Interest Calculator

Banks.java

```
package com.InterestCalculator;
//Base class containing all the basic details of the bank
public class Banks {
    //Declaration of attributes for identifying the Bank
    public String BankName;
    public String BankId;
    public String IFSCCode;
    public String MICRCode;
    public String BankLocation;
    public String phoneNumber;

    //Constructor for initializing the attributes
    public Banks(String bankName, String bankId, String IFSCCode, String
MICRCode, String bankLocation, String phoneNumber) {
        this.BankName = bankName;
        this.BankId = bankId;
        this.IFSCCode = IFSCCode;
        this.MICRCode = MICRCode;
        this.BankLocation = bankLocation;
        this.phoneNumber = phoneNumber;
    }

    //Display the bank details
    public void getBankDetails(){
        System.out.println(".....Bank Details Fetched.....");
        System.out.println("Bank Name:"+this.BankName);
        System.out.println("Bank Id:"+this.BankId);
```

```
        System.out.println("IFSC Code:"+this.IFSCCode);
        System.out.println("MICR Code:"+this.MICRCode);
        System.out.println("Bank Location:"+this.BankLocation);
        System.out.println("Phone Number:"+this.phoneNumber);
    }
}
```

Interests.java

```
package com.InterestCalculator;

public class Interests extends Banks{
    //Inheriting from the parent class Bank

    //Calling the parent method object
    public Interests(String bankName, String bankId, String IFSCCode, String
MICRCode, String bankLocation, String phoneNumber) {
        super(bankName, bankId, IFSCCode, MICRCode, bankLocation,
phoneNumber);
    }

    //Declaring the rate of interests and the time periods as private
modifiers
    private double PersonalLoan;
    private int PersonalLoanTimePeriod;
    private double HousingLoan;
    private int HousingLoanTimePeriod;
    private double educationalLoan;
    private int educationalLoanTimePeriod;
    private double goldLoan;
    private int goldLoanTimePeriod;
```

```
    private double goldRate;

    //setting the rate of interests and time period
    public void setInterests(double personalLoan, double housingLoan,
double EducationalLoan, double GoldLoan , int personalLoanTimePeriod,
int housingLoanTimePeriod, int EducationalLoanTimePeriod, int
GoldLoanTimePeriod, double GoldRate){
        PersonalLoan = personalLoan;
        HousingLoan = housingLoan;
        educationalLoan = EducationalLoan;
        goldLoan = GoldLoan;
        PersonalLoanTimePeriod = personalLoanTimePeriod;
        HousingLoanTimePeriod = housingLoanTimePeriod;
        educationalLoanTimePeriod = EducationalLoanTimePeriod;
        goldLoanTimePeriod = GoldLoanTimePeriod;
        goldRate = GoldRate;
    }

    //getter and setter for the private attributes


    public double getPersonalLoan() {
        return PersonalLoan;
    }

    public void setPersonalLoan(double personalLoan) {
        PersonalLoan = personalLoan;
    }

    public int getPersonalLoanTimePeriod() {
```

```
    return PersonalLoanTimePeriod;
}

public void setPersonalLoanTimePeriod(int personalLoanTimePeriod) {
    PersonalLoanTimePeriod = personalLoanTimePeriod;
}

public double getHousingLoan() {
    return HousingLoan;
}

public void setHousingLoan(double housingLoan) {
    HousingLoan = housingLoan;
}

public int getHousingLoanTimePeriod() {
    return HousingLoanTimePeriod;
}

public void setHousingLoanTimePeriod(int housingLoanTimePeriod) {
    HousingLoanTimePeriod = housingLoanTimePeriod;
}

public double getEducationalLoan() {
    return educationalLoan;
}

public void setEducationalLoan(double educationalLoan) {
    this.educationalLoan = educationalLoan;
}
```

```
    public int getEducationalLoanTimePeriod() {
        return educationalLoanTimePeriod;
    }

    public void setEducationalLoanTimePeriod(int
educationalLoanTimePeriod) {
