



VIT[®]

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

D.B.M.S. J-

COMPONENT

ITE1003

Slot: G2 + TG2

TOPIC:

Payroll Management System

Under the Guidance of:

Prof. BIMAL KUMAR RAY

**SCHOOL OF INFORMATION TECHNOLOGY &
ENGINEERING**

SUBMITTED BY:

- Shikhar Shrivastava - (19BIT0091)
- Piyush Kumar - (19BIT0130)
- Aman Pratap Singh - (18BIT0334)

Review 1

ACKNOWLEDGEMENT

The completion of this project would not have been possible without the kind support and help of many individuals. We would like to extend our sincere thanks to all of them. We would like to express our profound gratitude and deep regards to our mentor Professor Bimal Kumar Ray for his guidance and supervision throughout the semester as well as for providing necessary information regarding the project and also for his support in completing the project.

My thanks and appreciation also go to the fellow members of the group in developing the project and people who have willingly helped us out with the best of their abilities.

ABSTRACT

Payroll Management System is basically used to build an application program, that a company uses to manage a record of the employees working in the company. It records the information regarding salaries in detail along with the required details of the employees and the employer.

Only the administration has the legal rights to work with the system. Employees can only log on the system to see their current status. The Payroll Management System deals with the financial aspects of employee's salary, allowances, deductions, gross pay, net pay etc. and generation of pay-slips for a specific period. The outstanding benefit of Payroll Management System is its easy implementation.

A payroll system is software designed to organize all the tasks of employee payment and the filing of employee taxes. These tasks can include keeping track of hours, calculating wages, withholding taxes and deductions, printing and Delivering checks and paying employment taxes to the government.

Payroll software often requires very little input from the employer. The employer is required to input employee wage information and hours then the software calculates the information and performs withholdings automatically. Most payroll software is automatically updated whenever a tax law changes and will remind employers when to file various tax forms.

Data requirements

The following are data requirements for a Payroll management system:

- Company have a Company Id denoted by CID, a name CName. Company consists of different branches. There is one to many relations between company and branch (E.g.)

CName: - ABC Company, CID: - 1

- Company has a branch (n number of branches), Each branch has a CID, Branch Id (BID) Branch name (BNAME). There are several departments of a branch. There is one to many relations between company and branch There is one to one relation between branch and administrator.

(E.g.) CID: -1, BID: -1, BNAME: - IT CID: -1, BID: -2, BNAME: - HR CID: -1, BID: -3, BNAME: - IT CID: -1, BID: -4, BNAME: - HR

- Each branch is headed by anAdministrator. Each administrator has a Designation, Name, Department, Branch Id (BID) and Administrator Id (AID), Administrator year of joining.
- There is one to one relation between branch and administrator. There is one to many relation administrator and salary. There is one to many relation administrator and employee.

*(E.g.) Designation: -GM, Name: -MNQ PQU, Department: -IT, BID: -1, AID: -1 Admin Year Of joining: - 2000
Designation: -GM, Name: - PPQU QBG, Department: -IT, BID: -2, AID: -2 Admin Year Of joining: - 2003*

Designation: -GM, Name: -ABC GHI, Department: -HR, BID: -3,

AID: -3 Admin Year Of joining: - 2005

Designation: -GM, Name: -RTY UIO, Department: -HR, BID: -4,

AID: -4 Admin Year Of joining: - 2004

- Administrator employs Employee (n number of Employees).
Employee have a department and work under Administrator.
Employee have a Phone no, Date of Birth (DOB), Name (EName),
Employee Id (EID), Designation, Basic and Administrator Id (AID)
and

Year of joining.

There is one to many relations between administrator and Employee

There is one to many relations between salary and Employee
There is one to one relation between attendance and employee

*(E.g.)Phone No: -9999654413, DOB: -6/6/1980, EName: - BHU JIJO,
EID: - 1, Designation: - Manager, Basic: - 30000, AID: -1, Year of
joining: -2010*

*Phone No: -9845223453, DOB: -16/2/1984, EName: - BHU JIJO,
EID: - 2, Designation: - Manager, Basic: - 32000, AID: -1, Year of
joining: -2010*

*Phone No: -9998715443, DOB: -26/6/1980, EName: - BHU JIJO,
EID: - 3, Designation: - Manager, Basic: - 30800, AID: -1, Year of
joining: -2010*

- Each Employee gets Salary which is paid by Administrator on basis
of attendance. Attendance is mandatory for each employee.
Attendance comprises of status (whether present or absent), Date,
Intime, Out-time, leave type in-case of leave and leave id in-case of
leave. There is one to one relation between attendance and
Employee

(E.g.)

Status: - Present, Date: -11/7/18, In-time: -10:00, Out- time:19:00,

Leave type: NULL, Leave id: -NULL

Status: - Present, Date: -15/7/18, In-time: -10:05, Out- time:19:01,

Leave type: NULL, Leave id: -NULL

*Status: Absent, Date: -11/7/18, In-time: -NULL, Out-time: NULL,
leave type: Fever, Leave id: - 1090*

- Each Employee certain number of leaves. Leaves are of different types Medical, Causal, Emergency.

There is one to one relation between attendance and leave.

(E.g.)

*LID: - 101, Medical: - 10 Casual: - 45, Emergency: - 7 LID: - 103,
Medical: - 12 Casual: - 43, Emergency: - 7*

Administrator pays salary to n number of Employee.

- Salary has Basic (Basic payment), Medical Allowance, HRA, TA, DA, AID, Bonus, Over_time_amount, Increments. Salary is paid to employee on basis of their attendance. There is one to many relations between salary and Employee. There are one to many relation administrator and salary.

(E.g.)

*Basic: 30000, Medical Allowance: 5000, HRA: 2000, TA: 3000, DA:
-5000, AID: -1, BONUS: -1000, Over_time_amount: - 1000,
Increments: - 500 Basic: 32000, Medical Allowance: 3000, HRA:
1000, TA: 6000, DA: -8000, AID: -2, BONUS: -1500,
Over_time_amount: - 1200, Increments: - 300*

- Administrator pays taxes to the government.
Deduction has tax (Tax deduction), late hour deduction, advance deduction, mess deduction, other deduction. Taxes are paid on the basis of the rules set by the government. There is one to many relations between Administration and Deduction.

FUNCTIONAL REQUIREMENTS

1. Company:

Company has branches and a CID for reference.
There are n number of branches in a company CID
is primary key

2. Branch:

Kind of work done by employee depends on the
branch. Branch has administrators and a BID for
reference.

