Trainings
trng_desc
trng_type

NORMALIZATION

- **Login_User**

1NF

user_id	user_name	user_mobile	user_email	user_address	login_id	user_password	login_username

2NF

User_Info

user_id	user_name	user_mobile	user_email	user_address

Login_Info

login_id	user_id	login_username	user_password

3NF

User_Info

user_id	user_name	user_mobile	user_email	user_address

Login_Info

login_id	user_id	login_username	user_password

- **Login_Role**

1NF

login_id	login_username	user_password	Role_id	Role_name	Role_desc

2NF

Login_Info

login_id	role_id	login_username	user_password

Roles_Info

role_id	role_name	role_desc

3NF

Login_Info

login_id	role_id	login_username	user_password

Roles_Info

role_id	role_name	role_desc

- **Role_Permission**

1NF

role_id	Role_name	Role_desc	per_id	per_name	Per_module

2NF

Permission_Info

per_id	role_id	per_name	per_module

Roles_Info

role_id	role_name	role_desc

3NF

Permission_Info

per_id	role_id	per_name	per_module

Roles_Info

role_id	role_name	role_desc

- **Employee_Departments**

1NF

emp_id	emp_pass	emp_email	emp_mobile	emp_name	emp_add	dept_id	dept_desc	dept_name	dept_place	dept_type

2NF

Employee_INFO

emp_id	emp_pass	emp_email	emp_mobile	emp_name	emp_add	dept_id

3NF

Employee_INFO

emp_id	emp_pass	emp_email	emp_mobile	emp_name	emp_add	dept_id

- **Department_Info**

3NF

dept_id	dept_desc	dept_name	dept_place	dept_type

- **Departments_Trainings**

1NF

dept_id	dept_desc	dept_name	dept_place	dept_type	trng_id	trng_desc	trng_type

2NF

Department_Info

dept_id	dept_desc	dept_name	dept_place	dept_type

Trainings_info

trng_id	trng_desc	trng_type	dept_id

3NF

Department_Info

dept_id	dept_desc	dept_name	dept_place	dept_type

Trainings_info

trng_id	trng_desc	trng_type	dept_id

- **Employee_Salary**

1NF

emp_id	emp_pass	emp_email	emp_mobile	emp_name	emp_add	sal_id	sal_amt	sal_desc	sal_type

