Introduction

Database refers to a collection of electronic records that could be processed to produce useful information. The data can be accessed, modified, managed, controlled and organized to perform various data-processing operations. The data is typically indexed across rows, columns and tables that make workload processing and data querying efficient. There are different types of databases: Object-oriented, Relational, Distributed, Hierarchical, Network and others. In enterprise applications, databases involve mission-critical, security-sensitive and compliance-focused record items that have complicated logical relationships with other datasets and grow exponentially over time as the userbase increases. As a result, these organizations require technology solutions to maintain, secure, manage and process the data stored in databases. This is where Database Management System come into play.

Scenario and Problem Definition

Banking systems have been with us for as long as people have been using money. Banks and other financial institutions provide security for individuals, businesses and governments, alike. Let's recap what has been learned with this tutorial:

In general, what banks do is pretty easy to figure out. For the average person banks accept deposits, make loans, provide a safe place for money and valuables, and act as payment agents between merchants and banks.

Banks are quite important to the economy and are involved in such economic activities as issuing money, settling payments, credit intermediation, maturity transformation and money creation in the form of fractional reserve banking.

To make money, banks use deposits and whole sale deposits, share equity and fees and interest from debt, loans and consumer lending, such as credit cards and bank fees.

In addition to fees and loans, banks are also involved in various other types of lending and operations including, buy/hold securities, non-interest income, insurance and leasing and payment treasury services.

History has proven banks to be vulnerable to many risks, however, including credit, liquidity, market, operating, interesting rate and legal risks. Many global crises have been the result of such vulnerabilities and this has led to the strict regulation of state and national banks.

However, other financial institutions exist that are not restricted by such regulations. Such institutions include: savings and loans, credit unions, investment and merchant banks, shadow banks, Islamic banks and industrial banks.

Schema

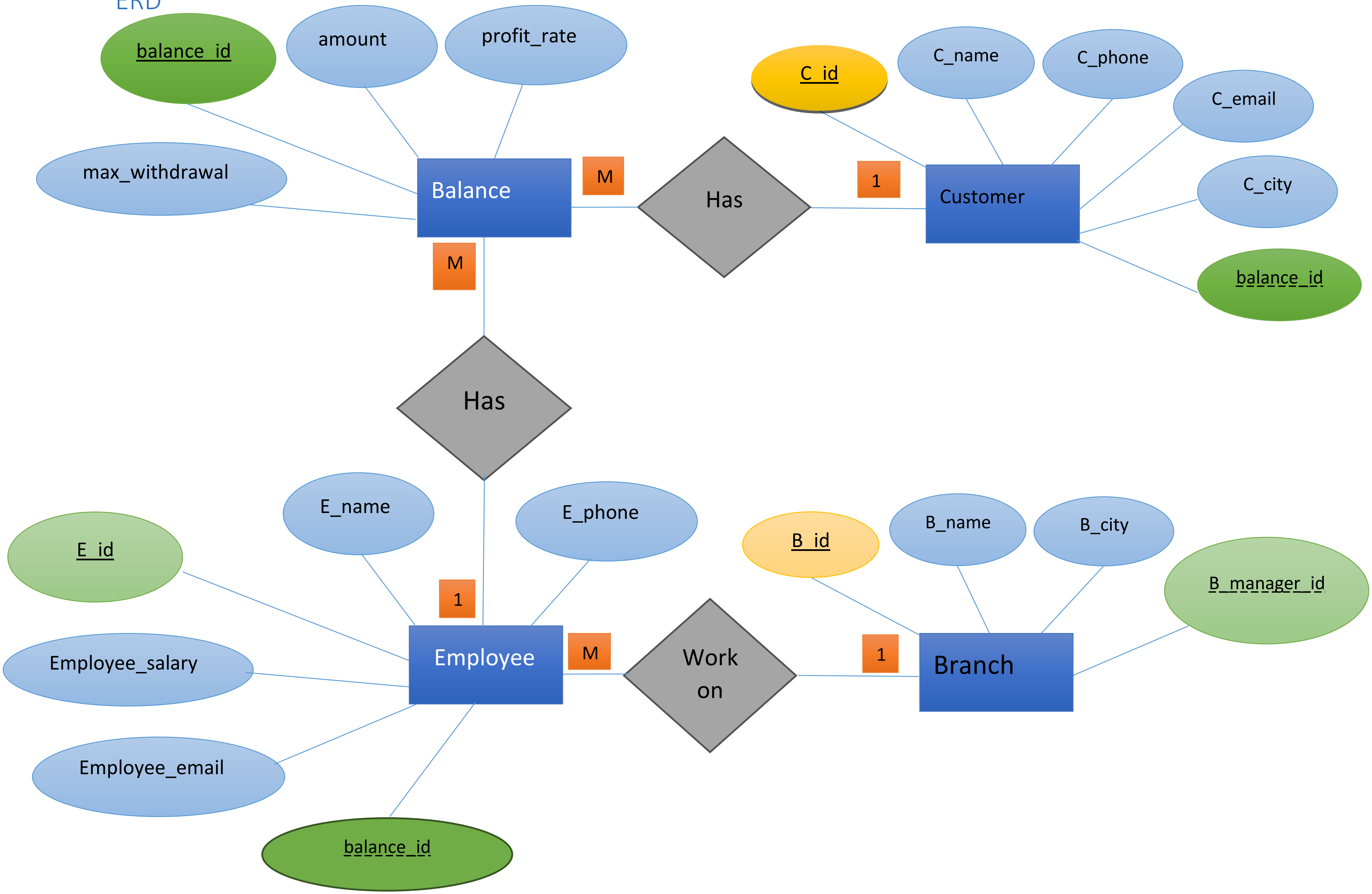
Balance (balance_id, amount, profit_rate, max_withdrawal)