BID is primary key CID is foreign key

3. Administrator:

Each branch is headed by Administrator.
Administrator heads a branch, employs an employee
and provides salary to employee.

Administrator is given AID for reference AID is
primary key

4. Employee:

Employee works in a department.
Employee work under administrator and are
provided salary. Each employee is given EID for
reference.

EID is primary key AID is foreign key.

5. Deduction:

Employee have tax deductions in Salary. Late hour deduction is also there. EID is primary key and EID is foreign key.

6. Salary:

Basic salary is provided to employee on the basis of attendance. Salary is provided to employee and is paid by administrator.

HRA, TA, DA, Bonus and medical allowance are provided on certain conditions.

Basic is primary key and AID is foreign key.

7. Attendance:

Salary is provided to employee on the basis of attendance. Attendance has Status for reference.

EID is primary key and EID is foreign key.

8. Leave:

Employee can take causal leave or emergency leave or medical leave. LID is the primary key.

9. Search:

This module helps to search the employee details EID wise, salary wise and designation wise.

10. Report:

Administrative reports contain reports like the Salary Report and the Employee Report.

SCENARIO

REMOVAL:

- In case of new tax rules all the calculation of Salary will be changed, in that case we need to remove the Salary and create NEW SALARY according to new rules.
- If timing (in-time and out-time) of company are changed, in that case old Attendance needs to be removed and the NEW ATTENDANCE according to new timings.
- If the overtime of employees is changed, in that case old salary needs to be removed and the new salary set according to overtime.
- If the TA and DA of employees is changed, in that case old salary needs to be removed and the new salary set according to TA and DA.

MODIFICATION:

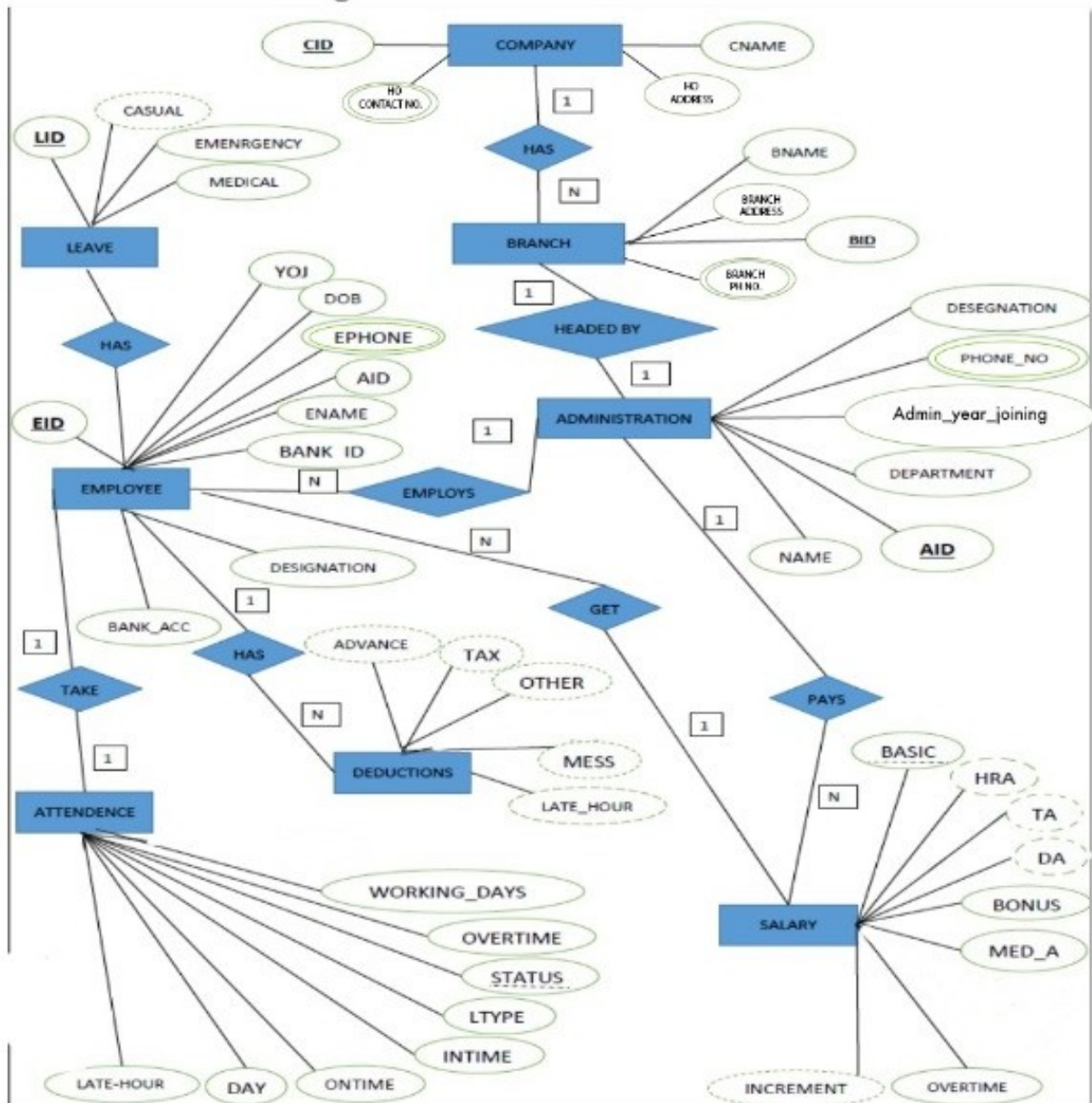
- In case of change in designation of an employee, we need to modify the designation. E.g. In case of promotion from Manager to Sr. Manager we need to modify.
- In case of change in phone number of an employee, we need to modify the Phone no.
E.g. In case of change in number from 9898989874 to 9009009009 we need to modify the Phone no value.
- In case of change in AID of an administrator , we need to modify the AID.
E.g. In case of change in AID of an administration from 3 to 5 we need to modify the AID
- In case of change in working days of attendance , we need to modify the working days option.
E.g. In case of change of working days 30 to 20 we need to modify in working days of attendance.

DATA RETRIEVAL:

- In case of calculating pension for a retired person, we require his then salary as well as current salary. We need to retrieve the previous (old) salary of that person.
- In case of providing increment and promotion we require his then salary, year of joining, current designation and leaves. We need to retrieve his old information, work report and leaves of past.
- In case of calculating number of leaves taken by an employee in a year. We need to retrieve his leaves from past months of the year.
- In case of knowing the time of working of an Administrator. We need to retrieve his year of joining from Admin year joining.
- In case giving salary to the employee. We need to retrieve the status of the employee in last past days.
- In case of calculating the extra time by an employee. We need to calculate the late hour.
- In case of increasing the salary we need to retrieve the extra time from overtime.