2NF

Employee_INFO

emp_id	emp_pass	emp_email	emp_mobile	emp_name	emp_add

Sal_Info

sal_id	emp_id	sal_amt	sal_desc	sal_type

3NF

Employee_INFO

emp_id	emp_pass	emp_email	emp_mobile	emp_name	emp_add

Sal_Info

sal_id	emp_id	sal_amt	sal_desc	sal_type

Schema Diagram

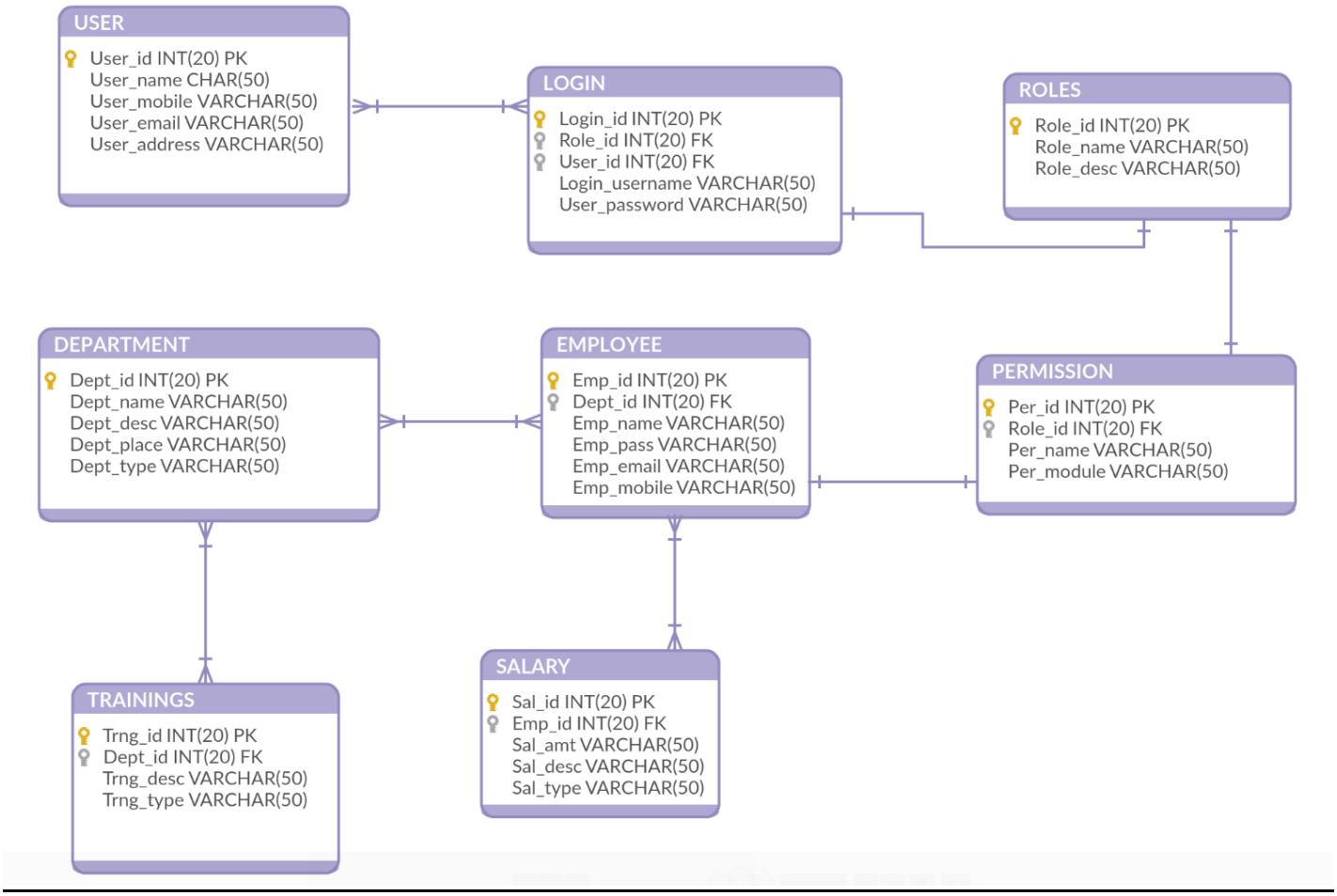


Table creation

create table login_info (login_id int not null,login_username varchar(50),user_password varchar(50),primary key(login_id))

create table User_Info(user_id int not null, user_name varchar(50), user_mobile varchar(50),user_email varchar(50), user_address varchar(50),primary key(user_id))

create table roles_Info(role_id int not null, role_name varchar(50),role_desc varchar(50),primary key(role_id))

create table permission_Info(per_id int not null,per_name varchar(50),per_module varchar(50),primary key(per_id))

create table employee_info

(emp_id int not null,emp_pass varchar(50),emp_email varchar(50),emp_mobile varchar(50),primary key(emp_id))

create table department_Info

(dept_id int not null, dept_desc varchar(50),dept_name varchar(50),dept_place varchar(50),dept_type varchar(50),primary key(dept_id))

create table trainings_info

(trng_id int not null,trng_desc varchar(50),trng_type varchar(50),primary key(trng_id))

create table sal_Info

(sal_id int not null,sal_amt varchar(50),sal_desc varchar(50),sal_type varchar(50),primary key(sal_id))

alter table login_info add (user_id int not null,role_id int not null,foreign key (user_id) references user_info(user_id),foreign key (role_id) references roles_info(role_id))

```
alter table permission_Info add (role_id int not null,foreign key (role_id) references
roles_info(role_id))

alter table employee_info add (dept_id int not null,foreign key (dept_id) references
department_Info(dept_id))

alter table trainings_info add (dept_id int not null,foreign key (dept_id) references
department_Info(dept_id))

alter table sal_Info add (emp_id int not null,foreign key (emp_id) references
employee_info(emp_id))
```