Customer (customer_id, customer_name, customer_phone, customer_email, customer_city, balance_id)

Employee (employee_id, employee_name, employee_phone, employee_email, employee_salary, balance_id)

Branch (branch_id, branch_name, branch_city, branch_manager_id)

ERD


balance_id
amount
profit_rate
max_withdrawal
Balance
M
Has
1
Customer
C_id
C_name
C_phone
C_email
C_city
balance_id
M
Has
1
E_name
E_phone
E_id
Employee_salary
Employee
Employee_email
balance_id
M
Work on
1
Branch
B_id
B_name
B_city
B_manager_id

Create tables

```
Create table Balance(balance_idvarchar (5),
                     Amountnumeric (20, 10),
                     profit_ratenumeric (3, 3),
                     max_withdrawalnumeric (5, 5),
                     primary key (balance_id)
                     );



Create table Customer(customer_idvarchar (5),
                     customer_namevarchar (20),
                     customer_phone varchar (15),
                     customer_email varchar (25),
                     customer_city varchar (15),
                     balance_idvarchar (5)
                     primary key (customer_id),
                          foreign key (balance_id) references Balance;
                     );
```

```
Create table Employee(employee_id varchar (5),
                  employee_name varchar (20),
                      employee_phone varchar (15),
                      employee _email varchar (20),
                      employee_salary varchar (6),
                  balance_id varchar (5)
                  primary key (employee_id),
                      foreign key (balance_id) references Balance;
                  );



Create table Branch(branch_id varchar (5),
                  branch_namevarchar (20),
                  branch_city varchar (4),
                  branch_manager_id varchar (6),
                  primary key (branch_id),
                    foreign key (branch_manager_id) references
                  Employee(employee_id)
                  );
```

Insert values

Balance

```
insert into BalanceVALUES ('1', 30000, 3.25,1000);
insert into BalanceVALUES ('2', 250000, 5.5,2000);
insert into BalanceVALUES ('3', 10000, 2.5,500);
insert into BalanceVALUES ('4', 1000, 5,300);
insert into BalanceVALUES ('5', 900000, 6,5000);
insert into BalanceVALUES ('6', 30000, 3.25,1000);
insert into BalanceVALUES ('7', 250000, 5.5,2000);
insert into BalanceVALUES ('8', 10000, 2.5,500);
insert into BalanceVALUES ('9', 1000, 5,300);
insert into BalanceVALUES ('10', 900000, 6,5000);
```

Customer

```
insert into CustomerVALUES ('11', 'c1', '079', 'c1@bank.com', 'city1', '1');
insert into CustomerVALUES ('22', 'c2', '078', 'c2@bank.com', 'city2', '2');
insert into CustomerVALUES ('33', 'c3', '077', 'c3@bank.com', 'city3', '3');
insert into CustomerVALUES ('44', 'c4', '078', 'c4@bank.com', 'city4', '4');
insert into CustomerVALUES ('55', 'c5', '077', 'c5@bank.com', 'city5', '5');
```

Employee

```
insert into EmployeeVALUES ('111', 'e1', '079', 'e1@bank.com', '2000', '6');

insert into EmployeeVALUES ('112', 'e1', '079', 'e1@bank.com', '2000', '4');

insert into EmployeeVALUES ('222', 'e2', '078', 'e2@bank.com', '2500', '7');

insert into EmployeeVALUES ('333', 'e3', '077', 'e3@bank.com', '5000', '8');

insert into EmployeeVALUES ('444', 'e4', '078', 'e4@bank.com', '8000', '9');

insert into EmployeeVALUES ('555', 'e5', '077', 'e5@bank.com', '9500', '10');
```

Branch

```
insert into Branch VALUES ('100', 'b1', 'Amman','111');

insert into Branch VALUES ('200', 'b2', 'Amman','222');

insert into Branch VALUES ('300', 'b3', 'Zarqa', '333');

insert into Branch VALUES ('400', 'b4', 'Karak', '444');

insert into Branch VALUES ('500', 'b5', 'Aqaba','555');
```

Tables

Balance

Balance (balance_id, amount, profit_rate, max_withdrawal)

Balance

balance_id	amount	profit_rate	max_withdrawal
1	3000	3.25	1000
2	250000	5.5	2000
3	10000	2.5	500
4	1000	5	300
5	900000	6	5000
6	30000	3.25	1000
7	250000	5.5	2000
8	10000	2.5	500
9	1000	5	300
10	900000	6	5000

