2110206027 **2110206009** **2110206056**

Mahmut Atsız **Berkay Karabulut** **Mert Taban**

Database Project

SCENARIO

Acme Corporation is a rapidly growing company with over 500 employees across various departments. They currently manage payroll manually using spreadsheets, which is becoming increasingly time-consuming and prone to errors. To streamline payroll processing, improve accuracy, and gain better insights into employee compensation, Acme Corporation decides to develop a new payroll management system.

Entity-1	Entity-2	Entity-3	Entity-4	Entity-5	Entity-6	Entity-7	Entity-8	Entity-9
Employees	JoiningDetails	Departments	Posts	BasicPay	Allowances	Deductions	Leaves	Salaries

ERD DEVELOPİNG

Entities

1)Employees

Employee information is entered into the system upon joining the company. This data is updated as needed throughout their employment.

2)Joining Details

Joining details are recorded when a new employee joins the company. They may be updated if the probation period changes.

3)Departments

Departments are created within the system to categorize employees based on their function within the company. Department names or locations may be updated as needed.

4)Posts

Posts are created to reflect the different job titles employees hold at Acme. Post names may be updated to reflect changes in job titles or responsibilities.

5)Basic Pay

Basic pay rates are established for each post. These rates may be updated periodically to reflect changes in compensation or cost of living.

6)Allowances

Allowances are configured within the system to reflect the different benefits offered to employees. These allowances may be updated or added as needed.

7)Deductions

Deductions are defined within the system to reflect the different withholdings from employee salaries. These deductions may be updated to comply with changes in tax regulations or employee benefit plans.

8)Leaves

Leaves are recorded within the system when employees take time off. This information may be used for various purposes, such as tracking leave trends or managing staffing levels.

9)Salaries

Salaries are automatically calculated and processed within the system for each pay period. This ensures timely and accurate employee compensation.

Attributes

	Employees	JoiningDetails	Department	Posts	BasicPay	Allowances	Deductions	Leaves	Salaries
Attribute 1	EmployeeID (Primary Key)	JoiningDetailesID (Primay Key)	DeptID (Primary Key)	PostID (Primary Key)	BasicPayID (Primary Key)	AllowanceID (Primary Key)	DeductionID (Primary Key)	LeaveID (Primary Key)	SalaryID (Primary Key)
Attribute 2	DeptID (Foreıgn key)	EmployeeID (Foreign key)	Department Name	EmployeeID (Foreign Key)	JoiningDetailesID (Foreign Key)	Allowance Name	LeaveID (Foreign Key)	Employee ID (Foreign Key)	Employee ID (Foreign Key)
Attribute 3	Name	Joining Date	Location	SalaryID (Foreign Key)	WorkHours	AllowanceAmount	Deduction Name	Start Date	BasicPayID (Foreign Key)
Attribute 4	Contact Info	JobType	ManagerName		PerHourCost	AlloweanceCheck	Description	End Date	AllowanceID (Foreign Key)
Attribute 5	Adress		Manager Email				DeductionAmount		DeductionID (Foreign Key)
Attribute 6	Emergency Contact		AccountentName						Gross Pay
Attribute 7			AccountentEmail						Net Pay
Attribute 8									Payment Date

Matrix Diagram

	Employees	JoiningDetails	Department	Posts	BasicPay	Allowances	Deductions	Leaves	Salaries
Employees				Receives wage by					
JoiningDetails					Used for				
Department				Send wage with					Informed to
Posts	The wage to		The wage by						
BasicPay		Changed by							Transferred to
Allowances									Transferred to
Deductions								Increase due to	Informed to
Leaves							Comes out		
Salaries			Calculated for		Receive transfer from	Receive transfer from	Reduce due to		

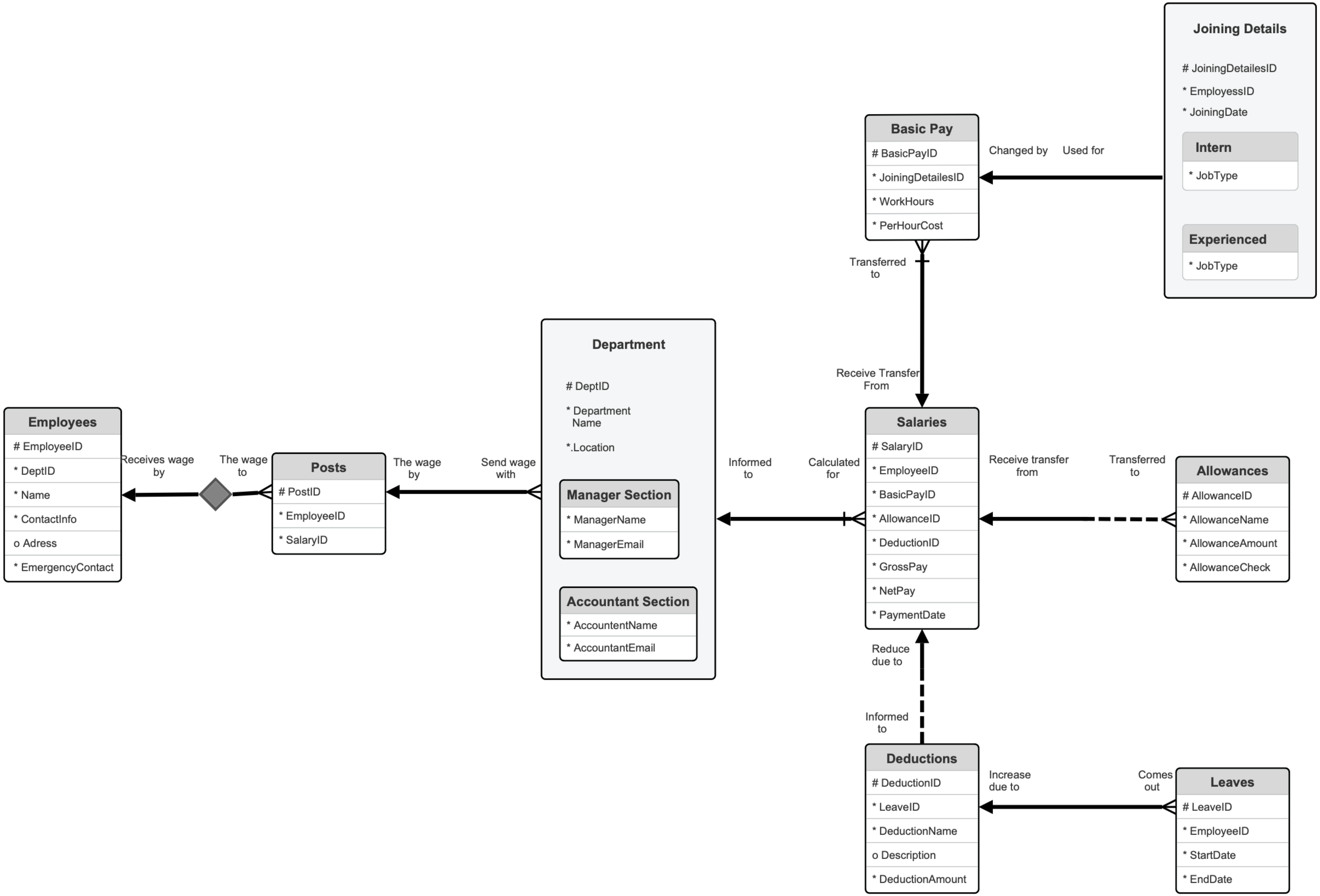
Supertype-Subtype

We can consider the 'Departments' entity as a supertype for the 'Manager Section' and 'Accountant Section' subtype entities. Same idea valid for Joining Details too (‘Joining Details’ supertype, and ‘Intern’ and ‘Experienced’ subtype entities.

Non-Transferable

We can consider the relationship between the 'Employees' and 'Posts' entities as a non-transferable relationship. After the post is assigned to an employee, it is exclusive to that employee and cannot be transferred to another employee.