- In case of hiring an employee, we need to retrieve salary of previous employee at that designation. So, we can hire the new employee according to the salary of previous employee at the same designation.

ER Diagram



Caption

Key Constraints

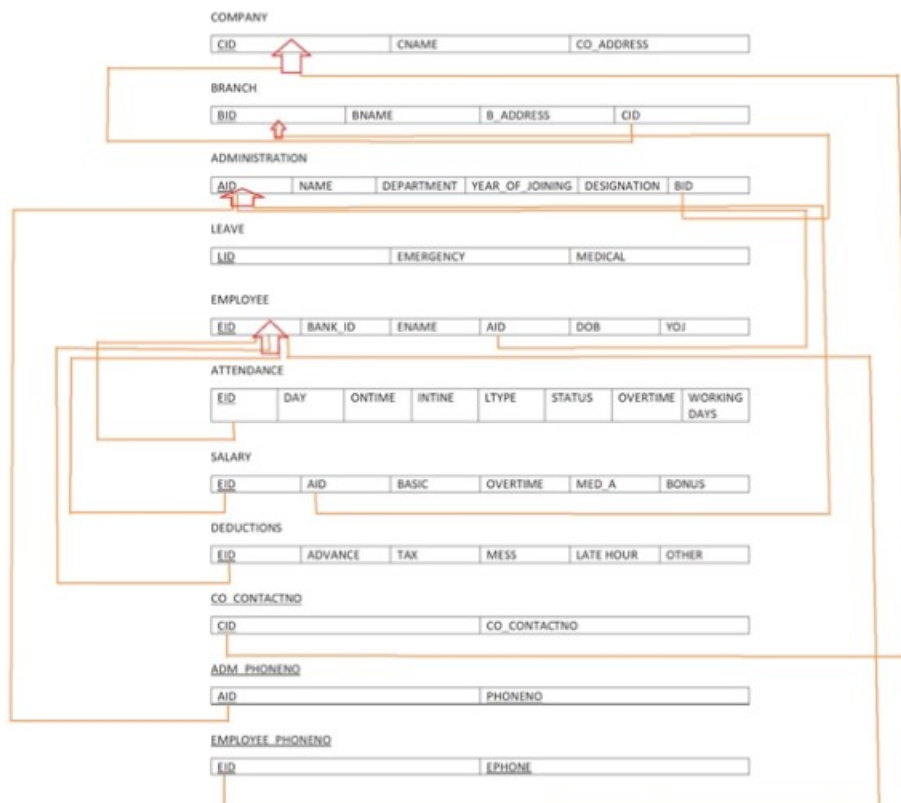
- NOT NULL. Ex: CID
- UNIQUE. Ex: EID
- DEFAULT. Ex: BASIC
- CHECK. Ex: DOB
- PRIMARY KEY. Ex: CID
- FOREIGN KEY. Ex: EID

Cardinality Constraint

- One to one. Ex: Branch to Administration
- One to many. Ex: Administration to Salary
- Many to one. Ex: Employee to Salary
- Many to many. No example in payroll man.

Review 2

Relational database scheme



Caption

Making the tables

Code to create Table:

- **create table** Company(
 CID number(10) primary key NOT NULL, CName varchar(10),
 HQaddress varchar(20),HQphone number(10), check(HQphone
 >10000000000)

);
- **create table** Branch(
 BID number(10) primary key NOT NULL,
 BName varchar(15),
 Baddress varchar(20),
 Bphone number(10),
 CID number(5),
 check(Bphone >10000000000),
 constraints fk_cid foreign key(CID) references Company(CID)
);
- **create table** Administration(
 AID number(10) primary key NOT NULL,
 AName varchar(20),
 Department varchar(15) NOT NULL,
 A_year_of_joining date NOT NULL,
 Designation varchar(15) NOT NULL,
 APhone number(10), check(APhone>10000000000),
 BID number(8),
 constraints fk_bid foreign key(BID) references Branch(BID)
);
- **create table** Employee(
 EID number(10) primary key NOT NULL,
 ENAME varchar(20),
 Designation varchar(15) NOT NULL,

DOB date,
YOJ date NOT NULL,
EPhone number(10) NOT NULL,
bank_account varchar(255) DEFAULT NULL,
bank_id number(6) DEFAULT NULL,
AID number(10),
check(EPhone>10000000000),
constraints fk_aid foreign key(AID) references Administration(AID));

● **create table**

Deduction(late_hour_deduction number(6)
default null,
advance_deduction number(6) DEFAULT NULL,
other_deduction number(6) DEFAULT NULL,
Tax_deduction number(11) DEFAULT NULL,
mess_deduction number(6) DEFAULT NULL
);

● **create table Salary(**

Basic number(15) primary key NOT NULL,
HRA number(10) DEFAULT NULL,
TA number(10) DEFAULT NULL,
DA number(10) DEFAULT NULL,
Bonus number(10) DEFAULT NULL,
Medical number(10) DEFAULT NULL,
increments number(6) DEFAULT NULL,
over_time_amount number(6) DEFAULT NULL
check(Basic>0));

● **create table Attendance(**

Intime varchar(10) DEFAULT NULL,
Ontime varchar(10) DEFAULT NULL,
Ltype varchar2(10) DEFAULT NULL,
Day date DEFAULT NULL,
Status varchar(10),
over_time_hours number(6) DEFAULT NULL,
working_days number(6) DEFAULT NULL,
late_hours number(6) default null
);

● **create table Leave(**

LID number(10) primary key NOT NULL,
Medical number(10) NULL,
Casual number(10) NULL,
Emergency number(10) NULL
);

TABLE COMPANY:

Live SQL

FeedbackHelpshikhar.shrivastava2019@vitstudent.ac.in

SQL Worksheet

ClearFindActionsSaveRun

```
1 create table Company(  
2   CID number(10) primary key NOT NULL,  
3   CName varchar(10),  
4   HQaddress varchar(20),  
5   HQphone number(10),  
6   check(HQphone >1000000000)  
7 );  
8
```

Table created.

Caption

Table created.

TABLE COMPANY

Column	Null?	Type
CID	NOT NULL	NUMBER(10,0)
CNAME	–	VARCHAR2(10)
HQADDRESS	–	VARCHAR2(20)
HQPHONE	–	NUMBER(10,0)

Download CSV4 rows selected.