Customer

Customer (customer_id, customer_name, customer_phone, customer_email, customer_city, balance_id)

Customer

customer_id	customer_name	phone	email	city	balance_id
11	C1	079	c1@bank.com	City1	1
22	C2	078	c2@bank.com	City2	2
33	C3	077	c3@bank.com	City3	3
44	C4	078	c4@bank.com	City4	4
55	C5	077	C5@bank.com	City5	5

Employee

Employee (employee_id, employee_name, employee_phone, employee_email, employee_salary, balance_id)

Employee

employee_id	name	phone	email	salary	**balance_id**
111	E1	079	e1@bank.com	2000	6
112	E2	079	e1@bank.com	2000	4
222	E2	077	e1@bank.com	2500	7
333	E3	078	e1@bank.com	5000	8
444	E4	077	e1@bank.com	8000	9
555	E5	078	e1@bank.com	9500	10

Branch

Branch (branch_id, branch_name, branch_city, branch_manager_id)

Branch

branch_id	Name	City	**branch_manager_id**
100	B1	Amman	111
200	B2	Amman	222
300	B3	Zarqa	333
400	B4	Karak	444
500	B5	Aqaba	555

Oracle

Basic Codes

Execute

```
execute_query;
```

Clear

```
clear_block;
```

Next

```
next_record;
```

Previous

```
previous_record;
```

Last

```
last_record;
```

First

```
first_record;
```

Save

```
commit;
```

Delete

```
go_block('employee');
delete_record;
```

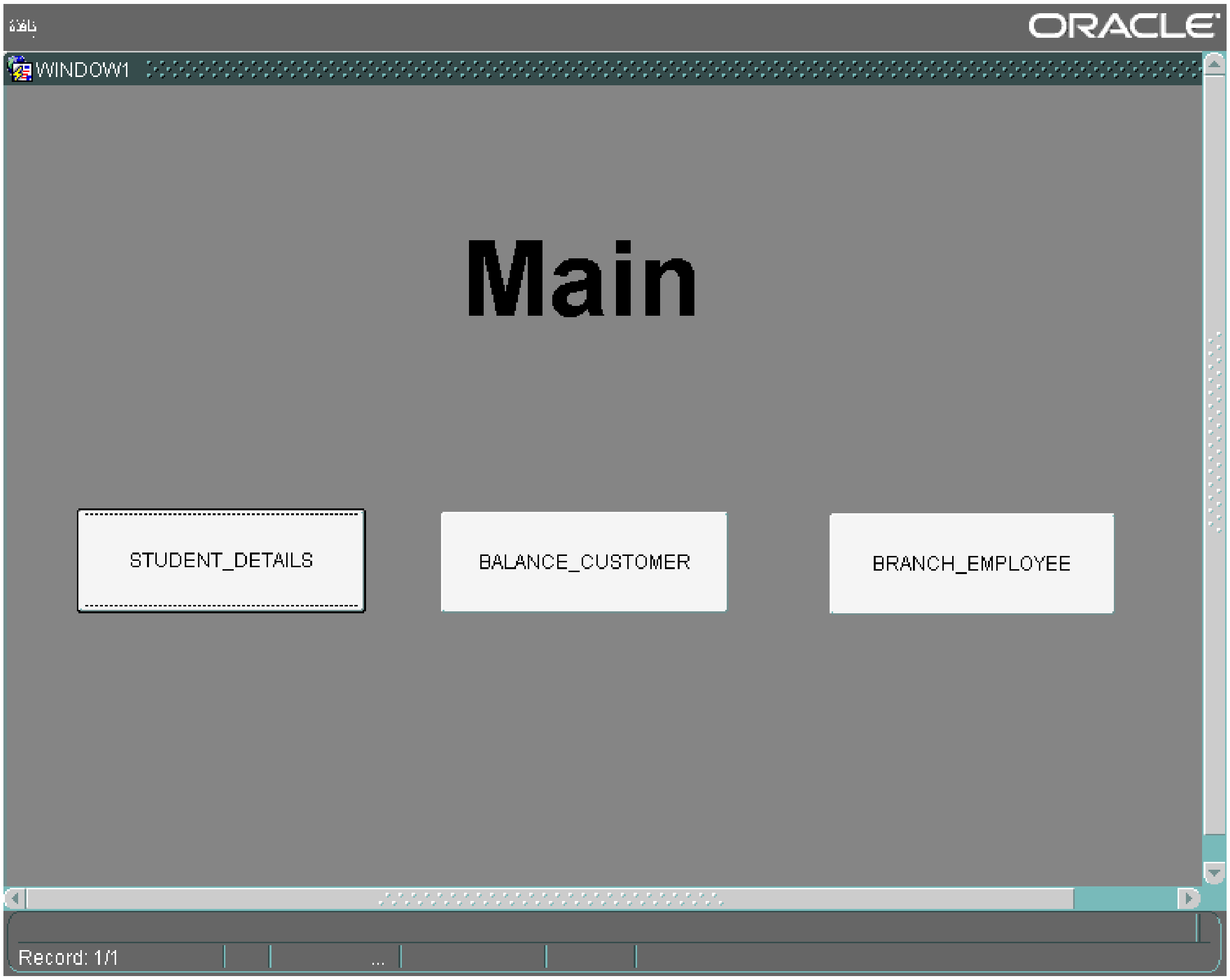
→ Then press (save) button to delete it from the database.