DATA INSERTION

User_Info

insert into User_Info(user_id, user_name,user_mobile,user_email,user_address) values (1,'rubu','09876543','rubu.aiub@gmail.com','uttara,dhaka')

insert into User_Info(user_id, user_name,user_mobile,user_email,user_address) values (2,'ifty','098766543','ifty.aiub@gmail.com','uttara,dhaka')

insert into User_Info(user_id, user_name,user_mobile,user_email,user_address) values (3,'sakib','09876574843','sakib.aiub@gmail.com','mirpur,dhaka')

insert into User_Info(user_id, user_name,user_mobile,user_email,user_address) values (4,'efaz','076543','efaz.aiub@gmail.com','uttara,dhaka')

insert into User_Info(user_id, user_name,user_mobile,user_email,user_address) values (5,'kala','09906543','laka.aiub@gmail.com','banani,dhaka')

select * from User_Info

USER_ID	USER_NAME	USER_MOBILE	USER_EMAIL	USER_ADDRESS
1	rubu	09876543	rubu.aiub@gmail.com	uttara,dhaka
2	ifty	098766543	ifty.aiub@gmail.com	uttara,dhaka
3	sakib	09876574843	sakib.aiub@gmail.com	mirpur,dhaka
4	efaz	076543	efaz.aiub@gmail.com	uttara,dhaka
5	kala	09906543	laka.aiub@gmail.com	banani,dhaka

roles_Info

insert into roles_Info(role_id,role_name,role_desc) values (1,'efaz','iyegfi3vihveivh31rpih')

insert into roles_Info(role_id,role_name,role_desc) values (2,'manager','iyegfi3vihveivh31rpih')

insert into roles_Info(role_id,role_name,role_desc) values (3,'intern','iyegfi3vihveivh31rpih')

insert into roles_Info(role_id,role_name,role_desc) values (4,'boss','iyegfi3vihveivh31rpih')

insert into roles_Info(role_id,role_name,role_desc) values (5,'DM','iyegfi3vihveivh31rpih')

select * from roles_Info

ROLE_ID	ROLE_NAME	ROLE_DESC
1	efaz	iyegfi3vihveivh31rpih
2	manager	iyegfi3vihveivh31rpih
3	intern	iyegfi3vihveivh31rpih
4	boss	iyegfi3vihveivh31rpih
5	DM	iyegfi3vihveivh31rpih

department_Info

insert into department_Info(dept_id,dept_desc,dept_name,dept_place,dept_type) values(1,'sales blah','sales','uttara','blah')

insert into department_Info(dept_id,dept_desc,dept_name,dept_place,dept_type) values(2,'accounting blah','accounting ','uttara','blah')

insert into department_Info(dept_id,dept_desc,dept_name,dept_place,dept_type) values(3,'marketing blah','marketing ','uttara','blah')

insert into department_Info(dept_id,dept_desc,dept_name,dept_place,dept_type) values(4,'finance blah','finance ','uttara','blah')

insert into department_Info(dept_id,dept_desc,dept_name,dept_place,dept_type) values(5,'admin blah','adminh ','uttara','blah')

select * from department_Info

DEPT_ID	DEPT_DESC	DEPT_NAME	DEPT_PLACE	DEPT_TYPE
1	sales blah	sales	uttara	blah
3	marketing blah	marketing	uttara	blah
2	accounting blah	accounting	uttara	blah
4	finance blah	finance	uttara	blah
5	admin blah	adminh	uttara	blah

login_Info

insert into login_Info(login_id,role_id,user_id,login_username,user_password) values (1,1,1,'rubu','rubu')

insert into login_Info(login_id,role_id,user_id,login_username,user_password) values (2,2,2,'ifty','ifty')

insert into login_Info(login_id,role_id,user_id,login_username,user_password) values (3,3,3,'sakib','sakib')

insert into login_Info(login_id,role_id,user_id,login_username,user_password) values (4,4,4,'efaz','efaz')

insert into login_Info(login_id,role_id,user_id,login_username,user_password) values (5,5,5,'kala','kala')

select * from login_Info

LOGIN_ID	LOGIN_USERNAME	USER_PASSWORD	USER_ID	ROLE_ID
1	rubu	rubu	1	1
2	ifty	ifty	2	2
3	sakib	sakib	3	3
4	efaz	efaz	4	4
5	kala	kala	5	5

