PROJECT 2

<u>NORMALIZATION PROJECT</u>

New Employee Personal and Banking Details Form (Bottom up Approach)

BIG DATA FOR MANAGERS & ANALYTICS- 1

Submitted to: Prof. Ashok Harnal

Submitted by: Group 6

Hritika Gupta, 045020

Khushi, 045027

Nandini Arya, 045034

BDA Batch- 04

RAMSAY
HEALTH CARE

NEW EMPLOYEE
PERSONAL AND BANKING DETAILS FORM

Given Names: _____ Surname: _____

Address: _____ Postcode: _____

Phone: h) _____ m) _____ Marital Status: _____

E-mail address: _____ Date of Birth: ___/___/___

JHC will use this email address to send you your payslips fortnightly and to communicate relevant hospital information to you.

I have received, read, understand and accept in full the terms and conditions of my contract of employment dated ___/___/___ Signature: _____

BANK ACCOUNT DETAILS: *(for direct payment of salary)*

Bank Name: _____

Branch Address: _____

BSB No: (branch No.)   _____

Account Number: _____

Name/s on Account: _____

MULTIPLE ACCOUNT DETAILS:

I authorise the Payroll Department to deduct $_____ each payday and forward to:

Bank Name: _____

Branch Address: _____

BSB No: (branch No.)   _____

Account Number: _____

Name/s on Account: _____

EMERGENCY CONTACTS

1. Name _____ Relationship _____

   Phone (Work) _____ (Mobile) _____

2. Name _____ Relationship _____

   Phone (Work) _____ (Mobile) _____

This form is a "New Employee Personal and Banking Details Form" used by Ramsay Health Care, likely for new hires to provide essential personal and financial information. This form contains these elements-

| Given Names | Surname | Address | Postcode | Phone (Home, Mobile) | Marital Status | Email Address | Date of Birth | Date of Employment | Signature | Bank Name | Branch Address | BSB No | Account Number | Name on Account | Deduction Amount | Emergency Contact Name | Relationship | Emergency Contact Phone (Work, Mobile) |

This form is used to collect all necessary personal, banking, and emergency contact details for the onboarding process of new employees. It ensures that the employer has the correct information to process payroll, communicate effectively, and reach out in case of an emergency.

1. Personal Information:

- **Given Names & Surname:** The employee's full name.
- **Address:** The employee's home address, including postcode.
- **Phone:** Two contact numbers are requested (home and mobile).
- **Email Address:** To send payslips and communicate hospital-related information.
- **Marital Status:** The employee's marital status (e.g., single, married).
- **Date of Birth:** The employee's birthdate.

2. Acknowledgement:

- The employee acknowledges having read, understood, and accepted the terms and conditions of their contract by signing and dating this section.

3. Bank Account Details (for direct payment of salary):

- **Bank Name & Branch Address:** The name of the bank and its branch where the employee's account is held.
- **BSB No.:** Bank-State-Branch number, a 6-digit number used for routing payments within Australia.
- **Account Number & Name(s) on Account:** The specific account number and name associated with it.

4. Multiple Account Details:

- If the employee wishes to split their salary into multiple accounts, they can authorize the payroll department to deduct a specified amount each payday to another account.

5. Emergency Contacts:

- **Name, Relationship, Phone (Work/Mobile):** The employee must provide contact information for two emergency contacts, including their relationship to the employee and their work and mobile numbers.

1NF

1NF is the first step in the normalization process of a relational database. A table is considered to be in 1NF if it meets the following criteria:

1. Unique Primary Key: Each table must have a unique identifier (primary key) for each record.
2. Atomic Values: Each field must contain atomic (indivisible) values. This means that each column should hold only a single value, not a set or list of values.
3. No Repeating Groups: There should be no repeating groups or arrays within a table. Each column must represent a single attribute of the entity.

To achieve 1NF, we need to ensure that each table has a unique primary key and that each field contains only atomic (indivisible) values, with no repeating groups or arrays.

First Normal Form is essential for creating a well-structured database that minimizes redundancy and maintains the integrity of the data by ensuring that all fields contain atomic values and that there are no repeating groups.

Issues Identified:

- Multiple phone numbers for each employee.
- Multiple bank accounts for each employee.