Next Form

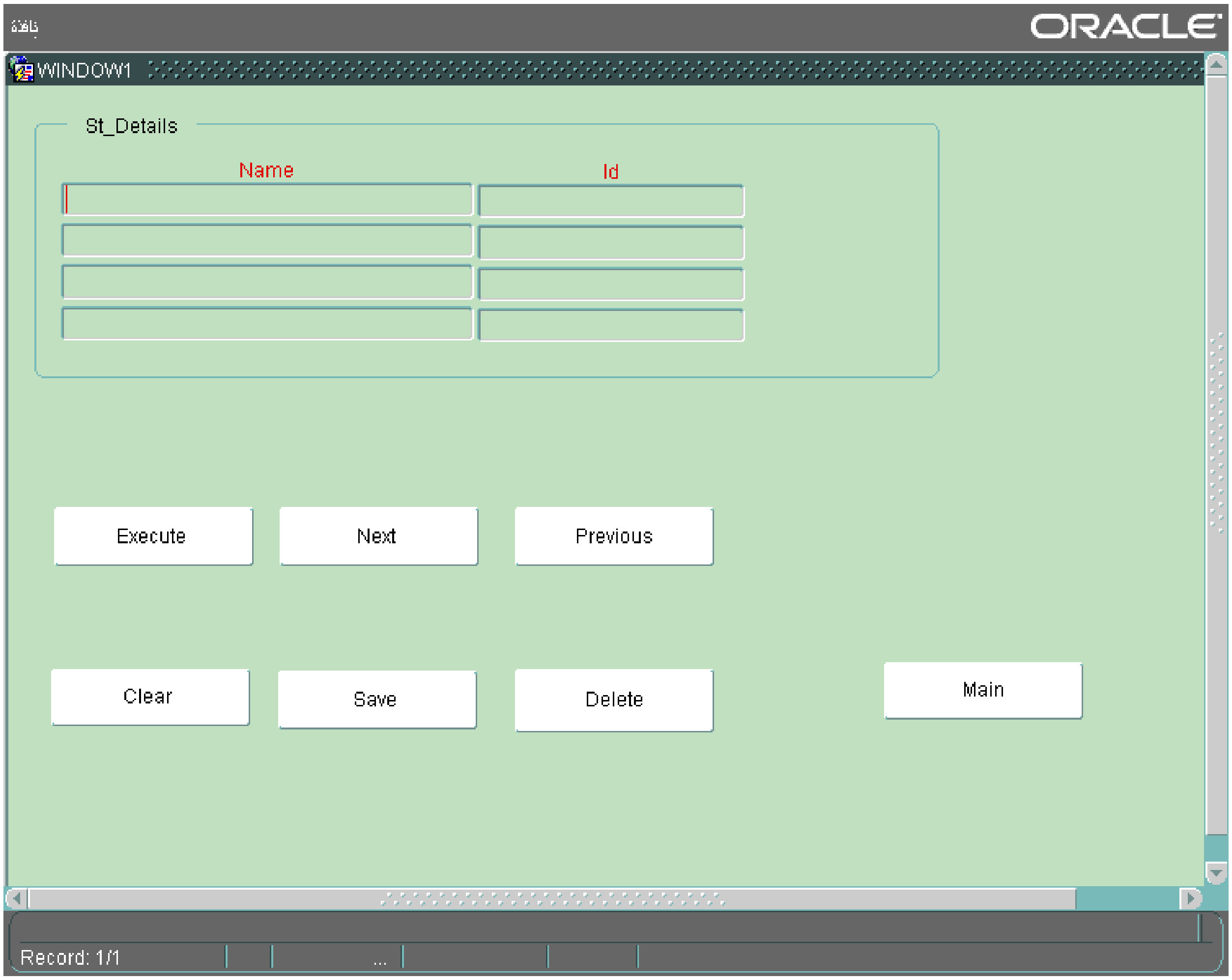
```
declare
A VARCHAR2(1000);
B NUMBER;
BEGIN
A:= GET_APPLICATION_PROPERTY(CURRENT_FORM);
B:= INSTR(A,'\',-1);
A:= SUBSTR(A,1,B);
CALL_FORM(A||'ALO_ADMIN.fmx');
END;
```