Caption

TABLE BRANCH:

```

17 create table Branch(
18 BID number(10) primary key NOT NULL,
19 BName varchar(15),
20 Baddress varchar(20),
21 Bphone number(10),
22 CID number(5),
23 check(Bphone >1000000000),
24 constraints fk_cid foreign key(CID) references Company(CID) );
25 describe Branch;
26

```

Caption

TABLE BRANCH

Column	Null?	Type
BID	NOT NULL	NUMBER(10,0)
BNAME	–	VARCHAR2(15)
BADDRESS	–	VARCHAR2(20)
BPHONE	–	NUMBER(10,0)
CID	–	NUMBER(5,0)

[Download CSV](#)

5 rows selected.

Caption

TABLE ADMINISTRATION:

```
33 create table Administration(  
34 AID number(10) primary key NOT NULL,  
35 AName varchar(20),  
36 Department varchar(15) NOT NULL,  
37 A_year_of_joining date NOT NULL,  
38 Designation varchar(15) NOT NULL,  
39 APhone number(10), check(APhone>1000000000),  
40 BID number(8),  
41 constraints fk_bid foreign key(BID) references Branch(BID)  
42 );  
43  
44
```

Table created.

Caption

TABLE ADMINISTRATION		
Column	Null?	Type
AID	NOT NULL	NUMBER(10,0)
ANAME	–	VARCHAR2(20)
DEPARTMENT	NOT NULL	VARCHAR2(15)
A_YEAR_OF_JOINING	NOT NULL	DATE
DESIGNATION	NOT NULL	VARCHAR2(15)
APHONE	–	NUMBER(10,0)
BID	–	NUMBER(8,0)

Download CSV

Caption

TABLE EMPLOYEE:

```
29 create table Employee(  
30 EID number(10) primary key NOT NULL,  
31 EName varchar(20),  
32 Designation varchar(15) NOT NULL,  
33 DOB date,  
34 YOJ date NOT NULL,  
35 EPhone number(10) NOT NULL, bank_account varchar(255) DEFAULT NULL, bank_id number(6) DEFAULT NULL, AID number(10),  
36 check(EPhone>1000000000),  
37 constraints fk_aid foreign key(AID) references Administration(AID) );  
38 |
```

Table created.

Caption

TABLE EMPLOYEE

Column	Null?	Type
EID	NOT NULL	NUMBER(10,0)
ENAME	–	VARCHAR2(20)
DESIGNATION	NOT NULL	VARCHAR2(15)
DOB	–	DATE
YOJ	NOT NULL	DATE
EPHONE	NOT NULL	NUMBER(10,0)
BANK_ACCOUNT	–	VARCHAR2(255)
BANK_ID	–	NUMBER(6,0)
AID	–	NUMBER(10,0)

Download CSV

9 rows selected.

Caption

TABLE DEDUCTION:

```
54
55 create table Deduction(
56 late_hour_deduction number(6) default null,
57 advance_deduction number(6) DEFAULT NULL,
58 other_deduction number(6) DEFAULT NULL,
59 Tax_deduction number(11) DEFAULT NULL,
60 mess_deduction number(6) DEFAULT NULL );
61
62
63
64
65
66
```

Table created.

Caption

TABLE DEDUCTION

Column	Null?	Type
LATE_HOUR_DEDUCTION	–	NUMBER(6,0)
ADVANCE_DEDUCTION	–	NUMBER(6,0)
OTHER_DEDUCTION	–	NUMBER(6,0)
TAX_DEDUCTION	–	NUMBER(11,0)
MESS_DEDUCTION	–	NUMBER(6,0)

Download CSV

5 rows selected.

TABLE SALARY:

```

63
64 create table Salary(
65 Basic number(15) primary key NOT NULL,
66 HRA number(10) DEFAULT NULL,
67 TA number(10) DEFAULT NULL,
68 DA number(10) DEFAULT NULL,
69 Bonus number(10) DEFAULT NULL,
70 Medical number(10) DEFAULT NULL,
71 increments number(6) DEFAULT NULL,
72 over_time_amount number(6) DEFAULT NULL
73 check(Basic>0));
74
75

```

Table created.

Caption

TABLE ATTENDANCE:

```

75
76 create table Attendance(
77 Intime varchar(10) DEFAULT NULL,
78 Ontime varchar(10) DEFAULT NULL,
79 Ltype varchar2(10) DEFAULT NULL,
80 Day date DEFAULT NULL,
81 Status varchar(10),
82 over_time_hours number(6) DEFAULT NULL,
83 working_days number(6) DEFAULT NULL,
84 late_hours number(6) default null );
85

```

Table created.

Caption

TABLE ATTENDANCE

Column	Null?	Type
INTIME	–	VARCHAR2(10)
ONTIME	–	VARCHAR2(10)
LTYPE	–	VARCHAR2(10)
DAY	–	DATE
STATUS	–	VARCHAR2(10)
OVER_TIME_HOURS	–	NUMBER(6,0)
WORKING_DAYS	–	NUMBER(6,0)
LATE_HOURS	–	NUMBER(6,0)

[Download CSV](#)

8 rows selected.

Caption

TABLE LEAVE:

```

86 create table Leave(
87 LID number(10) primary key NOT NULL,
88 Medical number(10) NULL,
89 Casual number(10) NULL,
90 Emergency number(10) NULL );
91
92
93
94

```

Table created.

Caption

TABLE LEAVE

Column	Null?	Type
LID	NOT NULL	NUMBER(10,0)
MEDICAL	–	NUMBER(10,0)
CASUAL	–	NUMBER(10,0)
EMERGENCY	–	NUMBER(10,0)

[Download CSV](#)

4 rows selected.

Caption

CODE TO INSERT VALUES TO TABLE:

TABLE COMPANY VALUES:

**insert into Company values ('142',
'SAMSUNG','NewDelhi','8103292041');**

**insert into Company values ('143',
'Apple','Mumbai','9826752661');**

1 row(s) inserted.

CID	CNAME	HQADDRESS	HQPHONE
142	SAMSUNG	NewDelhi	8103292041
143	Apple	Mumbai	9826752661

Download CSV

2 rows selected.

Caption

TABLE BRANCH VALUES:

insert into Branch

**values('1154','samsung','Mumbai','123456
7899','143');**

insert into Branch

**values('1238','nokia','New
Delhi','2345678999','142');**

insert into Branch

**values('3211','apple','Chennai','34567899
99','144');**

insert into Branch

**values('7216','vivo','Kolkata','4567899999
,','142');**

insert into Branch

**values('4574','oppo','Pune','5678999999','
142');**

BID	BNAME	BADDRESS	BPHONE	CID
1154	samsung	Mumbai	1234567899	143
1238	nokia	New Delhi	2345678999	142
3211	apple	Chennai	3456789999	144
7216	vivo	Kolkata	4567899999	142
4574	oppo	Pune	5678999999	142

Download CSV

5 rows selected.