ERD-Diagram


Joining Details
# JoiningDetailesID
* EmployessID
* JoiningDate
Intern
* JobType
Experienced
* JobType
Basic Pay
# BasicPayID
* JoiningDetailesID
* WorkHours
* PerHourCost
Changed by
Used for
Transferred to
Receive Transfer From
Department
# DeptID
* Department Name
*.Location
Manager Section
* ManagerName
* ManagerEmail
Accountant Section
* AccountentName
* AccountantEmail
Employees
# EmployeeID
* DeptID
* Name
* ContactInfo
o Adress
* EmergencyContact
Receives wage by
The wage to
Posts
# PostID
* EmployeeID
* SalaryID
The wage by
Send wage with
Informed to
Calculated for
Salaries
# SalaryID
* EmployeeID
* BasicPayID
* AllowanceID
* DeductionID
* GrossPay
* NetPay
* PaymentDate
Receive transfer from
Transferred to
Allowances
# AllowanceID
* AllowanceName
* AllowanceAmount
* AllowanceCheck
Reduce due to
Informed to
Deductions
# DeductionID
* LeaveID
* DeductionName
o Description
* DeductionAmount
Increase due to
Comes out
Leaves
# LeaveID
* EmployeeID
* StartDate
* EndDate

SQL DDL STATMENTS

```
CREATE TABLE Department (
    DeptID NUMBER PRIMARY KEY,
    DepartmentName VARCHAR2(100),
    Location VARCHAR2(100),

    ManagerName VARCHAR2(100),
    ManagerEmail VARCHAR2(100),

    AccountantName VARCHAR2(100),
    AccountantEmail VARCHAR2(100)
);



CREATE TABLE Employees (
    EmployeeID NUMBER PRIMARY KEY,
    DeptID NUMBER,
    Name VARCHAR2(100),
    ContactInfo VARCHAR2(100),
    Address VARCHAR2(100),
    EmergencyContact VARCHAR2(100),
    CONSTRAINT fk_DeptID FOREIGN KEY (DeptID) REFERENCES Department(DeptID)
);



CREATE TABLE JoiningDetails (
    JoiningDetailsID NUMBER PRIMARY KEY,
    EmployeeID NUMBER,
    JoiningDate DATE,
    JobType VARCHAR2(100),
    CONSTRAINT fk_EmployeeID FOREIGN KEY (EmployeeID) REFERENCES
Employees(EmployeeID)
);
```

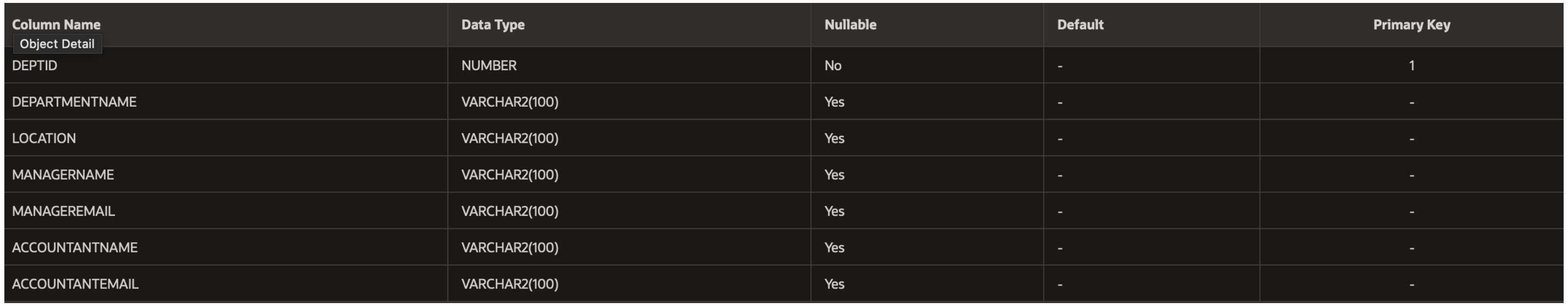
```
CREATE TABLE BasicPay (
    BasicPayID NUMBER PRIMARY KEY,
    JoiningDetailsID NUMBER,
    WorkHours NUMBER,
    PerHourCost NUMBER,
    CONSTRAINT fk_JoiningDetailsID FOREIGN KEY (JoiningDetailsID) REFERENCES JoiningDetails(JoiningDetailsID)
);


CREATE TABLE Allowances (
    AllowanceID NUMBER PRIMARY KEY,
    AllowanceName VARCHAR2(100),
    AllowanceAmount NUMBER,
    AllowanceCheck VARCHAR2(10)
);



CREATE TABLE Leaves (
    LeaveID NUMBER PRIMARY KEY,
    EmployeeID NUMBER,
    StartDate DATE,
    EndDate DATE
);


CREATE TABLE Deductions (
    DeductionID NUMBER PRIMARY KEY,
    LeaveID NUMBER,
    DeductionName VARCHAR2(100),
    Description VARCHAR2(255),
    DeductionAmount NUMBER,
    CONSTRAINT fk_LeaveID FOREIGN KEY (LeaveID) REFERENCES
Leaves(LeaveID)
);
```

```
CREATE TABLE Salaries (
    SalaryID NUMBER PRIMARY KEY,
    EmployeeID NUMBER,
    BasicPayID NUMBER,
    AllowanceID NUMBER,
    DeductionID NUMBER,
    GrossPay NUMBER,
    NetPay NUMBER,
    PaymentDate DATE,
    CONSTRAINT fk_EmployeeID FOREIGN KEY (EmployeeID) REFERENCES Employees(EmployeeID),
    CONSTRAINT fk_BasicPayID FOREIGN KEY (BasicPayID) REFERENCES BasicPay(BasicPayID),
    CONSTRAINT fk_AllowanceID FOREIGN KEY (AllowanceID) REFERENCES Allowances(AllowanceID),
    CONSTRAINT fk_DeductionID FOREIGN KEY (DeductionID) REFERENCES Deductions(DeductionID)
);
```

```
CREATE TABLE Posts (
    PostID NUMBER PRIMARY KEY,
    EmployeeID NUMBER,
    SalaryID NUMBER,
    CONSTRAINT fk_EmployeeID FOREIGN KEY (EmployeeID) REFERENCES Employees(EmployeeID),
    CONSTRAINT fk_SalaryID FOREIGN KEY (SalaryID) REFERENCES Salaries(SalaryID)
);
```

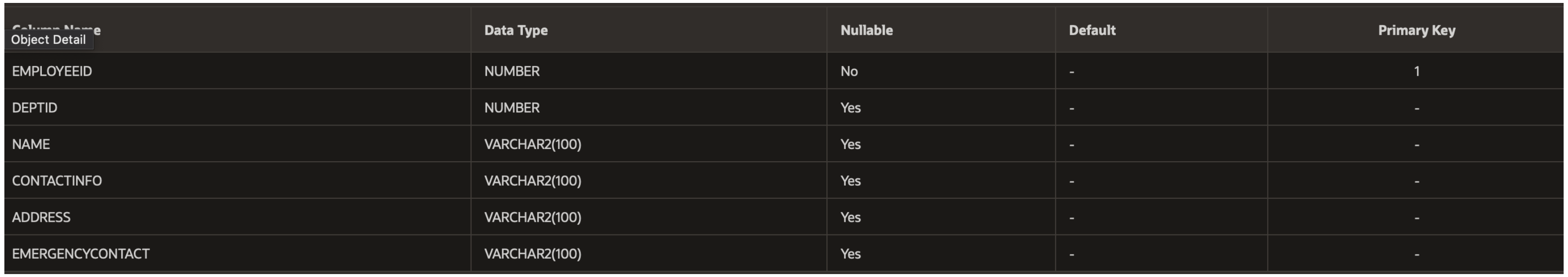
PHYSICAL DATABASE

Department:



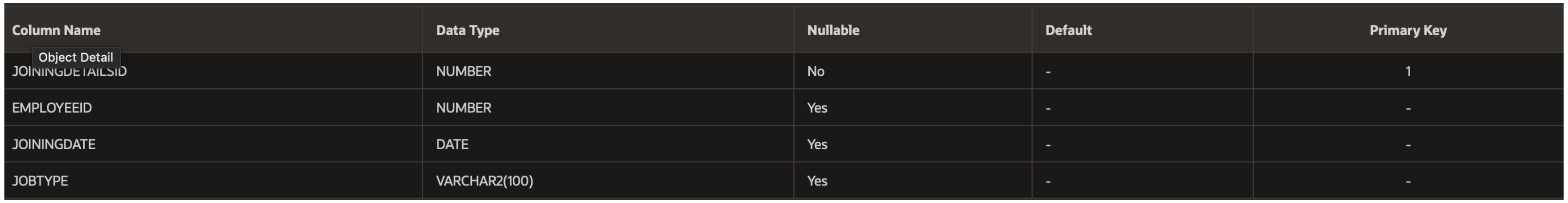
Column Name	Data Type	Nullable	Default	Primary Key
DEPTID	NUMBER	No	-	1
DEPARTMENTNAME	VARCHAR2(100)	Yes	-	-
LOCATION	VARCHAR2(100)	Yes	-	-
MANAGERNAME	VARCHAR2(100)	Yes	-	-
MANAGEREMAIL	VARCHAR2(100)	Yes	-	-
ACCOUNTANTNAME	VARCHAR2(100)	Yes	-	-
ACCOUNTANTEMAIL	VARCHAR2(100)	Yes	-	-

Employees:



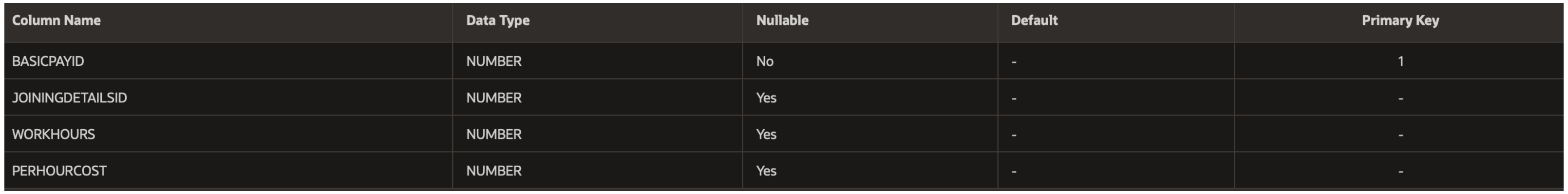
Column Name	Data Type	Nullable	Default	Primary Key
EMPLOYEEID	NUMBER	No	-	1
DEPTID	NUMBER	Yes	-	-
NAME	VARCHAR2(100)	Yes	-	-
CONTACTINFO	VARCHAR2(100)	Yes	-	-
ADDRESS	VARCHAR2(100)	Yes	-	-
EMERGENCYCONTACT	VARCHAR2(100)	Yes	-	-

Joining Details:



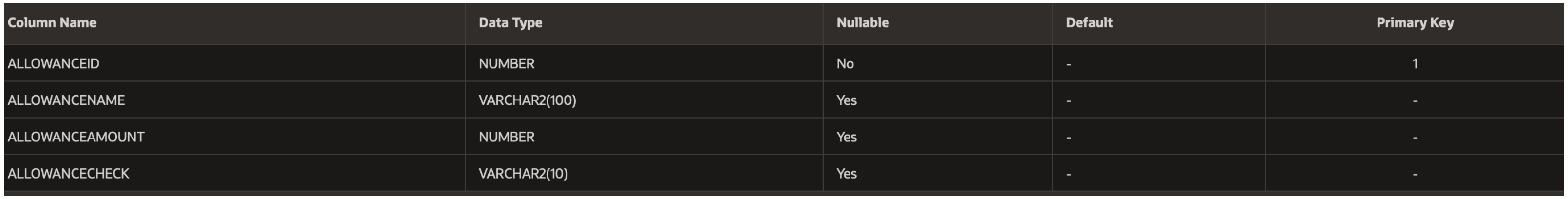
Column Name	Data Type	Nullable	Default	Primary Key
JOININGDETAILSID	NUMBER	No	-	1
EMPLOYEEID	NUMBER	Yes	-	-
JOININGDATE	DATE	Yes	-	-
JOBTYPE	VARCHAR2(100)	Yes	-	-

Basic Pay:



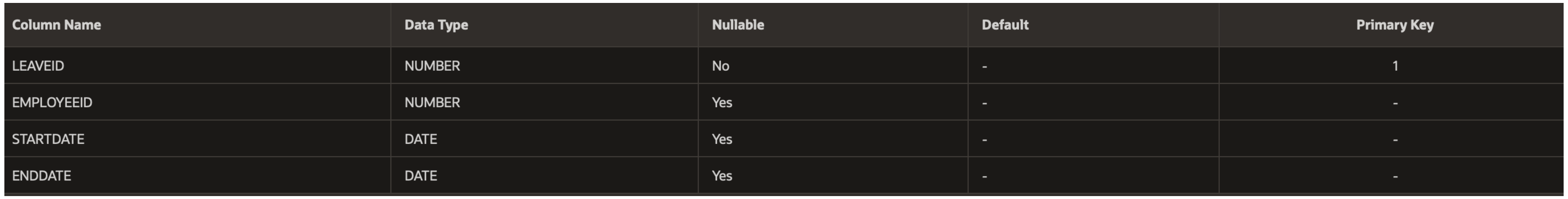
Column Name	Data Type	Nullable	Default	Primary Key
BASICPAYID	NUMBER	No	-	1
JOININGDETAILSID	NUMBER	Yes	-	-
WORKHOURS	NUMBER	Yes	-	-
PERHOURCOST	NUMBER	Yes	-	-

Allowances:



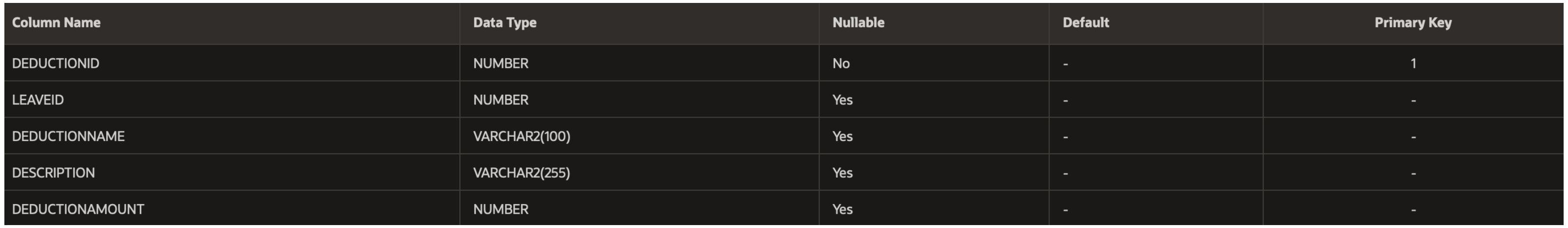
Column Name	Data Type	Nullable	Default	Primary Key
ALLOWANCEID	NUMBER	No	-	1
ALLOWANCENAME	VARCHAR2(100)	Yes	-	-
ALLOWANCEAMOUNT	NUMBER	Yes	-	-
ALLOWANCECHECK	VARCHAR2(10)	Yes	-	-

Leaves:



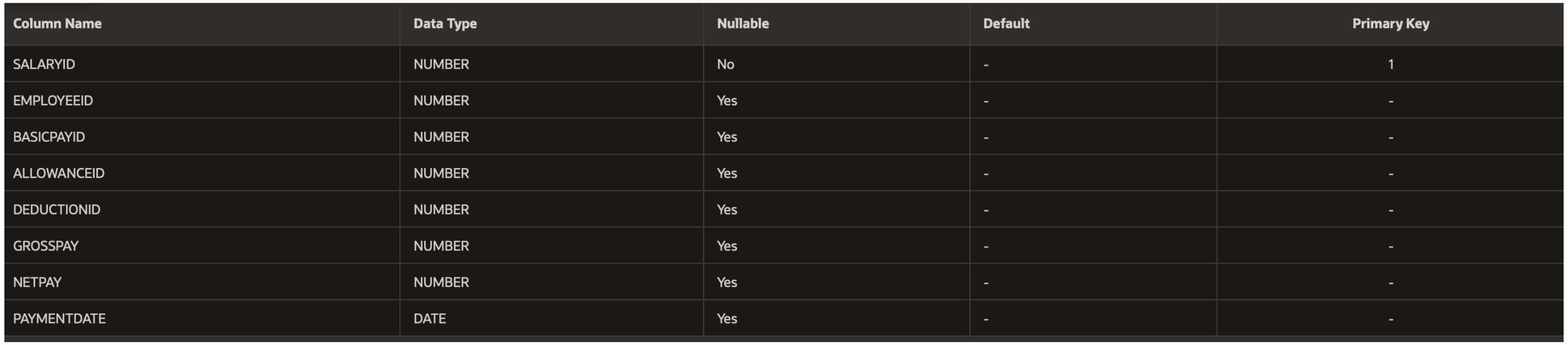
Column Name	Data Type	Nullable	Default	Primary Key
LEAVEID	NUMBER	No	-	1
EMPLOYEEID	NUMBER	Yes	-	-
STARTDATE	DATE	Yes	-	-
ENDDATE	DATE	Yes	-	-

Deductions:



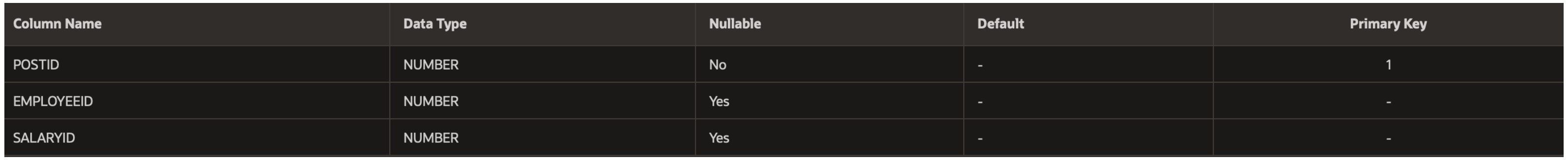
Column Name	Data Type	Nullable	Default	Primary Key
DEDUCTIONID	NUMBER	No	-	1
LEAVEID	NUMBER	Yes	-	-
DEDUCTIONNAME	VARCHAR2(100)	Yes	-	-
DESCRIPTION	VARCHAR2(255)	Yes	-	-
DEDUCTIONAMOUNT	NUMBER	Yes	-	-

Salaries:



Column Name	Data Type	Nullable	Default	Primary Key
SALARYID	NUMBER	No	-	1
EMPLOYEEID	NUMBER	Yes	-	-
BASICPAYID	NUMBER	Yes	-	-
ALLOWANCEID	NUMBER	Yes	-	-
DEDUCTIONID	NUMBER	Yes	-	-
GROSSPAY	NUMBER	Yes	-	-
NETPAY	NUMBER	Yes	-	-
PAYMENTDATE	DATE	Yes	-	-