Employee Table:

EmployeeID	GivenNames	Surname	Address	Postcode
101	Hritika	Gupta	Gajraula	244235
102	Nandini	Arya	Hapur	245101

MaritalStatus	EmailAddress	DateOfBirth	DateOfEmployment
Single	hritikagupta@gmail.com	22-01-1995	01-06-2024
Single	ng@gmail.com	25-07-1995	01-07-2024

Signature	EmergencyContactName	Relationship	EmergencyPhoneWork	EmergencyPhoneMobile
E. Gupta	Nina Gupta	Mother	5558765	1023456789
N.Arya	Khushi	sister	1235123	1235647119

Employee Phone Numbers Table:

PhoneID	EmployeeID	PhoneType	PhoneNumber
101	101	Home	5551234
102	102	Mobile	5555678

Employee Bank Account Details Table:

AccountNumber	EmployeeID	NameOnAccount	BankName	BranchAddress	BSBNo
123456789	101	Hritika	BankA	Malviya Nagar	123456
121345789	101	Hritika	BankB	Hauz khas	124678
121345785	102	Nandini	BankA	Malviya Nagar	123456

2NF

A table is considered to be in 2NF if it meets the following criteria:

1. The table is in First Normal Form (1NF).
2. All non-key attributes are fully dependent on the primary key.

In other words, a table is in 2NF if every non-key attribute depends on the entire primary key, not just a part of it. This eliminates partial dependencies.

To achieve 2NF, we need to ensure that all non-key attributes are fully dependent on the primary key. If a non-key attribute depends on only a part of the primary key, it should be moved to a separate table.

Issues Identified and Fixed:

Emergency contact details are dependent on EmployeeID, but they are not directly related to other attributes.

Bank details are repeated for each account.

Employee Table:

EmployeeID	GivenNames	Surname	Address	Postcode
101	Hritika	Gupta	Gajraula	244235
102	Nandini	Arya	Hapur	245101

MaritalStatus	EmailAddress	DateOfBirth	DateOfEmployment	Signature
Single	hritikagupta@gmail.com	22-01-1995	01-06-2024	E. Gupta
Single	ng@gmail.com	25-07-1995	01-07-2024	N.Arya

Employee Phone Numbers Table:

PhoneID	EmployeeID	PhoneType	PhoneNumber
101	101	Home	5551234
102	102	Mobile	5555678

Employee Account Details Table:

AccountNumber	EmployeeID	NameOnAccount	BankName
123456789	101	Hritika	BankA
121345789	101	Hritika	BankB
121345785	102	Nandini	BankA

Employee Bank Details Table:

BankName	BranchAddress	BSBNo
BankA	Malviya Nagar	123456
BankB	Hauz khas	124678
BankA	Malviya Nagar	123456

Emergency Contact Details:

EmergencyId	EmployeeID	Contact Name	Relationship	Phone(Work)	PhoneMobile
1	101	Nina Gupta	Mother	5558765	1023456789
2	102	Khushi	sister	1235123	1235647119

3NF

A table is considered to be in 3NF if it meets the following criteria:

1. The table is in Second Normal Form (2NF).
2. There are no transitive dependencies. This means that non-key attributes must not depend on other non-key attributes.

In simpler terms, every non-key attribute should depend only on the primary key and not on any other non-key attribute.

To achieve 3NF, we need to identify and eliminate any transitive dependencies. If a non-key attribute depends on another non-key attribute, we should move that attribute to a separate table.

- **Employee Table: No transitive dependencies. All attributes are directly dependent on EmployeeID.**
- **Phone Table: No transitive dependencies. PhoneType and PhoneNumber are dependent on PhoneID.**
- **Bank Table: No transitive dependencies. BankName, BranchAddress, and BSBNo are dependent on BankID.**
- **Account Table: No transitive dependencies. BankID, AccountNumber, and NameOnAccount are dependent on AccountID.**
- **EmergencyContact Table: No transitive dependencies. ContactName, Relationship, PhoneWork, and PhoneMobile are dependent on ContactID.**

In conclusion, the process of normalizing the database through 1NF, 2NF, and 3NF ensures a robust and efficient data structure that minimizes redundancy and maintains data integrity. This comprehensive normalization process not only optimized data storage and retrieval but also significantly reduced the risk of anomalies, resulting in a well-structured, scalable, and maintainable database system.