permission_Info

insert into permission_Info(per_id,role_id,per_name,per_module) values (1,1,'lkasdhfg','jhrf')

insert into permission_Info(per_id,role_id,per_name,per_module) values (2,2,'lkdhfg','jhrf')

insert into permission_Info(per_id,role_id,per_name,per_module) values (3,3,'lkdhfg','jhrf')

insert into permission_Info(per_id,role_id,per_name,per_module) values (4,4,'lkdhfg','jhrf')

insert into permission_Info(per_id,role_id,per_name,per_module) values (5,5,'lkasdg','jhrf')

select * from permission_Info

PER_ID	PER_NAME	PER_MODULE	ROLE_ID
1	lkasdhfg	jhrf	1
2	lkdhfg	jhrf	2
3	lkdhfg	jhrf	3
4	lkdhfg	jhrf	4
5	lkasdg	jhrf	5

Employee_INFO

insert into employee_info(emp_id,dept_id ,emp_pass,emp_email,emp_mobile) values (1,1,'rubu','rubu.aiub@gmail.com','09876543')

insert into employee_info(emp_id,dept_id ,emp_pass,emp_email,emp_mobile) values (2,2,'ifty','ifty.aiub@gmail.com','09876543')

insert into employee_info(emp_id,dept_id ,emp_pass,emp_email,emp_mobile) values (3,3,'sakib','sakib.aiub@gmail.com','09876543')

insert into employee_info(emp_id,dept_id ,emp_pass,emp_email,emp_mobile) values (4,4,'efaz','efaz.aiub@gmail.com','09876543')

insert into employee_info(emp_id,dept_id ,emp_pass,emp_email,emp_mobile) values (5,5,'kala','laka.aiub@gmail.com','09876543')

select * from employee_info

EMP_ID	EMP_PASS	EMP_EMAIL	EMP_MOBILE	DEPT_ID
1	rubu	rubu.aiub@gmail.com	09876543	1
2	ifty	ifty.aiub@gmail.com	09876543	2
3	sakib	sakib.aiub@gmail.com	09876543	3
4	efaz	efaz.aiub@gmail.com	09876543	4
5	kala	laka.aiub@gmail.com	09876543	5

Training_info

insert into trainings_info(trng_id,trng_desc,trng_type,dept_id) values (1,'uttara','sales',1)

insert into trainings_info(trng_id,trng_desc,trng_type,dept_id) values (2,'uttara','accounting',2)

insert into trainings_info(trng_id,trng_desc,trng_type,dept_id) values (3,'mirpur','marketting',3)

insert into trainings_info(trng_id,trng_desc,trng_type,dept_id) values (4,'uttara','finance',4)

insert into trainings_info(trng_id,trng_desc,trng_type,dept_id) values (5,'banani','adminh',5)

select * from trainings_info

TRNG_ID	TRNG_DESC	TRNG_TYPE	DEPT_ID
1	uttara	sales	1
2	uttara	accounting	2
3	mirpur	marketting	3
4	uttara	finance	4
5	banani	adminh	5

Sal_Info

insert into sal_Info(sal_id,emp_id,sal_amt,sal_desc,sal_type) values (1,1,'500000','saleshu','bdt')

insert into sal_Info(sal_id,emp_id,sal_amt,sal_desc,sal_type) values (2,2,'50000','saleshu','bdt')

insert into sal_Info(sal_id,emp_id,sal_amt,sal_desc,sal_type) values (3,3,'50000','saleshu','bdt')

insert into sal_Info(sal_id,emp_id,sal_amt,sal_desc,sal_type) values (4,4,'500','saleshu','bdt')

insert into sal_Info(sal_id,emp_id,sal_amt,sal_desc,sal_type) values (5,5,'50','saleshu','bdt')

select * from sal_Info

SAL_ID	SAL_AMT	SAL_DESC	SAL_TYPE	EMP_ID
1	500000	saleshu	bdt	1
2	50000	saleshu	bdt	2
3	50000	saleshu	bdt	3
4	500	saleshu	bdt	4
5	50	saleshu	bdt	5