        this.educationalLoanTimePeriod = educationalLoanTimePeriod;
    }

    public double getGoldLoan() {
        return goldLoan;
    }

    public void setGoldLoan(double goldLoan) {
        this.goldLoan = goldLoan;
    }

    public int getGoldLoanTimePeriod() {
        return goldLoanTimePeriod;
    }

    public void setGoldLoanTimePeriod(int goldLoanTimePeriod) {
        this.goldLoanTimePeriod = goldLoanTimePeriod;
    }

    public double getGoldRate() {
        return goldRate;
    }
```

```
    public void setGoldRate(double goldRate) {
        this.goldRate = goldRate;
    }
}
```

InterestCalculator.java

```
package com.InterestCalculator;

public class InterestCalculator {
    //Formula to find simpleInterest SI = (PNR)/100
    //calculate interest for gold loans
    public double calculateInterest(double noOfGramsOfGold, double
goldRate, double rate, int timePeriod){
        /*System.out.println("gold in grams"+noOfGramsOfGold);
        System.out.println("Gold Rate:"+goldRate);
        System.out.println("Interest:"+rate);
        System.out.println("Time Period:"+timePeriod);*/
        return (noOfGramsOfGold*goldRate) * timePeriod*rate/100;
    }
    //calculate interest for other loans except gold loans
    public double calculateInterest(double principal, double rate, int
timePeriod){
        /*System.out.println("Principal:"+principal);
        System.out.println("Interest:"+rate);
        System.out.println("Time Period:"+timePeriod);*/
        return (principal*timePeriod*rate)/100;

    }
}
```

Main.java

```
package com.InterestCalculator;

import java.util.Scanner;
public class Main{
    public static void main(String[] args){
        //Creating Bank Object- SBI
         Interests SBI = new Interests("SBI",
"85SAD","SBIB0085623","964563215","Attibele","04244875362");
        //Setting rate of interests and time period- SBI
         SBI.setInterests(4.0,5.6,3.0,3.45, 2, 15,3,1, 2500.56);

        //Creating Bank Object- INDIAN BANK
         Interests INDIAN_BANK = new Interests("Indian Bank",
"85FB8","IDIB656AS623","862345956","JP Nagar","04244237585");
        //Setting rate of interests and time period- INDIAN BANK
         INDIAN_BANK.setInterests(4.2,5.0,2.7,3.8, 2, 16,2,1, 2756.85);

        //Creating Bank Object- AXIS BANK
         Interests AXIS = new Interests("Axis",
"8A511D","AXIB0SD56623","8454613225","Bommasandara","04244726453");
        //Setting rate of interests and time period- AXIS BANK
         AXIS.setInterests(2.0,4.0,1.7,2.95, 1, 17,2,1, 2223.46);

        //Creating Bank Object- HDFC BANK
         Interests HDFC = new Interests("HDFC",
"7WE86","HDF986S232","8662105753","BTM Layout","04244153642");
        //Setting rate of interests and time period- HDFC BANK
         HDFC.setInterests(4.0,5.8,3.2,3.7, 4, 13,4,1, 2900.56);
```

```
        //Creating Bank Object- ICICI BANK
         Interests ICICI = new Interests("ICICI",
"178AS","ICI6563S589","753056352","Electronic City","04244756325");
        //Setting rate of interests and time period- ICICI BANK
         ICICI.setInterests(4.7,4.23,7.1,4.5, 3, 17,4,1, 2575.44);

        //Creating objects for classes present in the package
         Scanner sc = new Scanner(System.in);
         DisplayInterest di = new DisplayInterest();
         InterestCalculator ic = new InterestCalculator();
         UserDetails ud = new UserDetails();
         ud.getUserDetails(sc);//get user details from the user

        //Select Bank from the available options
         String youChose;
         do {
            System.out.println("Select a
bank:------------------------------------------------");
            System.out.print("Enter 1 for State Bank of India\nEnter 2 for Indian
Bank\nEnter 3 for Axis Bank\nEnter 4 for HDFC Bank\nEnter 5 for ICICI
Bank\nYour choice?");
            int bankChoice = sc.nextInt();