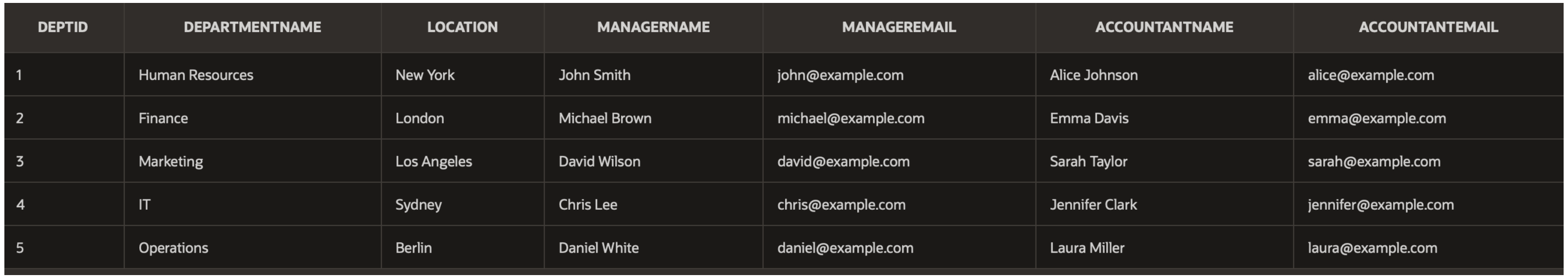
Posts:



Column Name	Data Type	Nullable	Default	Primary Key
POSTID	NUMBER	No	-	1
EMPLOYEEID	NUMBER	Yes	-	-
SALARYID	NUMBER	Yes	-	-

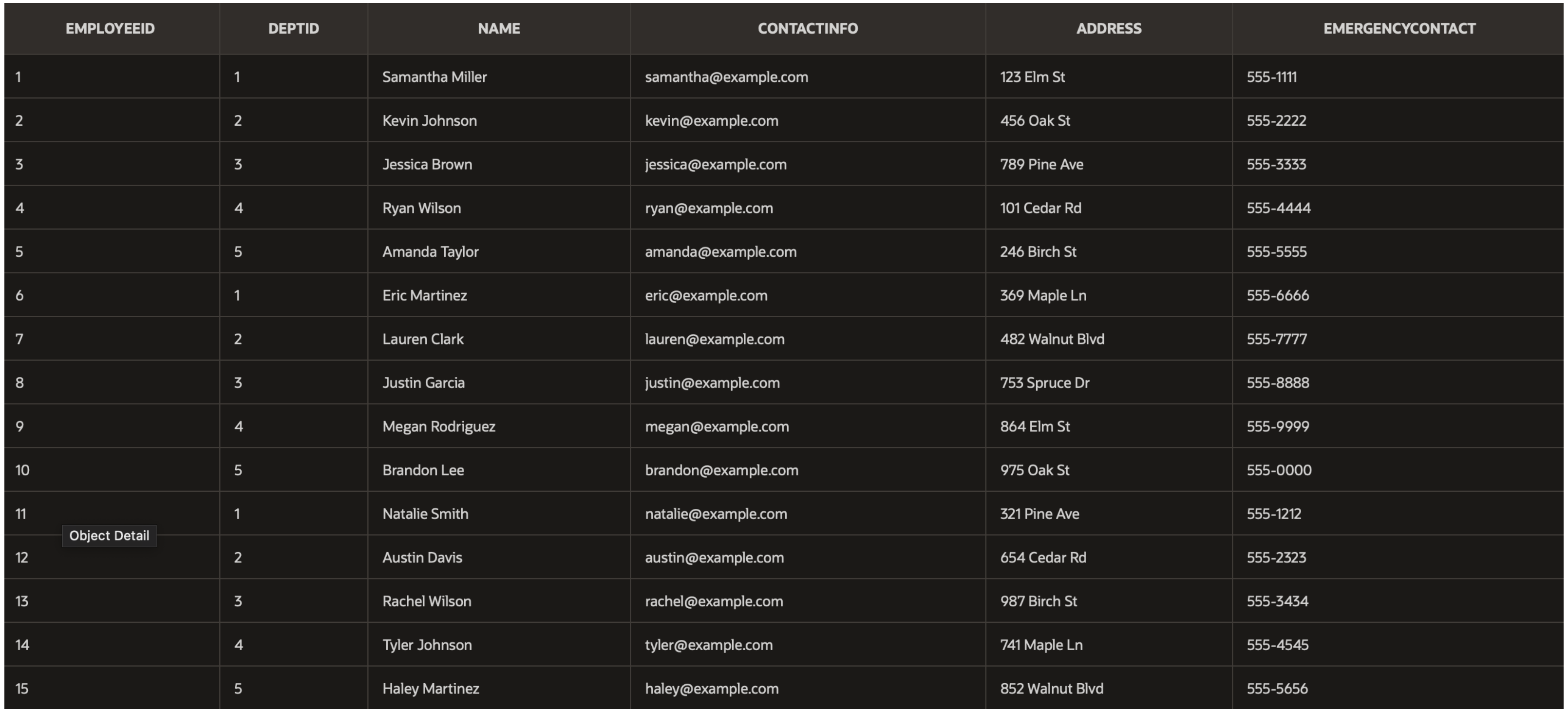
TABLE INSTANCES

Department:



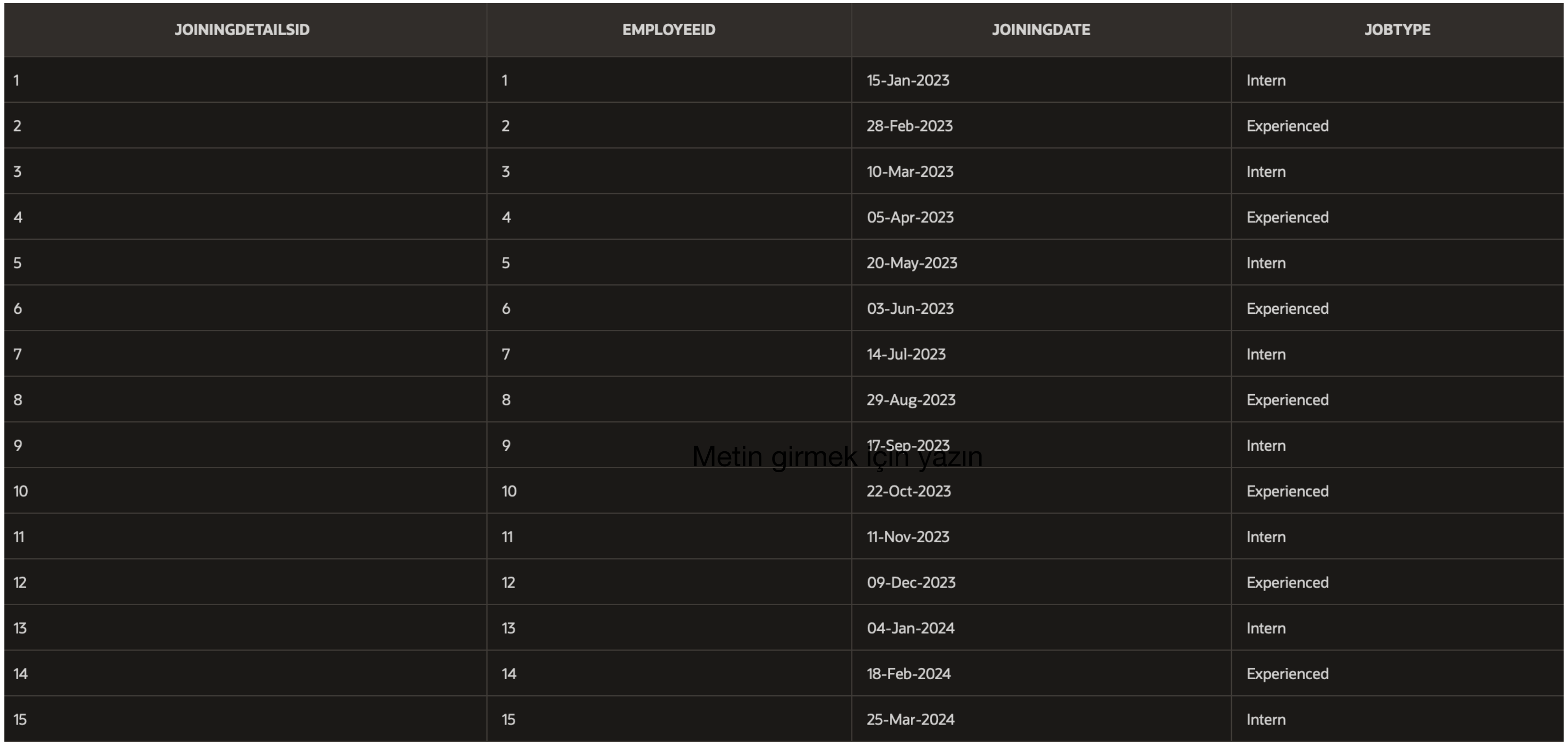
DEPTID	DEPARTMENTNAME	LOCATION	MANAGERNAME	MANAGEREMAIL	ACCOUNTANTNAME	ACCOUNTANTEMAIL
1	Human Resources	New York	John Smith	john@example.com	Alice Johnson	alice@example.com
2	Finance	London	Michael Brown	michael@example.com	Emma Davis	emma@example.com
3	Marketing	Los Angeles	David Wilson	david@example.com	Sarah Taylor	sarah@example.com
4	IT	Sydney	Chris Lee	chris@example.com	Jennifer Clark	jennifer@example.com
5	Operations	Berlin	Daniel White	daniel@example.com	Laura Miller	laura@example.com