Group Function

1. select max(sal_amt) from sal_Info

Results Explain Describe Saved SQL History

MAX(SAL_AMT)
500000

1 rows returned in 0.02 seconds CSV Export

2. select sum(sal_amt) from sal_Info

Results Explain Describe Saved SQL History

SUM(SAL_AMT)
600550

1 rows returned in 0.00 seconds CSV Export

3. select round(avg(sal_amt),2) from sal_Info

Results Explain Describe Saved SQL History

ROUND(AVG(SAL_AMT),2)
120110

1 rows returned in 0.00 seconds CSV Export

Single Row Function

1. SELECT emp_id, salary, NVL (commission_pct,0) FROM sal_info

2. SELECT UPPER (trng_type),LOWER (trng_desc) FROM training_info

3. SELECT CONCAT (trng_type, trng_desc) FROM training_info

Subquery

1. select sal_id,sal_amt from sal_Info where sal_amt=(select max(sal_amt) from sal_Info)

Results Explain Describe Saved SQL History

SAL_ID	SAL_AMT
1	500000

1 rows returned in 0.00 seconds CSV Export

---

2. select sal_id,sal_amt from sal_Info where sal_amt=(select min(sal_amt) from sal_Info)

Results Explain Describe Saved SQL History

SAL_ID	SAL_AMT
5	50

1 rows returned in 0.00 seconds CSV Export

3. select sal_id,sal_amt from sal_Info where sal_amt>(select min(sal_amt) from sal_Info)

Results Explain Describe Saved SQL History

SAL_ID	SAL_AMT
1	500000
2	50000
3	50000
4	500

4 rows returned in 0.00 seconds CSV Export

Joining

1. select user_name,login_username from login_info,user_info where login_info.user_id=user_info.user_id

Results Explain Describe Saved SQL History

USER_NAME	LOGIN_USERNAME
rubu	rubu
ifty	ifty
sakib	sakib
efaz	efaz
kala	kala

5 rows returned in 0.02 seconds CSV Export

2. select emp_email,dept_place from employee_info,department_info where employee_info.dept_id=department_info.dept_id

Results Explain Describe Saved SQL History

EMP_EMAIL	DEPT_PLACE
rubu.aiub@gmail.com	uttara
sakib.aiub@gmail.com	uttara
ifty.aiub@gmail.com	uttara
efaz.aiub@gmail.com	uttara
laka.aiub@gmail.com	uttara

5 rows returned in 0.00 seconds CSV Export

3. select emp_email,dept_name from employee_info,department_info where employee_info.dept_id=department_info.dept_id

Results Explain Describe Saved SQL History

EMP_EMAIL	DEPT_NAME
rubu.aiub@gmail.com	sales
sakib.aiub@gmail.com	marketing
ifty.aiub@gmail.com	accounting
efaz.aiub@gmail.com	finance
laka.aiub@gmail.com	adminh

5 rows returned in 0.00 seconds CSV Export

PL/SQL QUERIES

FUNCTION

1.
```
CREATE OR REPLACE FUNCTION totalEMP
RETURN number IS
      total number(2) := 0;
BEGIN
SELECT count(*) into total
FROM employee;

RETURN total;
END;
DECLARE
  c number(12);
BEGIN
  c := totalEMP();
  dbms_output.put_line('Total No of employees: ' || c);
END;
```

2.
```
CREATE OR REPLACE FUNCTION totaldept
RETURN number IS
      total number(2) := 0;
BEGIN
SELECT count(*) into total
FROM department;

RETURN total;
END;
DECLARE
  c number(12);
BEGIN
  c := totaldept();
  dbms_output.put_line('Total No of departments: ' || c);
END;
```

PROCEDURE

1.
```
CREATE OR REPLACE PROCEDURE greetings
AS
BEGIN
   dbms_output.put_line('Welcome to work!');
END;
EXECUTE greetings;
BEGIN
   greetings;
END;
```

2.
```
CREATE OR REPLACE PROCEDURE error
AS
BEGIN
   dbms_output.put_line('oops something is wrong!');
END;
EXECUTE error;
BEGIN
   error;
END;
```