Caption

TABLE ADMINISTRATION VALUES:

insert into Administration values(17658,'Ramesh Singh','HR','01-JAN-20','DGM',7732934808,1154);

insert into Administration values(23754,'Khushi Kumar','RD','02-JAN-20','GM',9799713910,1238);

insert into Administration values(10694,'John Edward' ,' Finance','02-FEB-20', 'Director', 9876534808, 3211);

insert into Administration values(92345,'Alok Misra','Marketing','02-JAN-29','DGM',7342534808,3211);

insert into Administration values(77864,'Lokesh Pratap','IT','02-MAR-17','DGM',6163098763,4574);

```

34 insert into Administration values(23754,'Khushi Kumar','RD','02-JAN-20','GM',9799713910,1238);
35 insert into Administration values(10694,'John Edward' ,' Finance','02-FEB-20', 'Director', 9876534808, 3211);
36 insert into Administration values(92345,'Alok Misra','Marketing','02-JAN-29','DGM',7342534808,3211);
37 insert into Administration values(77864,'Lokesh Pratap','IT','02-MAR-17','DGM',6163098763,4574);
38 SELECT * FROM Administration;

```

AID	ANAME	DEPARTMENT	A_YEAR_OF_JOINING	DESIGNATION	APHONE	BID
92345	Alok Misra	Marketing	02-JAN-29	DGM	7342534808	3211
23754	Khushi Kumar	RD	02-JAN-20	GM	9799713910	1238
10694	John Edward	Finance	02-FEB-20	Director	9876534808	3211
77864	Lokesh Pratap	IT	02-MAR-17	DGM	6163098763	4574

Caption

TABLE EMPLOYEE VALUES:

insert into Employee values(1262567,'Hitesh Gorantla','Manager','02-FEB-2019','03-FEB-2017',9812047898,'HDFC BANK',55123,17658);
insert into Employee values(3162561,'Rishi Tej','Ass. Manager','02-FEB-2015','03-FEB-2013',6312047893,'SBI BANK',12320,23754);
insert into Employee values(1234567,'Jatin Minon','Supervisor','02-FEB-2013','03-FEB-2014',7890047898,'ICIC BANK',56231,10694);
insert into Employee values(7652569,'KumarAnmol','Worker','04-FEB-2019','07-FEB-2017',8912009489,'HDFC BANK',32343,92345);
insert into Employee values(1269976,'JK Singh','Sr. Manager','02-FEB-2011','03-FEB-2012',9834547788,'SBI BANK',87980,77864);

51 lues(1262567,'Hitesh Gorantla','Manager','02-FEB-2019','03-FEB-2017',9812047898,'HDFC BANK',55123,17658);
52 lues(3162561,'Rishi Tej','Ass. Manager','02-FEB-2015','03-FEB-2013',6312047893,'SBI BANK',12320,23754);
53 lues(1234567,'Jatin Minon','Supervisor','02-FEB-2013','03-FEB-2014',7890047898,'ICIC BANK',56231,10694);
54 lues(7652569,'Kumar Anmol','Worker','04-FEB-2019','07-FEB-2017',8912009489,'HDFC BANK',32343,92345);
55 lues(1269976,'JK Singh','Sr. Manager','02-FEB-2011','03-FEB-2012',9834547788,'SBI BANK',87980,77864);

EID	ENAME	DESIGNATION	DOB	YOJ	EPHONE	BANK_ACCOUNT	BANK_ID	AID
3162561	Rishi Tej	Ass. Manager	02-FEB-15	03-FEB-13	6312047893	SBI BANK	12320	23754

Caption

TABLE SALARY VALUES:

insert into Salary values(25000,2500,2000,1000,4500,2500,0,1000);
insert into Salary values(35000,2100,1000,1000,3500,1700,0,2100);
insert into Salary values(15000,2700,2500,1000,2500,1200,0,1700);
insert into Salary values(27000,3500,2050,1000,3500,3200,0,2500);
insert into Salary values(29000,500,1700,1000,1500,3100,0,1100);

114 insert into Salary values(35000,2100,1000,1000,3500,1700,0,2100);
115 insert into Salary values(15000,2700,2500,1000,2500,1200,0,1700);
116 insert into Salary values(27000,3500,2050,1000,3500,3200,0,2500);
117 insert into Salary values(29000,500,1700,1000,1500,3100,0,1100);
118 select * from Salary;
119

BASIC	HRA	TA	DA	BONUS	MEDICAL	INCREMENTS	OVER_TIME_AMOUNT
25000	2500	2000	1000	4500	2500	0	1000
35000	2100	1000	1000	3500	1700	0	2100
15000	2700	2500	1000	2500	1200	0	1700
27000	3500	2050	1000	3500	3200	0	2500
29000	500	1700	1000	1500	3100	0	1100

[Download CSV](#)

Caption

TABLE ATTENDANCE VALUES:

insert into Attendance values('1262567','102','Vacation', '16-OCT-19', 'Absent',100,3,17658);

insert into Attendance values('3162561','103','Vacation', '18-OCT-19', 'Present',100,3,17658);

insert into Attendance values('1234567','101','Vacation','11-OCT-15', 'Absent',100,3,17658);

insert into Attendance values('7652569','104','Vacation', '15-OCT-17', 'Present',100,3,17658);

insert into Attendance values('1269976','106','Vacation', '19-OCT-18', 'Absent',100,3,17658);

95 insert into Attendance values('1262567','102','Vacation', '16-OCT-19', 'Absent',100,3,17658);

96 insert into Attendance values('3162561','103','Vacation', '18-OCT-19', 'Present',100,3,17658);

97 insert into Attendance values('1234567','101','Vacation', '11-OCT-15', 'Absent',100,3,17658);

98 insert into Attendance values('7652569','104','Vacation', '15-OCT-17', 'Present',100,3,17658);

99 insert into Attendance values('1269976','106','Vacation', '19-OCT-18', 'Absent',100,3,17658);

100 select * from Attendance;