Creating Database:

```sql
CREATE DATABASE EMPLOYEE1;
USE EMPLOYEE1;
```

Creating Tables:

```sql
CREATE TABLE Employee (
    EmployeeID CHAR(3) PRIMARY KEY,
    GivenNames VARCHAR(50) NOT NULL,
    Surname VARCHAR(50) NOT NULL,
    Address VARCHAR(100),
    Postcode CHAR(6),
    MaritalStatus ENUM('MARRIED', 'SINGLE', 'DIVORCE') NOT NULL,
    EmailAddress VARCHAR(100),
    DateOfBirth DATE NOT NULL,
    DateOfEmployment DATE NOT NULL,
    Signature VARCHAR(50) NOT NULL
);

CREATE TABLE Phone (
    PhoneID CHAR(3) PRIMARY KEY,
    EmployeeID CHAR(3),
    PhoneType ENUM('home', 'work') NOT NULL,
    PhoneNumber VARCHAR(10) NOT NULL,
    FOREIGN KEY (EmployeeID) REFERENCES Employee(EmployeeID),
    CHECK (
        (PhoneType = 'work' AND LENGTH(PhoneNumber) = 7) OR
        (PhoneType = 'home' AND LENGTH(PhoneNumber) = 10)
    )
);

CREATE TABLE Bank (
    BankName VARCHAR(50),
    BranchAddress VARCHAR(100) NOT NULL UNIQUE,
    BSBNo CHAR(6) NOT NULL UNIQUE,
    PRIMARY KEY (BankName)
);

CREATE TABLE BankAccount (
    AccountNumber VARCHAR(9) PRIMARY KEY,
    EmployeeID CHAR(3) NOT NULL,
    NameOnAccount VARCHAR(50),
    BankName VARCHAR(50) NOT NULL,
    FOREIGN KEY (EmployeeID) REFERENCES Employee(EmployeeID),
    FOREIGN KEY (BankName) REFERENCES Bank(BankName)
```

```sql
);

CREATE TABLE EmergencyContact (
    EmergencyId CHAR(1) PRIMARY KEY,
    EmployeeID CHAR(3) NOT NULL,
    ContactName VARCHAR(50) NOT NULL,
    Relationship ENUM('MOTHER', 'FATHER', 'WIFE', 'HUSBAND', 'SISTER',
'BROTHER', 'COUSIN'),
    PhoneWork CHAR(7),
    PhoneMobile CHAR(10),
    FOREIGN KEY (EmployeeID) REFERENCES Employee(EmployeeID)
);
```

INSERT STATEMENTS:

    1. **Employee**

```sql
INSERT INTO Employee
VALUES ('101', 'John', 'Doe', '123 Main St, Sydney', '2000', 'MARRIED',
'johndoe@example.com', '1985-05-15', '2010-06-01', 'JohnDoeSignature');

INSERT INTO Employee
VALUES ('102', 'Jane', 'Smith', '456 Market St, Melbourne', '3000', 'SINGLE',
'janesmith@example.com', '1990-08-25', '2015-03-12', 'JaneSmithSignature');
```

    2. **Phone**

```sql
INSERT INTO Phone
VALUES ('P01', '101', 'home', '0412345678');

INSERT INTO Phone
VALUES ('P02', '102', 'work', '7890123');
```

    3. **Bank**

```sql
INSERT INTO Bank
VALUES ('Commonwealth Bank', '123 Pitt St, Sydney', '123456');

INSERT INTO Bank
VALUES ('ANZ Bank', '789 Collins St, Melbourne', '654321');
```