Forms

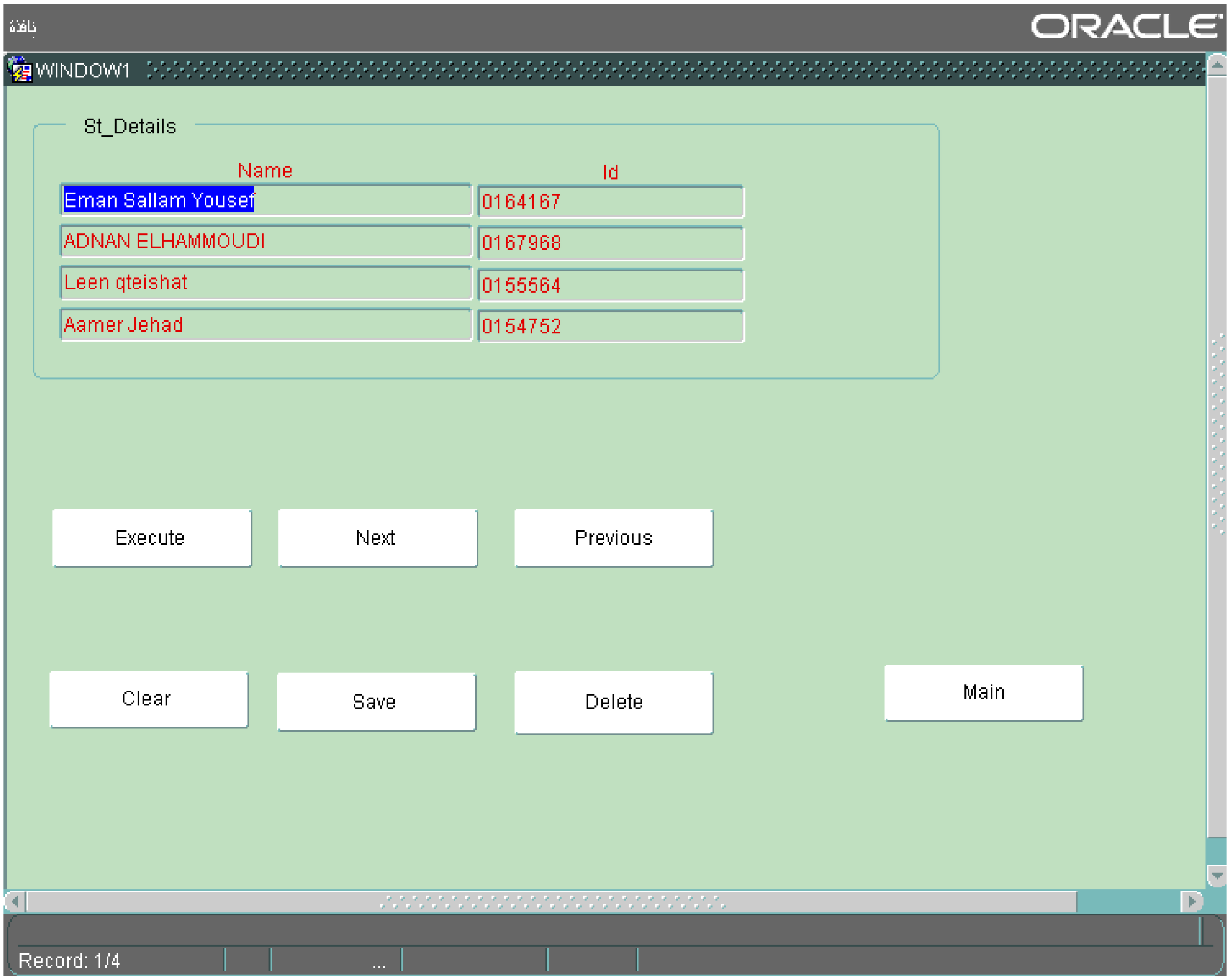
Main Form



Students Form



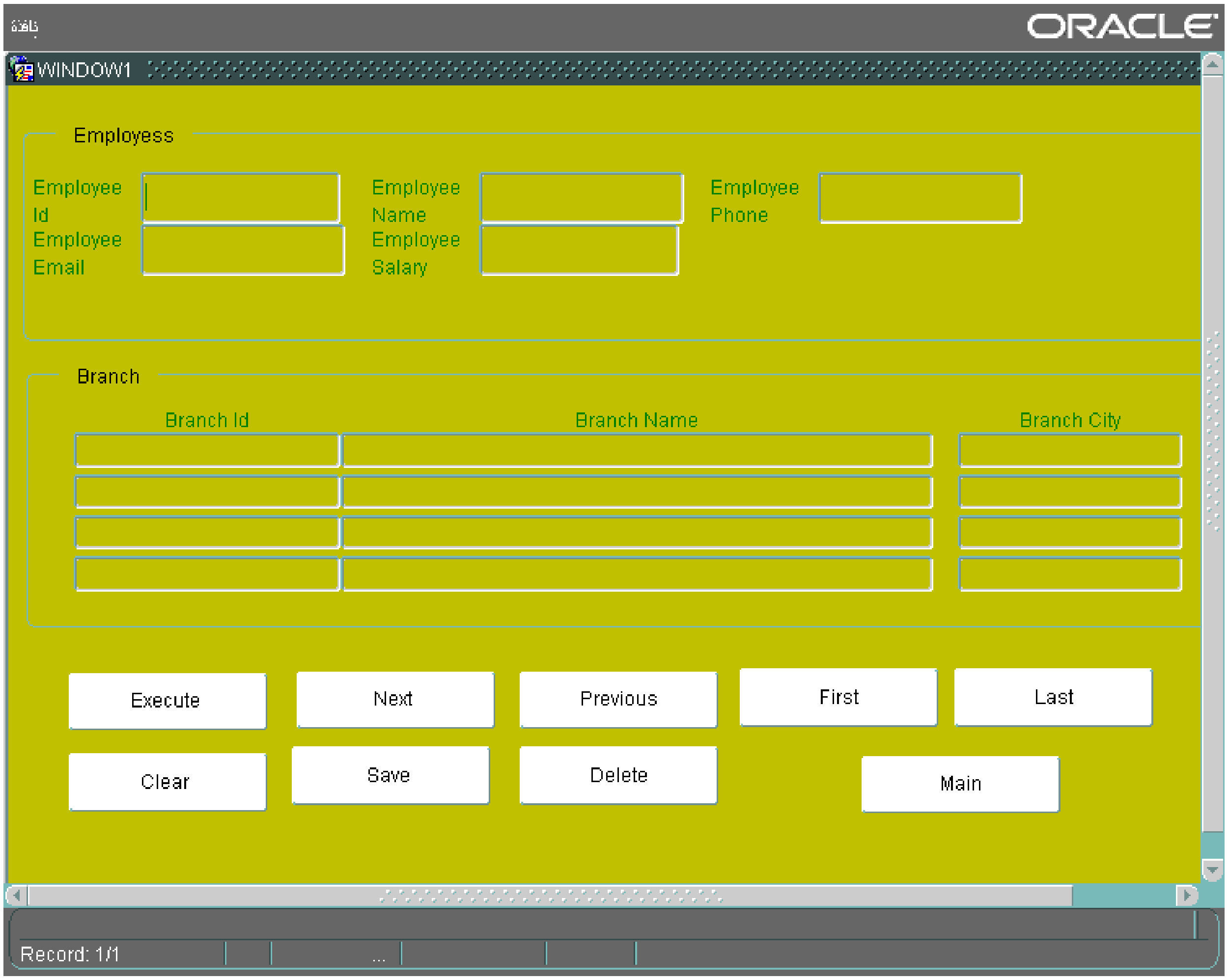
Students Form with Data