101 create table Salary

INTIME	ONTIME	LTYPE	DAY	STATUS	OVER_TIME_HOURS	WORKING_DAYS	LATE_HOURS
1262567	102	Vacation	16-OCT-19	Absent	100	3	17658
3162561	103	Vacation	18-OCT-19	Present	100	3	17658
1234567	101	Vacation	11-OCT-15	Absent	100	3	17658
7652569	104	Vacation	15-OCT-17	Present	100	3	17658
1269976	106	Vacation	19-OCT-18	Absent	100	3	17658
1262567	101	Vacation	17-OCT-19	Absent	100	3	17658

Download CSV

Caption

TABLE DEDUCTION VALUES:

insert into Deduction values(1262567,2500,2500,2000,1000,4500);

insert into Deduction values(3162561,3500,2100,1000,1000,3500);

insert into Deduction values(1234567,1500,2700,2500,1000,2500);

insert into Deduction values(7652569,2700,3500,2050,1000,3500);

insert into Deduction values(1269976,2900,500,1700,1000,1500);

79

insert into Deduction values(12627,2500,2500,2000,1000);

80

insert into Deduction values(3161,3500,2100,1000,1000);

81

insert into Deduction values(12347,1500,2700,2500,1000);

82

insert into Deduction values(7659,2700,3500,2050,1000);

83

insert into Deduction values(12696,2900,500,1700,1000);

84

select * from Deduction;

1 row(s) inserted.

LATE_HOUR_DEDUCTION	ADVANCE_DEDUCTION	OTHER_DEDUCTION	TAX_DEDUCTION	MESS_DEDUCTION
12627	2500	2500	2000	1000
3161	3500	2100	1000	1000
12347	1500	2700	2500	1000
7659	2700	3500	2050	1000
12696	2900	500	1700	1000

[Download CSV](#)

Caption

TABLE LEAVE VALUES:

insert into Leave values(101,3,0,3);
insert into Leave values(102,2,2,0);
insert into Leave values(103,1,5,3);
insert into Leave values(104,3,9,2);
insert into Leave values(105,0,8,6);

65	insert into Leave values(102,2,2,0);
66	insert into Leave values(103,1,5,3);
67	insert into Leave values(104,3,9,2);
68	insert into Leave values(105,0,8,6);
69	select * from Leave;

LID	MEDICAL	CASUAL	EMERGENCY
101	3	0	3
102	2	2	0
103	1	5	3
104	3	9	2

Caption

Review 3

- There should be at least four different scenarios of removal of old data.

DELETE

1. DELETE FROM Leave WHERE LID='102';

138

139

DELETE FROM Leave WHERE LID='102';

140

|

141

142

1 row(s) deleted.

Caption

2. *DELETE FROM Company WHERE CID='142';*

```
138
139 DELETE FROM Leave WHERE LID='102';
140 DELETE FROM Employee WHERE EID='1262567';
141
142 DELETE FROM Company WHERE CID='142';
1 row(s) inserted.

1 row(s) deleted.
```

Caption

3. *DELETE FROM Leave WHERE LID='102';*

```
139 DELETE FROM Leave WHERE LID='102';
140 DELETE FROM Employee WHERE EID='1262567';
141
142 DELETE FROM Leave WHERE LID='105';
1 row(s) inserted.

1 row(s) deleted.
```

Caption

4. *DELETE FROM Employee;*

```

138
139 DELETE FROM Leave WHERE LID='102';
140 DELETE FROM Employee WHERE EID='1262567';
141
142 DELETE FROM Leave WHERE LID='105';
143
144 DELETE FROM Employee;
145
146
1 row(s) deleted.

0 row(s) deleted.

1 row(s) deleted.

1 row(s) deleted.

```

Caption

● Four different scenarios for modification of existing data

UPDATE

1.
Update branch
set baddress = 'Banglore'
where bid = 1238;

```

155 Update branch
156 set baddress = 'Banglore'
157 where bid = 1238;
158
159
1 row(s) updated.

```

Caption

154	
155	Update branch
156	set baddress = 'Banglore'
157	where bid = 1238;
158	select * from Branch;
159	
160	

BID	BNAME	BADDRESS	BPHONE	CID
1238	nokia	Banglore	2345678999	142
7216	vivo	Kolkata	4567899999	142
4574	oppo	Pune	5678999999	142

Download CSV

Caption

2.

Update branch set baddress = 'Banglore' where bid = 1238;

Update branch set baddress = 'Banglore' where bid = 1238;

(s) updated.

Caption

3.

update salary set bonus = 5000 where basic > 25000;

190

191 update salary set bonus = 5000 where basic > 25000;

192 select * from Salary;

193

3 row(s) updated.

BASIC	HRA	TA	DA	BONUS	MEDICAL	INCREMENTS	OVER_TIME_AMOUNT
25000	2500	2000	1000	4500	2500	0	1000
35000	2100	1000	1000	5000	1700	0	2100
15000	2700	2500	1000	2500	1200	0	1700
27000	3500	2050	1000	5000	3200	0	2500
29000	500	1700	1000	5000	3100	0	1100

Caption

4.

Update branch set baddress = 'gwalior' where bid = 1238;

*select * from branch;*

194

195 Update branch set baddress = 'gwalior' where bid = 1238;

196 select * from branch;

197

198

1 row(s) updated.

BID	BNAME	BADDRESS	BPHONE	CID
1238	nokia	gwalior	2345678999	142
7216	vivo	Kolkata	4567899999	142
4574	oppo	Pune	5678999999	142

Download CSV

3 rows selected.

Caption

- Eight different scenarios for data retrieval

SELECT

1. select A.AID, B.BID, B.BNAME FROM
ADMINISTRATION A, BRANCH B
WHERE A.BID = B.BID;

```
162 select A.AID, B.BID, B.BNAME FROM ADMINISTRATION A, BRANCH B
163 WHERE A.BID = B.BID;
164
```

AID	BID	BNAME
23754	1238	nokia

[Download CSV](#)

Caption

2. select a.aid, b.bid, c.cname from
administration a,
branch b, company c where b.cid = c.cid
and cname = 'SAMSUNG' and a.bid = b.bid;

```
164  
165 select a.aid, b.bid, c.cname from administration a,  
166 branch b, company c where b.cid = c.cid  
167 and cname = 'SAMSUNG' and a.bid = b.bid;  
168
```

AID	BID	CNAME
23754	1238	SAMSUNG

[Download CSV](#)

Caption

3. select e.eid, e.ename, e.eid, s.hra, s.ta
from employee e,
salary s where e.eid = s.basic and
designation = 'Manager';

```
70 select e.eid, e.ename, e.eid, s.hra, s.ta from employee e,  
71 salary s where e.eid = s.basic and designation = 'Manager';  
72
```

data found

Caption

4.