    4. **Bank account**

```sql
INSERT INTO BankAccount
VALUES ('123456789', '101', 'John Doe', 'Commonwealth Bank');

INSERT INTO BankAccount
VALUES ('987654321', '102', 'Jane Smith', 'ANZ Bank');
```
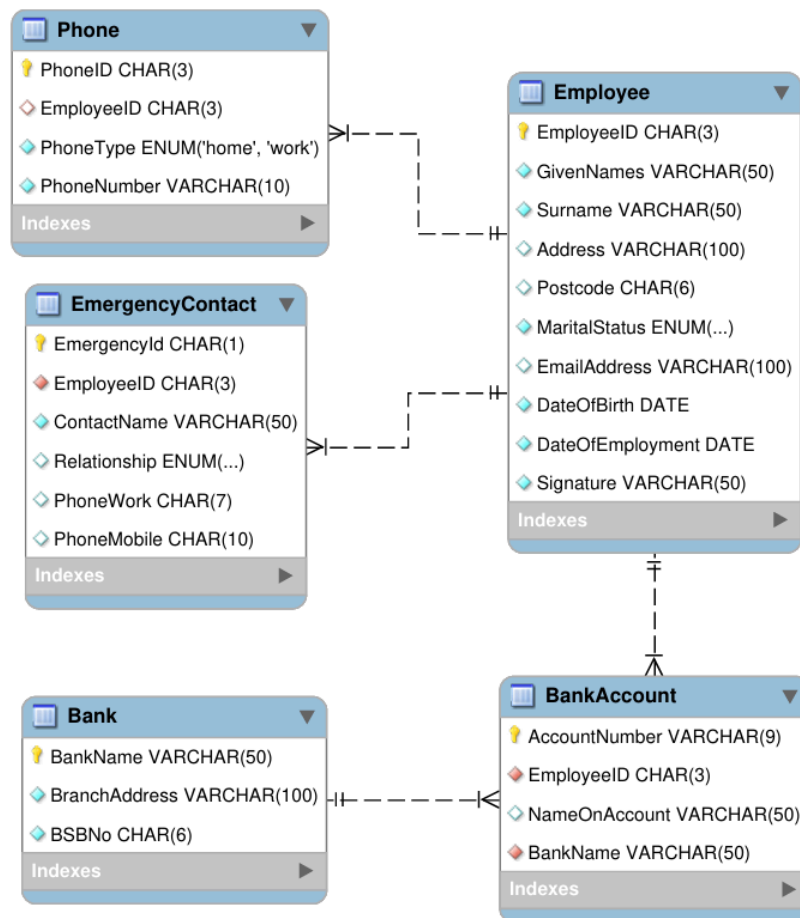
5. EmergencyContact

INSERT INTO EmergencyContact
VALUES ('1', '101', 'Mary Doe', 'WIFE', '2345678', '0411223344');

INSERT INTO EmergencyContact
VALUES ('2', '102', 'Michael Smith', 'BROTHER', '3456789', '0422334455');

ERD DIAGRAM:

GRANT ACCESS:

Human Resources (HR) : Needs access to manage employee records, phone details, and emergency contact information.
Privileges: SELECT, INSERT, UPDATE on employee, phone_details, emergency_contact_no.

Payroll Department: Needs access to view employee information, phone details, bank details, and bank account details for payroll processing.
Privileges: SELECT on employee, phone_details, bank_details, bank_account_details.

Compliance Officer: Needs read-only access to all tables for auditing and compliance checks.
Privileges: SELECT on all tables.
Code:

```sql
-- Grant privileges to Human Resources
GRANT SELECT, INSERT, UPDATE ON EMPLOYEE1.employee TO
'hr_user'@'localhost';
GRANT SELECT, INSERT, UPDATE ON EMPLOYEE1.phone_details TO
'hr_user'@'localhost';
GRANT SELECT, INSERT, UPDATE ON EMPLOYEE1.emergency_contact_no TO
'hr_user'@'localhost';

-- Grant privileges to Payroll Department
GRANT SELECT ON EMPLOYEE1.employee TO 'payroll_user'@'localhost';
GRANT SELECT ON EMPLOYEE1.phone_details TO 'payroll_user'@'localhost';
GRANT SELECT ON EMPLOYEE1.bank_details TO 'payroll_user'@'localhost';
GRANT SELECT ON EMPLOYEE1.bank_account_details TO 'payroll_user'@'localhost';

-- Grant privileges to Compliance Officer
GRANT SELECT ON EMPLOYEE1.employee TO 'compliance_user'@'localhost';
GRANT SELECT ON EMPLOYEE1.phone_details TO 'compliance_user'@'localhost';
GRANT SELECT ON EMPLOYEE1.bank_details TO 'compliance_user'@'localhost';
GRANT SELECT ON EMPLOYEE1.bank_account_details TO
'compliance_user'@'localhost';
GRANT SELECT ON EMPLOYEE1.emergency_contact_no TO
'compliance_user'@'localhost';
```