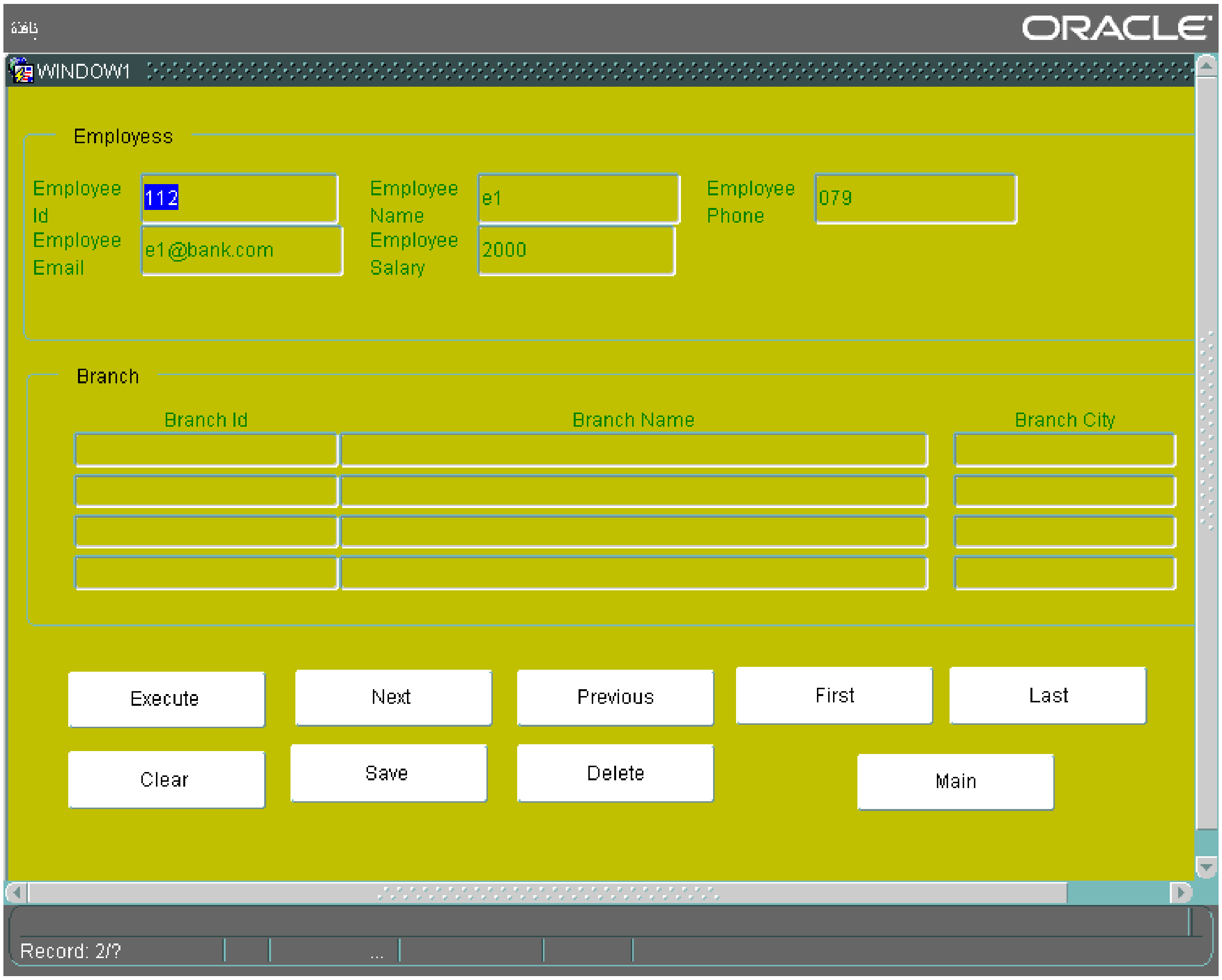
Name	Id
Eman Sallam Yousef	0164167
ADNAN ELHAMMOUDI	0167968
Leen qteishat	0155564
Aamer Jehad	0154752