SELECT medical, casual
FROM Leave;

```
174 SELECT medical, casual
175 FROM Leave;
176
```

no data found

MEDICAL	CASUAL
3	0
1	5
3	9

[Download CSV](#)
3 rows selected.

Caption

5.

Select * from Company;

```
178 Select * from Company;
179
```

CID	CNAME	HQADDRESS	HQPHONE
143	Apple	Mumbai	9826752661
142	SAMSUNG	NewDelhi	8103292041

[Download CSV](#)

Caption

6.
SELECT CID
from company
where CID = 143;

180 SELECT CID
181 from company
182 where CID = 143;
183

CID
143

Download CSV

Caption

7.
Select * from Salary;

184
185 Select * from Salary;
186

Download CSV

BASIC	HRA	TA	DA	BONUS	MEDICAL	INCREMENTS	OVER_TIME_AMOUNT
25000	2500	2000	1000	4500	2500	0	1000
35000	2100	1000	1000	3500	1700	0	2100
15000	2700	2500	1000	2500	1200	0	1700
27000	3500	2050	1000	3500	3200	0	2500
29000	500	1700	1000	1500	3100	0	1100

Download CSV
5 rows selected.

Caption

8.

Select * from Company,salary;

186
187 `Select * from Company,salary;`
188

CID	CNAME	HQADDRESS	HQPHONE	BASIC	HRA	TA	DA	BONUS	MEDICAL	INCREMENTS	OVER_TIME_AMOUNT
143	Apple	Mumbai	9826752661	25000	2500	2000	1000	4500	2500	0	1000
143	Apple	Mumbai	9826752661	35000	2100	1000	1000	3500	1700	0	2100
143	Apple	Mumbai	9826752661	15000	2700	2500	1000	2500	1200	0	1700
143	Apple	Mumbai	9826752661	27000	3500	2050	1000	3500	3200	0	2500
143	Apple	Mumbai	9826752661	29000	500	1700	1000	1500	3100	0	1100
142	SAMSUNG	NewDelhi	8103292041	25000	2500	2000	1000	4500	2500	0	1000
142	SAMSUNG	NewDelhi	8103292041	35000	2100	1000	1000	3500	1700	0	2100
142	SAMSUNG	NewDelhi	8103292041	15000	2700	2500	1000	2500	1200	0	1700
142	SAMSUNG	NewDelhi	8103292041	27000	3500	2050	1000	3500	3200	0	2500
142	SAMSUNG	NewDelhi	8103292041	29000	500	1700	1000	1500	3100	0	1100

Download CSV
0 rows selected.

Caption

NVL

SELECT Company.cname,NVL(company.HQaddress,'address not know')

FROM Company;

138 `insert into Company values('145', 'SAMSUNG',NULL,'8103292041');`
139
140 `SELECT Company.cname,NVL(company.HQaddress,'address not know')`
141
142 `FROM Company;`
143
144
145
146

CNAME	NVL (COMPANY.HQADDRESS, 'ADDRESSNOTKNOW')
SAMSUNG	NewDelhi
Apple	Mumbai
SAMSUNG	address not know

Caption

NULLIF

```
SELECT NULLIF(HQphone,0)
FROM Company;
```

142	SELECT NULLIF(HQphone,0)
143	FROM Company;
144	
145	
146	
147	

NULLIF(HQPHONE,0)
8103292041
9826752661

[Download CSV](#)

2 rows selected.

Caption

Join query ordered by clause

```
SELECT Company.cname, Branch.BID  
FROM Company  
JOIN Branch  
ON Company.CID=Branch.CID ORDER BY Branch.BID DESC;
```

```
173  
174 SELECT Company.cname, Branch.BID  
175 FROM Company  
176 JOIN Branch  
177 ON Company.CID=Branch.CID ORDER BY Branch.BID DESC;  
178  
179
```

CNAME	BID
SAMSUNG	7216
SAMSUNG	4574
SAMSUNG	1238

[Download CSV](#)

Caption

Order by

```
select basic from salary where DA = 1000 order by basic desc;
```

```
150 select basic from salary where DA = 1000 order by basic desc;  
151
```

BASIC
35000
29000

Caption

HAVING

```
SELECT Employee.ENAME, COUNT(SALARY.BASIC) AS BASE  
FROM (Salary INNER JOIN Employee ON Salary.basic = Employee.EID) GROUP BY ENAME HAVING  
COUNT(Salary.Basic) > 0;
```

```
153 SELECT Employee.ENAME, COUNT(SALARY.BASIC) AS BASE  
154 FROM (Salary INNER JOIN Employee ON Salary.basic = Employee.EID) GROUP BY ENAME HAVING COUNT(Salary.  
155  
156
```

no data found

Caption

Correlated query

```
SELECT Aname,Department  
FROM Administration  
WHERE AID IN(  
SELECT AID  
FROM Employee  
WHERE Administration.AID=Employee.AID  
);
```

```

156
157 SELECT Aname,Department
158 FROM Administration
159 WHERE AID IN(
160 SELECT AID
161 FROM Employee|
162 WHERE Administration.AID=Employee.AID
163 );
164
165

```

ANAME	DEPARTMENT
Khushi Kumar	RD

[Download CSV](#)

Caption

UnCorrelated query

```

SELECT cname
FROM Company
WHERE CID IN(
SELECT CID
FROM Branch
WHERE Branch.BID BETWEEN '1000' AND '2000');

```

```

166 SELECT cname
167 FROM Company
168 WHERE CID IN(
169 SELECT CID
170 FROM Branch
171 WHERE Branch.BID BETWEEN '1000' AND '2000');|
172
173

```

CNAME
SAMSUNG

[Download CSV](#)

Caption

left outer join

```

SELECT Administration.AID, Employee.ENAME
FROM Administration
LEFT OUTER JOIN Employee
ON Administration.AID= Employee.AID

```

```

185 SELECT Administration.AID, Employee.ENAME
186 FROM Administration
187 LEFT OUTER JOIN Employee
188 ON Administration.AID= Employee.AID
189

```

AID	ENAME
23754	Rishi Tej

[Download CSV](#)

Caption

Set query from MINUS operator

```
SELECT EID,ENAME  
FROM EMPLOYEE  
WHERE EID > 2000  
MINUS  
SELECT CID,CNAME  
FROM COMPANY  
WHERE CID > 1000  
ORDER BY 2;
```

```
138 SELECT EID,ENAME  
139 FROM EMPLOYEE  
140 WHERE EID > 2000  
141 MINUS  
142 SELECT CID,CNAME  
143 FROM COMPANY  
144 WHERE CID > 1000  
145 ORDER BY 2;
```

1 row(s) inserted.