RECORD

1.
```
DECLARE
   user_rec user_info%rowtype;
BEGIN
   SELECT * into user_rec
   FROM user
   WHERE id = 2;
   dbms_output.put_line('Mobile: ' || user_rec.user_mobile);
   dbms_output.put_line('Name: ' || user_rec.user_name);
   dbms_output.put_line('Address: ' || user_rec.user_address);
END;
```
2.
```
DECLARE
   emp_rec emp_info%rowtype;
BEGIN
   SELECT * into emp_rec
   FROM employee
   WHERE id = 2;
   dbms_output.put_line('Mobile: ' || emp_rec.emp_mobile);
   dbms_output.put_line('Name: ' || emp_rec.emp_name);
   dbms_output.put_line('Email: ' || emp_rec.emp_email);
END;
```

CURSOR

1. DECLARE

```
   total_rows number(2);
BEGIN
   UPDATE salary
   SET sal_amt = sal_amt + 500;
   IF sql%notfound THEN
      dbms_output.put_line('no salary selected');
   ELSIF sql%found THEN
      total_rows := sql%rowcount;
      dbms_output.put_line( total_rows || ' salary selected ');
   END IF;
END;
```

2. DECLARE

```
   total_rows number(2);
BEGIN
   UPDATE salary
   SET sal_amt = sal_amt - 500;
   IF sql%notfound THEN
      dbms_output.put_line('no salary selected');
   ELSIF sql%found THEN
      total_rows := sql%rowcount;
      dbms_output.put_line( total_rows || ' salary selected ');
   END IF;
END;
```

TRIGGER

1.
```
CREATE OR REPLACE TRIGGER display_salary_changes
BEFORE DELETE OR INSERT OR UPDATE ON salary
FOR EACH ROW
WHEN (NEW.ID > 0)
DECLARE
   sal_diff number;
BEGIN
   sal_diff := :NEW.salary  - :OLD.salary;
   dbms_output.put_line('Old salary: ' || :OLD.salary);
   dbms_output.put_line('New salary: ' || :NEW.salary);
   dbms_output.put_line('Salary difference: ' || sal_diff);
END;
```

2.
```
CREATE OR REPLACE TRIGGER user_deleted
after DELETE ON user
FOR EACH ROW
BEGIN
  dbms_output.put_line('A user Deleted');
END;
/
select * from user ;
DELETE FROM user WHERE  user_id=4;
```

PACKAGE

1. CREATE PACKAGE user_package AS

   PROCEDURE display_b_name(user_id user.b_id%TYPE);

   END user_package;

   /

   CREATE OR REPLACE PACKAGE BODY user_package AS

   PROCEDURE display_b_name(user_id user.b_id%TYPE) IS

   user_name user.b_name%TYPE;

   BEGIN

   SELECT b_name INTO user_name

   FROM user

   WHERE b_id = user_id;

   dbms_output.put_line(user Name: '|| user_name);

   END display_b_name;

   END user_package;

   /

   begin

   user_package.display_b_name('4');

   end

2. CREATE PACKAGE emp_package AS

   PROCEDURE display_e_name(employee_id employee.e_id%TYPE);

   END employee _package;

   /

   CREATE OR REPLACE PACKAGE BODY employee _package AS

   PROCEDURE display_e_name(employee _id employee.e_id%TYPE) IS

```
    employee_name us employee er.e_name%TYPE;
  BEGIN
    SELECT e_name INTO employee_name
    FROM user
    WHERE e_id = employee_id;
    dbms_output.put_line(employee Name: '|| employee_name);
  END display_e_name;
END employee_package;
/

begin
employee_package.display_e_name('4');
end
```

Conclusion:

This project was done as a part of Midterm project for Advanced Database Management System. It was undertaken to plan, design and develop a Human Resource management system database.

This database allows to manage employee information by automating core HR, benefits and payroll processes for increased efficiency and productivity. It also reduces the time of computation taken between process by timely generating the necessary reports and statistics. This database will be able to reduce redundant data and error scope by easily creating accurate reporting and analysis. This database will ensure the security of employee information. The aim of creating such a database is to reduce routine administration and promote a paperless environment.

Further Improvement:

The advantages of this software are that this can be enhanced, modified or changed to the growing requirements to the client.

Some of the improvements that can be made to this HRMS are that it can be modified to include employee appraisal features that exits in many complexes HRMS.