Employees:



EMPLOYEEID	DEPTID	NAME	CONTACTINFO	ADDRESS	EMERGENCYCONTACT
1	1	Samantha Miller	samantha@example.com	123 Elm St	555-1111
2	2	Kevin Johnson	kevin@example.com	456 Oak St	555-2222
3	3	Jessica Brown	jessica@example.com	789 Pine Ave	555-3333
4	4	Ryan Wilson	ryan@example.com	101 Cedar Rd	555-4444
5	5	Amanda Taylor	amanda@example.com	246 Birch St	555-5555
6	1	Eric Martinez	eric@example.com	369 Maple Ln	555-6666
7	2	Lauren Clark	lauren@example.com	482 Walnut Blvd	555-7777
8	3	Justin Garcia	justin@example.com	753 Spruce Dr	555-8888
9	4	Megan Rodriguez	megan@example.com	864 Elm St	555-9999
10	5	Brandon Lee	brandon@example.com	975 Oak St	555-0000
11	1	Natalie Smith	natalie@example.com	321 Pine Ave	555-1212
12	2	Austin Davis	austin@example.com	654 Cedar Rd	555-2323
13	3	Rachel Wilson	rachel@example.com	987 Birch St	555-3434
14	4	Tyler Johnson	tyler@example.com	741 Maple Ln	555-4545
15	5	Haley Martinez	haley@example.com	852 Walnut Blvd	555-5656

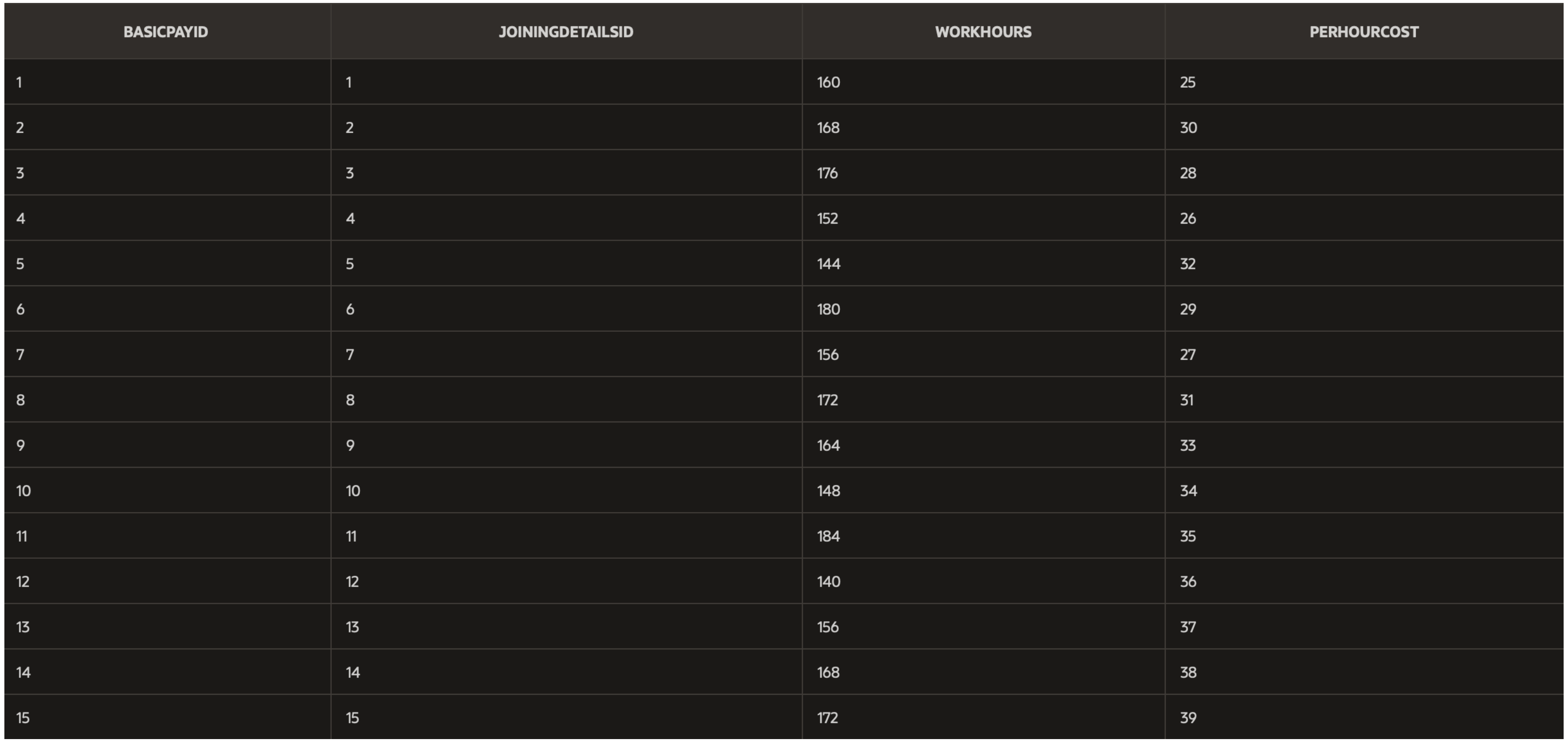
Joining Details:



JOININGDETAILSID	EMPLOYEEID	JOININGDATE	JOBTYPE
1	1	15-Jan-2023	Intern
2	2	28-Feb-2023	Experienced
3	3	10-Mar-2023	Intern
4	4	05-Apr-2023	Experienced
5	5	20-May-2023	Intern
6	6	03-Jun-2023	Experienced
7	7	14-Jul-2023	Intern
8	8	29-Aug-2023	Experienced
9	9	17-Sep-2023	Intern
10	10	22-Oct-2023	Experienced
11	11	11-Nov-2023	Intern
12	12	09-Dec-2023	Experienced
13	13	04-Jan-2024	Intern
14	14	18-Feb-2024	Experienced
15	15	25-Mar-2024	Intern

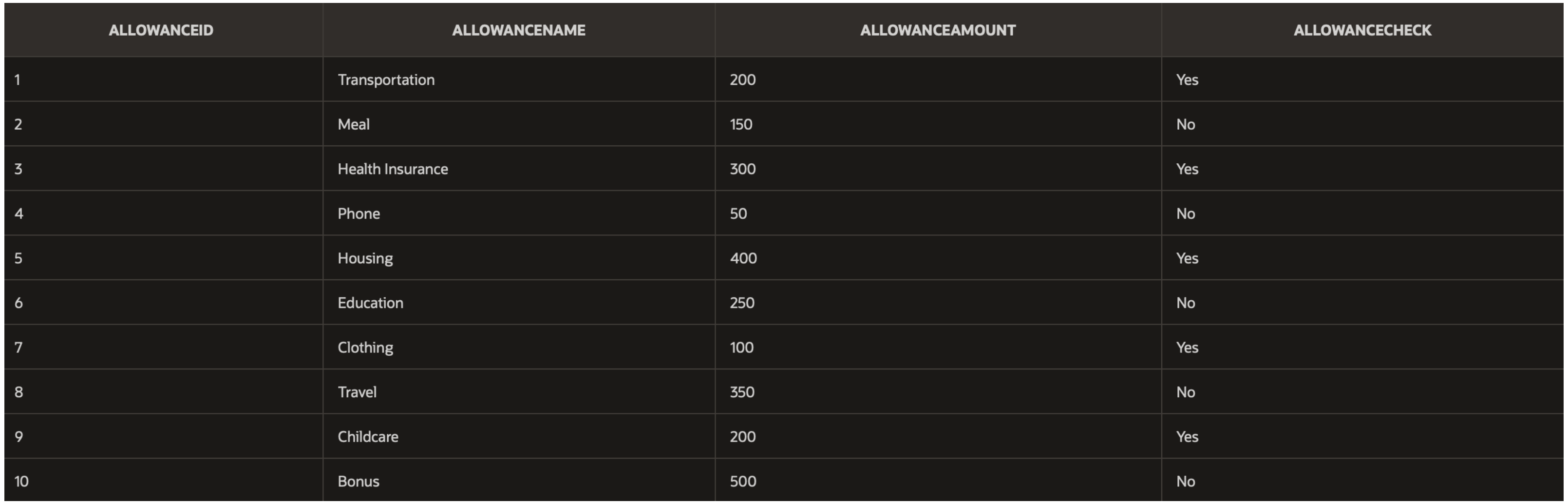
Metin girmek için yazın

Basic Pay:



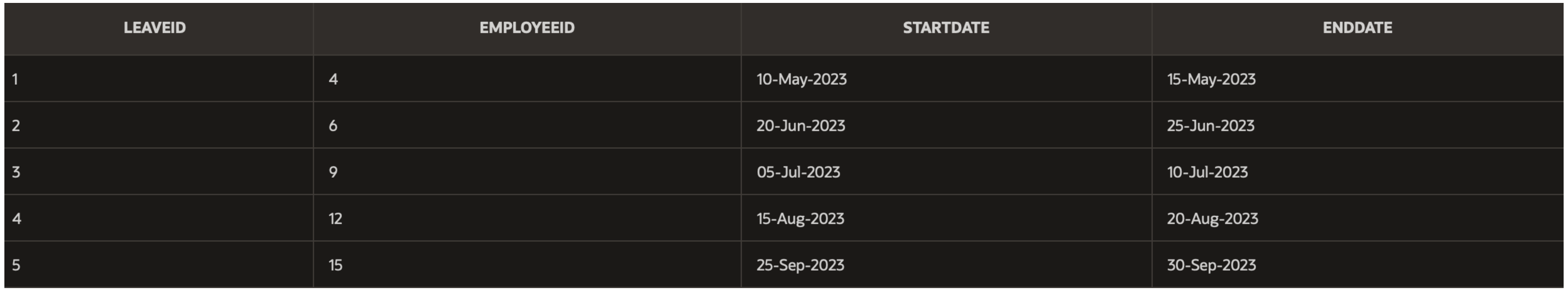
BASICPAYID	JOININGDETAILSID	WORKHOURS	PERHOURCOST
1	1	160	25
2	2	168	30
3	3	176	28
4	4	152	26
5	5	144	32
6	6	180	29
7	7	156	27
8	8	172	31
9	9	164	33
10	10	148	34
11	11	184	35
12	12	140	36
13	13	156	37
14	14	168	38
15	15	172	39

Allowances:



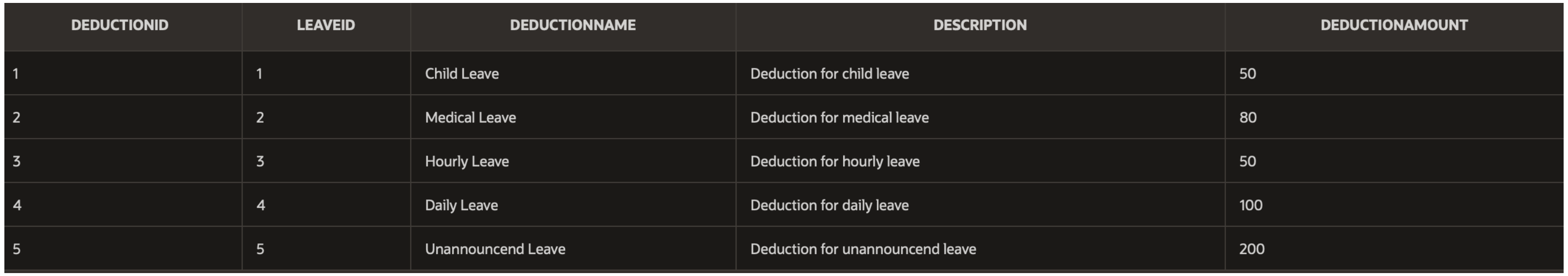
ALLOWANCEID	ALLOWANCENAME	ALLOWANCEAMOUNT	ALLOWANCECHECK
1	Transportation	200	Yes
2	Meal	150	No
3	Health Insurance	300	Yes
4	Phone	50	No
5	Housing	400	Yes
6	Education	250	No
7	Clothing	100	Yes
8	Travel	350	No
9	Childcare	200	Yes
10	Bonus	500	No

Leaves:



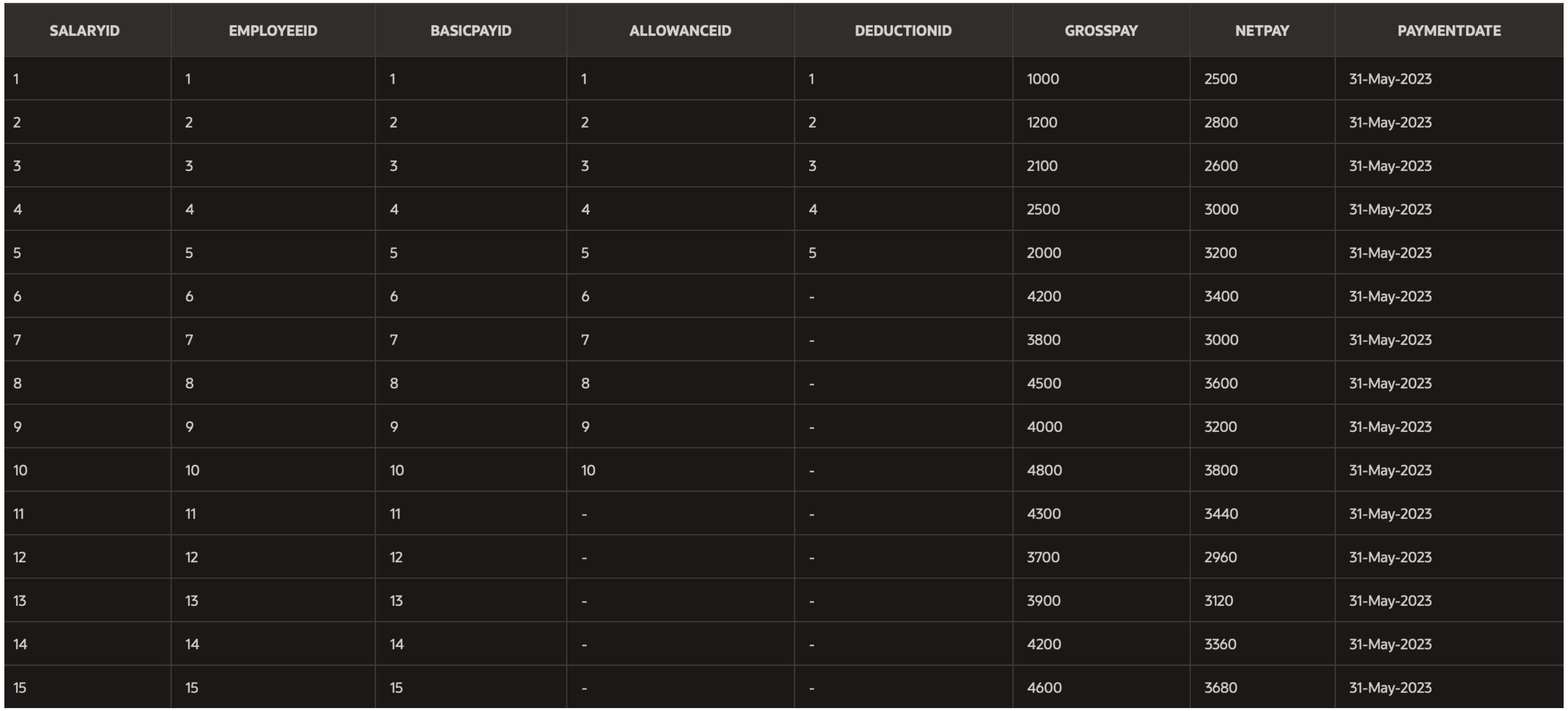
LEAVEID	EMPLOYEEID	STARTDATE	ENDDATE
1	4	10-May-2023	15-May-2023
2	6	20-Jun-2023	25-Jun-2023
3	9	05-Jul-2023	10-Jul-2023
4	12	15-Aug-2023	20-Aug-2023
5	15	25-Sep-2023	30-Sep-2023

Deductions:



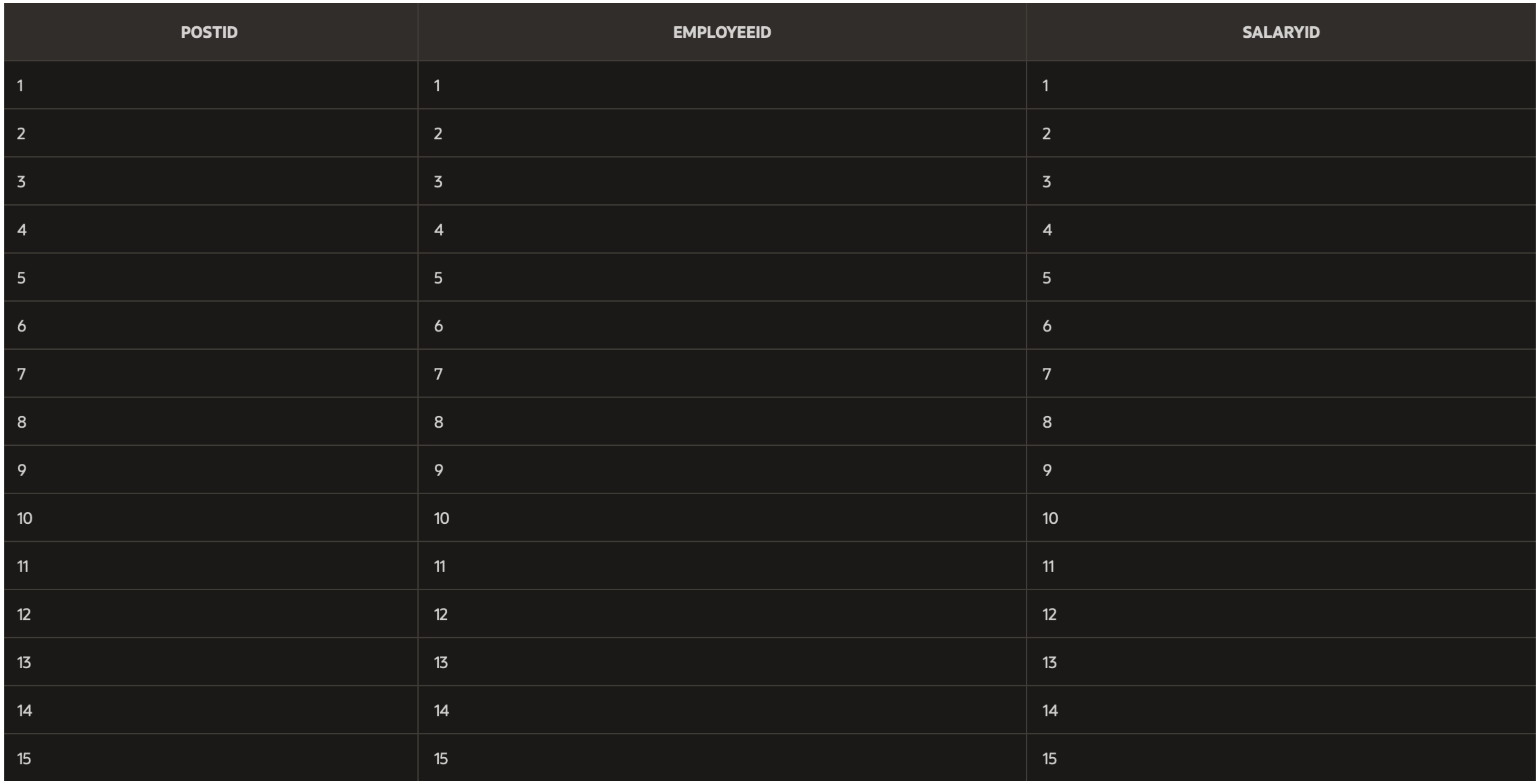
DEDUCTIONID	LEAVEID	DEDUCTIONNAME	DESCRIPTION	DEDUCTIONAMOUNT
1	1	Child Leave	Deduction for child leave	50
2	2	Medical Leave	Deduction for medical leave	80
3	3	Hourly Leave	Deduction for hourly leave	50
4	4	Daily Leave	Deduction for daily leave	100
5	5	Unannouncend Leave	Deduction for unannouncend leave	200

Salaries:



SALARYID	EMPLOYEEID	BASICPAYID	ALLOWANCEID	DEDUCTIONID	GROSSPAY	NETPAY	PAYMENTDATE
1	1	1	1	1	1000	2500	31-May-2023
2	2	2	2	2	1200	2800	31-May-2023
3	3	3	3	3	2100	2600	31-May-2023
4	4	4	4	4	2500	3000	31-May-2023
5	5	5	5	5	2000	3200	31-May-2023
6	6	6	6	-	4200	3400	31-May-2023
7	7	7	7	-	3800	3000	31-May-2023
8	8	8	8	-	4500	3600	31-May-2023
9	9	9	9	-	4000	3200	31-May-2023
10	10	10	10	-	4800	3800	31-May-2023
11	11	11	-	-	4300	3440	31-May-2023
12	12	12	-	-	3700	2960	31-May-2023
13	13	13	-	-	3900	3120	31-May-2023
14	14	14	-	-	4200	3360	31-May-2023
15	15	15	-	-	4600	3680	31-May-2023

Posts:

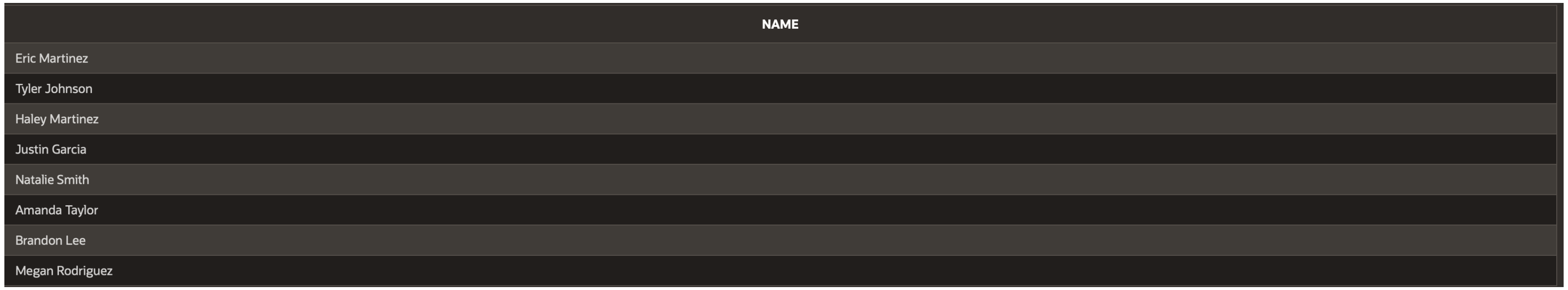


POSTID	EMPLOYEEID	SALARYID
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5
6	6	6
7	7	7
8	8	8
9	9	9
10	10	10
11	11	11
12	12	12
13	13	13
14	14	14
15	15	15

DML COMMANDS

Subquery Statement:

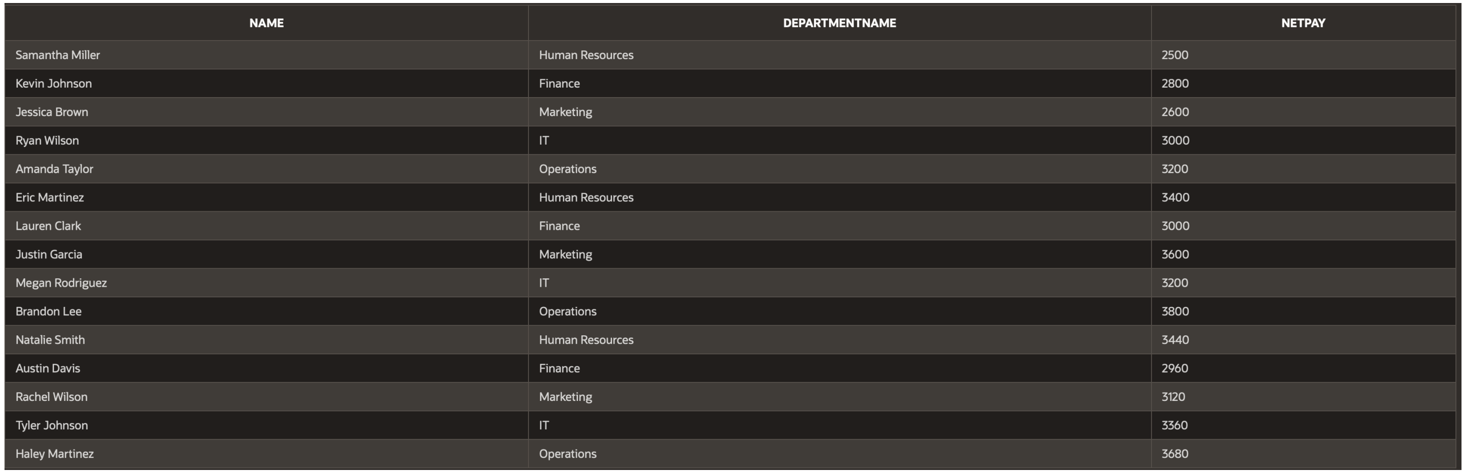
```
SELECT Name
FROM Employees
WHERE EmployeeID IN (SELECT EmployeeID FROM Salaries WHERE NetPay > (SELECT AVG(NetPay) FROM Salaries));
```



NAME
Eric Martinez
Tyler Johnson
Haley Martinez
Justin Garcia
Natalie Smith
Amanda Taylor
Brandon Lee
Megan Rodriguez

Join Statement:

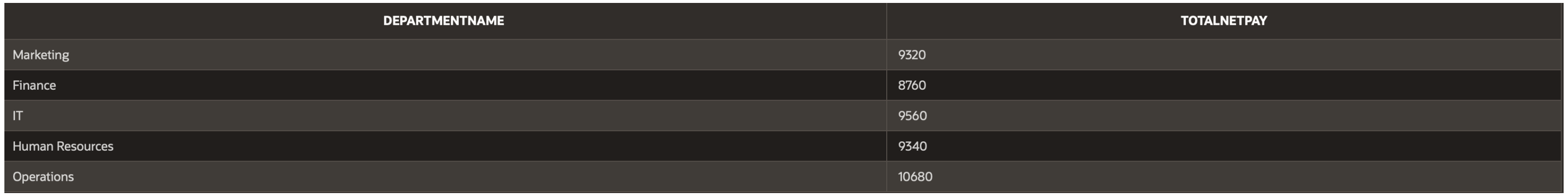
```
SELECT E.Name, D.DepartmentName, S.NetPay
FROM Employees E
JOIN Department D ON E.DeptID = D.DeptID
JOIN Salaries S ON E.EmployeeID = S.EmployeeID;
```



NAME	DEPARTMENTNAME	NETPAY
Samantha Miller	Human Resources	2500
Kevin Johnson	Finance	2800
Jessica Brown	Marketing	2600
Ryan Wilson	IT	3000
Amanda Taylor	Operations	3200
Eric Martinez	Human Resources	3400
Lauren Clark	Finance	3000
Justin Garcia	Marketing	3600
Megan Rodriguez	IT	3200
Brandon Lee	Operations	3800
Natalie Smith	Human Resources	3440
Austin Davis	Finance	2960
Rachel Wilson	Marketing	3120
Tyler Johnson	IT	3360
Haley Martinez	Operations	3680

Group By Statement:

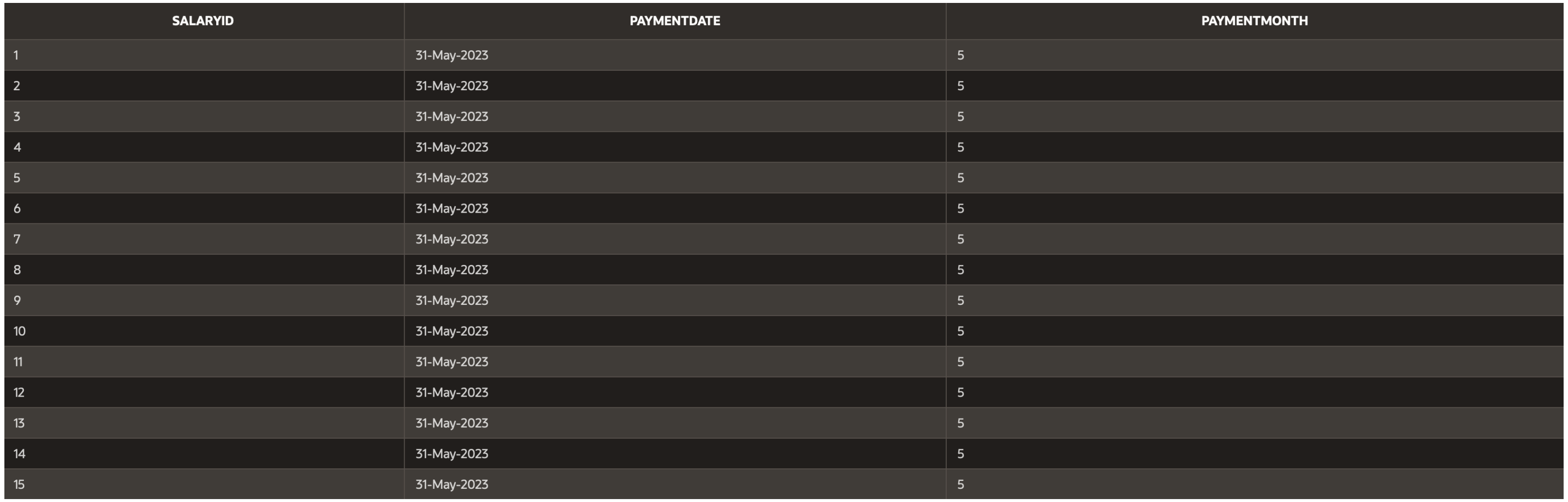
```
SELECT D.DepartmentName, SUM(S.NetPay) AS TotalNetPay
FROM Employees E
JOIN Department D ON E.DeptID = D.DeptID
JOIN Salaries S ON E.EmployeeID = S.EmployeeID
GROUP BY D.DepartmentName;
```



DEPARTMENTNAME	TOTALNETPAY
Marketing	9320
Finance	8760
IT	9560
Human Resources	9340
Operations	10680

Date Function Statement:

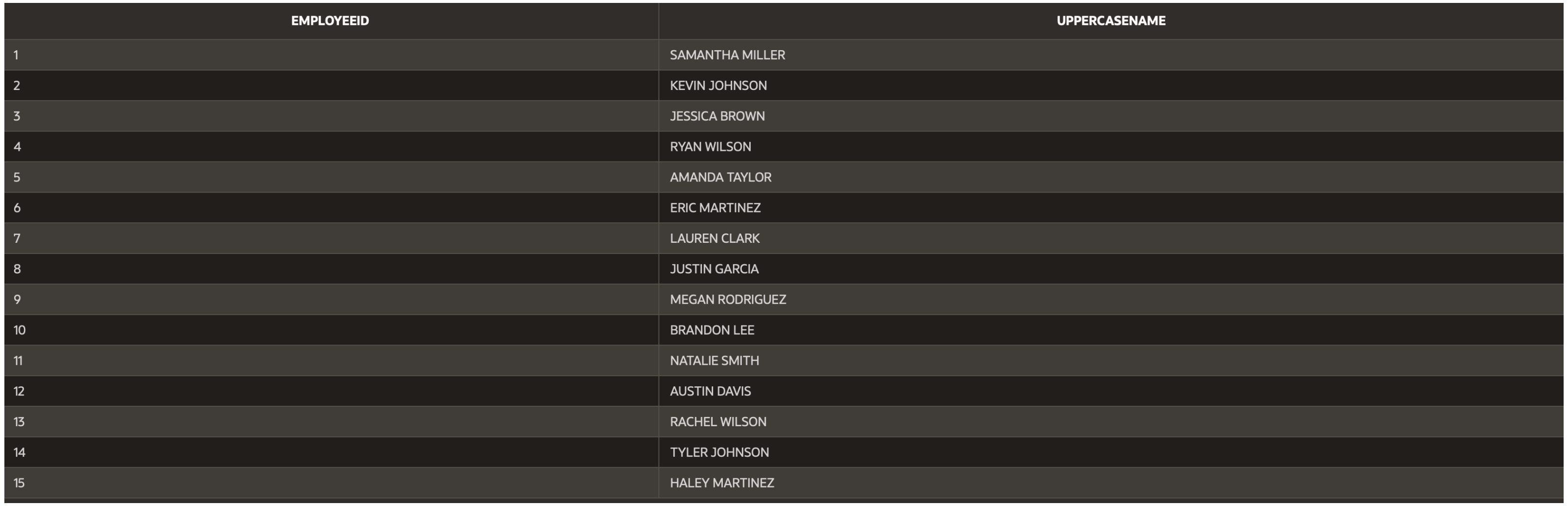
```
SELECT SalaryID, PaymentDate, EXTRACT(MONTH FROM PaymentDate) AS PaymentMonth
FROM Salaries;
```



SALARYID	PAYMENTDATE	PAYMENTMONTH
1	31-May-2023	5
2	31-May-2023	5
3	31-May-2023	5
4	31-May-2023	5
5	31-May-2023	5
6	31-May-2023	5
7	31-May-2023	5
8	31-May-2023	5
9	31-May-2023	5
10	31-May-2023	5
11	31-May-2023	5
12	31-May-2023	5
13	31-May-2023	5
14	31-May-2023	5
15	31-May-2023	5

Character Funtion Statement:

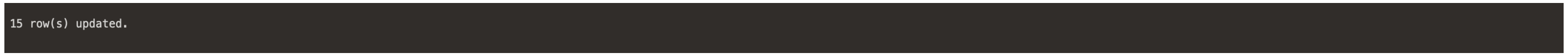
```
SELECT EmployeeID, UPPER(Name) AS UpperCaseName
FROM Employees;
```



EMPLOYEEID	UPPERCASENAME
1	SAMANTHA MILLER
2	KEVIN JOHNSON
3	JESSICA BROWN
4	RYAN WILSON
5	AMANDA TAYLOR
6	ERIC MARTINEZ
7	LAUREN CLARK
8	JUSTIN GARCIA
9	MEGAN RODRIGUEZ
10	BRANDON LEE
11	NATALIE SMITH
12	AUSTIN DAVIS
13	RACHEL WILSON
14	TYLER JOHNSON
15	HALEY MARTINEZ

Update Function Statement:

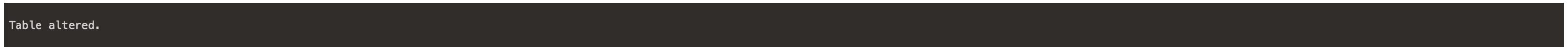
```
UPDATE Salaries
SET GrossPay = GrossPay * 1.1;
```



```
15 row(s) updated.
```

Alter Table Statement:

```
ALTER TABLE Salaries
ADD Bonus NUMBER;
```



```
Table altered.
```