EID	ENAME
3162561	Rishi Tej

[Download CSV](#)

Caption

PL/SQL

Define and implement two PL/SQL function involving cursor.

```
CREATE OR REPLACE FUNCTION no_of_days(id number)
RETURN NUMBER IS
joining DATE;
dateofbirth DATE;
total_age NUMBER;
CURSOR c3 is SELECT yoj FROM employee WHERE eid = id;
CURSOR c4 is SELECT dob FROM employee WHERE eid = id;
BEGIN
OPEN c3;
OPEN c4;
FETCH c3 into joining;
FETCH c4 into dateofbirth;
total_age := joining - dateofbirth;
return total_age;
CLOSE c3;
CLOSE c4;
end;
/

BEGIN
dbms_output.put_line('Total number of days ' || no_of_days(3162561));
end;
/
```

```
253 CREATE OR REPLACE FUNCTION no_of_days(id number)
254 RETURN NUMBER IS
255 joining DATE;
256 dateofbirth DATE;
257 total_age NUMBER;
258 CURSOR c3 is SELECT yoj FROM employee WHERE eid = id;
259 CURSOR c4 is SELECT dob FROM employee WHERE eid = id;
260 BEGIN
261 OPEN c3;
262 OPEN c4;
263 FETCH c3 into joining;
264 FETCH c4 into dateofbirth;
265 total_age := joining - dateofbirth;
266 return total_age;
267 CLOSE c3;
268 CLOSE c4;
269 end;
270 /
271
272 BEGIN
273 dbms_output.put_line('Total number of days ' || no_of_days(3162561));
274 end;
275 /
```

Caption

```

264  FETCH c4 into dateofbirth;
265  total_age := joining - dateofbirth;
266  return total_age;
267  CLOSE c3;
268  CLOSE c4;
269  end;
270  /
271
272  BEGIN
273  dbms_output.put_line('Total number of days ' || no_of_days(3162561));
274  end;
275  /

```

Function created.

Statement processed.

Total number of days -729

Caption

2.

```

CREATE OR REPLACE FUNCTION tsalary(id number)
RETURN NUMBER IS
basicsalary NUMBER;
hrasalary NUMBER;
total_salary_fund NUMBER;
CURSOR c5 is SELECT basic FROM salary WHERE basic = id;
CURSOR c6 is SELECT hra FROM salary WHERE basic = id;
BEGIN
OPEN c5;
OPEN c6;
FETCH c5 into basicsalary;
FETCH c6 into hrasalary;
total_salary_fund := basicsalary + hrasalary;
return total_salary_fund;
CLOSE c5;
CLOSE c6;
end;
/

```

```

BEGIN
dbms_output.put_line('Total salary fund ' || tsalary(25000));
end;
/

```

SQL Worksheet

```

279 RETURN NUMBER IS
280 basicsalary NUMBER;
281 hrasalary NUMBER;
282 total_salary_fund NUMBER;
283 CURSOR c5 is SELECT basic FROM salary WHERE basic = id;
284 CURSOR c6 is SELECT hra FROM salary WHERE basic = id;
285 BEGIN
286 OPEN c5;
287 OPEN c6;
288 FETCH c5 INTO basicsalary;
289 FETCH c6 INTO hrasalary;
290 total_salary_fund := basicsalary + hrasalary;
291 return total_salary_fund;
292 CLOSE c5;
293 CLOSE c6;
294 end;
295 /
296
297 BEGIN
298 dbms_output.put_line('Total salary fund ' || tsalary(25000));
299 end;
300 /
301

```

```

222 FETCH c6 INTO hrasalary; Caption
223 total_salary_fund := basicsalary + hrasalary;
224 return total_salary_fund;
225 CLOSE c5;
226 CLOSE c6;
227 end;
228 /
229
230 BEGIN
231 dbms_output.put_line('Total salary fund ' || tsalary(25000));
232 end;
233 /
234
235

```

Function created.

Statement processed.

Total salary fund 27500

Caption

Two PL/SQL procedure involving cursor for the database under consideration (i. e. required for the project).

```
CREATE OR REPLACE procedure details(employeeid NUMBER) is
CURSOR c1 is SELECT * from Employee WHERE eid = employeeid;
r_c1 c1%ROWTYPE;
begin
open c1;
fetch c1 into r_c1;
DBMS_OUTPUT.PUT_LINE('employee name: '||r_c1.ename);
DBMS_OUTPUT.PUT_LINE('employee department: '||r_c1.designation);
close c1;
end;
/
call details(3162561);
```

```
238
239 CREATE OR REPLACE procedure details(employeeid NUMBER) is
240 CURSOR c1 is SELECT * from Employee WHERE eid = employeeid;
241 r_c1 c1%ROWTYPE;
242 begin
243 open c1;
244 fetch c1 into r_c1;
245 DBMS_OUTPUT.PUT_LINE('employee name: '||r_c1.ename);
246 DBMS_OUTPUT.PUT_LINE('employee department: '||r_c1.designation);
247 close c1;
248 end;
249 /
250 call details(3162561);
251
```

```
statement processed.
employee name: Rishi Tej
employee department: Ass. Manager
```

Caption


```

CREATE OR REPLACE procedure company_branch(id NUMBER) is
CURSOR c2 is SELECT * from branch WHERE cid = id;
r_c2 c2%ROWTYPE;
begin
open c2;
fetch c2 into r_c2;
DBMS_OUTPUT.PUT_LINE('branch name: '||r_c2.bname);
DBMS_OUTPUT.PUT_LINE('branch address: '||r_c2.baddress);
DBMS_OUTPUT.PUT_LINE('branch phone: '||r_c2.bphone);
close c2;
end;
/
call company_branch('142');

```

```

168
169 CREATE OR REPLACE procedure company_branch(id NUMBER) is
170 CURSOR c2 is SELECT * from branch WHERE cid = id;
171 r_c2 c2%ROWTYPE;
172 begin
173 open c2;
174 fetch c2 into r_c2;
175 DBMS_OUTPUT.PUT_LINE('branch name: '||r_c2.bname);
176 DBMS_OUTPUT.PUT_LINE('branch address: '||r_c2.baddress);
177 DBMS_OUTPUT.PUT_LINE('branch phone: '||r_c2.bphone);
178 close c2;
179 end;
180 /
181 call company_branch('142');

```

Procedure created.

Statement processed.

branch name: nokia

branch address: Bangalore

branch phone: 2345678999

Caption

Thank you!