Record: 1/4

Employee Form



Employee Form with Data



WINDOW1

Employess

Employee Id: 112

Employee Name: e1

Employee Phone: 079

Employee Email: e1@bank.com

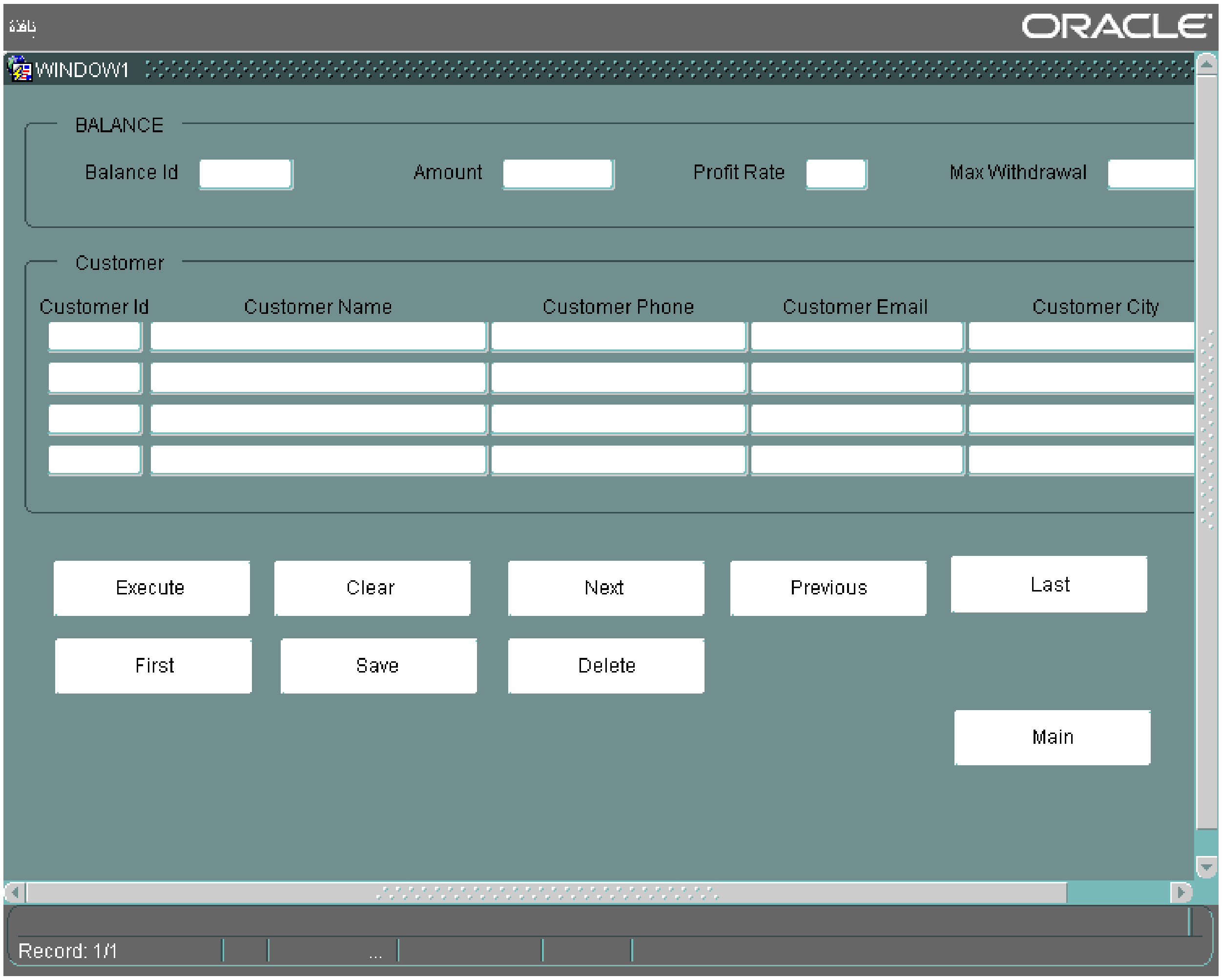
Employee Salary: 2000

Branch

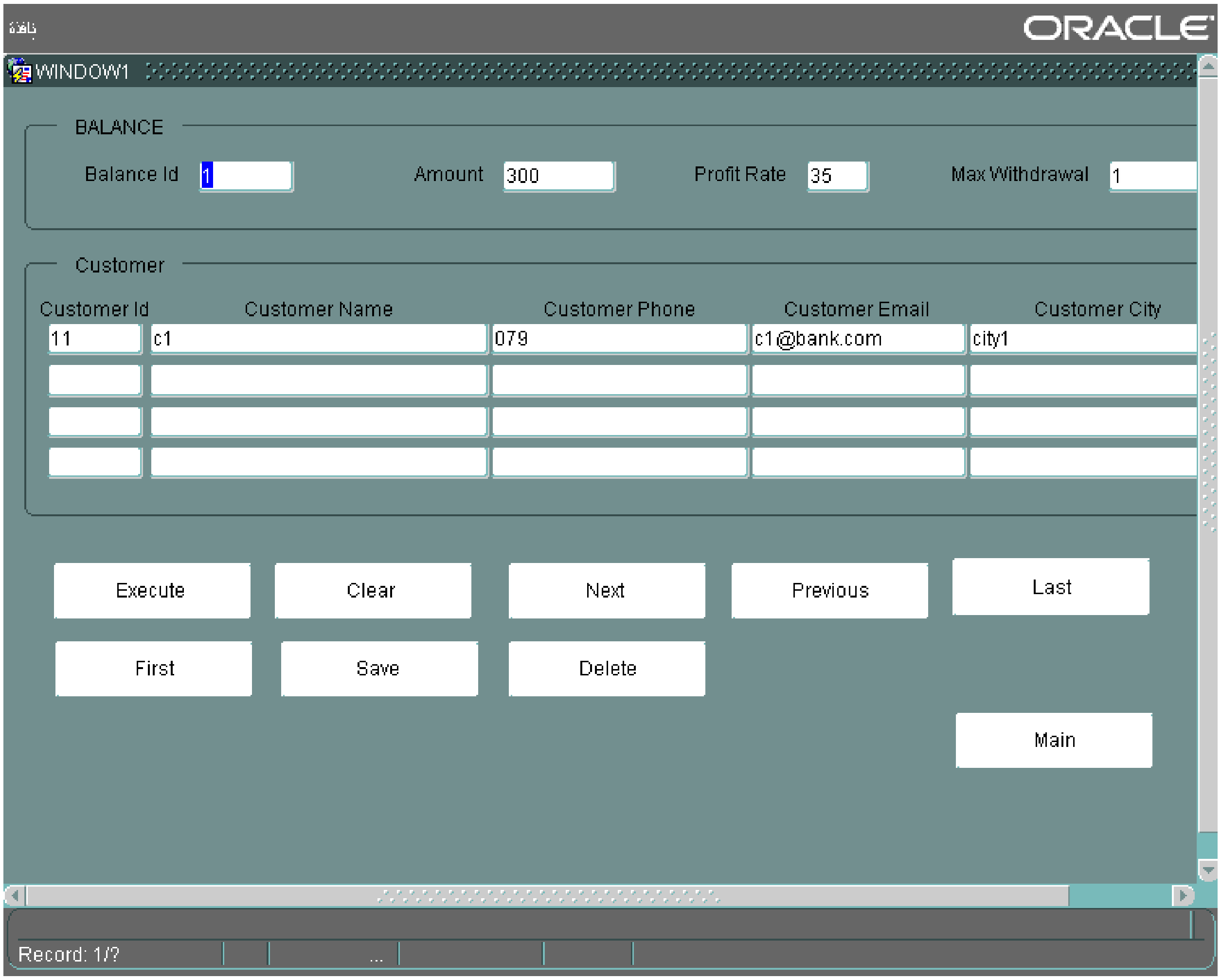
Branch Id	Branch Name	Branch City

Record: 2/?

Balance and Customer Form



Balance and Customer Form with Data



BALANCE

Balance Id 1 Amount 300 Profit Rate 35 Max Withdrawal 1

Customer

Customer Id	Customer Name	Customer Phone	Customer Email	Customer City
11	c1	079	c1@bank.com	city1

Record: 1/?

Conclusion

The project contained a simplified bank system

- The bank is organized into branches. Each branch is located in a particular city and is identified by a unique id. Each branch has a manager Manages the branch.
- Bank customers are identified by their unique ID. The bank stores customer's ID, name and the city where the customer lives.
- Bankers are also identified by their unique ID. The bank stores the name and email address of each employee

References

- Data base Book
- www.investopedia.com
- www.bmc.com/blogs