            //Select the type of loan
            System.out.print("Enter 1 for gold loan\nEnter 2 for Personal
Loan\nEnter 3 for Housing Loan\nEnter 4 for Educational Loan\nYour
Choice?");
            int typeOfLoan = sc.nextInt();
            double PrincipalAmount = 0.0;
            double noOfGramsOfGold = 0.0;
```

```
            //Get the weight of gold in grams if it is gold loan option(1) or get
the principal option(2,3,4)
            if (typeOfLoan == 1) {
                System.out.print("Enter Gold weight (in grams):");
                noOfGramsOfGold = sc.nextDouble();
            } else {
                System.out.print("Enter the Principal Amount (in rupees):");
                PrincipalAmount = sc.nextDouble();
            }

System.out.println("----------------------------------------------------------------
--------------");
            //display user details
            ud.displayUserDetails();
            double SimpleInterest;
            switch (bankChoice) {//controlling the flow using switch
                //SBI Bank
                case 1 -> {
                    if (typeOfLoan == 1) {//Gold Loan
                        //findTotalInterest
                        SimpleInterest = ic.calculateInterest(noOfGramsOfGold,
SBI.getGoldRate(), SBI.getGoldLoan(), SBI.getGoldLoanTimePeriod());
                        SBI.getBankDetails();
                        di.displayAmount("Gold Loan", noOfGramsOfGold,
SBI.getGoldRate(), SBI.getGoldLoan(), SBI.getGoldLoanTimePeriod(),
SimpleInterest);
                    } else if (typeOfLoan == 2) {//Personal Loan
                        //findTotalInterest
                        SimpleInterest = ic.calculateInterest(PrincipalAmount,
SBI.getPersonalLoan(), SBI.getPersonalLoanTimePeriod());
```

```
                    SBI.getBankDetails();
                    di.displayAmount("Personal Loan", PrincipalAmount,
SBI.getPersonalLoan(), SBI.getPersonalLoanTimePeriod(), SimpleInterest);
                } else if (typeOfLoan == 3) {//Housing Loan
                    //findTotalInterest
                    SimpleInterest = ic.calculateInterest(PrincipalAmount,
SBI.getHousingLoan(), SBI.getHousingLoanTimePeriod());
                    SBI.getBankDetails();
                    di.displayAmount("Housing Loan", PrincipalAmount,
SBI.getHousingLoan(), SBI.getHousingLoanTimePeriod(), SimpleInterest);
                } else if (typeOfLoan == 4) {//Educational Loan
                    //findTotalInterest
                    SimpleInterest = ic.calculateInterest(PrincipalAmount,
SBI.getEducationalLoan(), SBI.getEducationalLoanTimePeriod());
                    SBI.getBankDetails();
                    di.displayAmount("Educational Loan", PrincipalAmount,
SBI.getEducationalLoan(), SBI.getEducationalLoanTimePeriod(),
SimpleInterest);
                }
            }

            //INDIAN BANK
            case 2 -> {
                if (typeOfLoan == 1) {//Gold Loan
                    //findTotalInterest
                    SimpleInterest = ic.calculateInterest(noOfGramsOfGold,
INDIAN_BANK.getGoldRate(), INDIAN_BANK.getGoldLoan(),
INDIAN_BANK.getGoldLoanTimePeriod());
                    INDIAN_BANK.getBankDetails();
```

```
                di.displayAmount("Gold Loan", noOfGramsOfGold,
INDIAN_BANK.getGoldRate(), INDIAN_BANK.getGoldLoan(),
INDIAN_BANK.getGoldLoanTimePeriod(), SimpleInterest);
            } else if (typeOfLoan == 2) {//Personal Loan
                //findTotalInterest
                SimpleInterest = ic.calculateInterest(PrincipalAmount,
INDIAN_BANK.getPersonalLoan(),
INDIAN_BANK.getPersonalLoanTimePeriod());
                INDIAN_BANK.getBankDetails();
                di.displayAmount("Personal Loan", PrincipalAmount,
INDIAN_BANK.getPersonalLoan(),
INDIAN_BANK.getPersonalLoanTimePeriod(), SimpleInterest);
            } else if (typeOfLoan == 3) {//Housing Loan
                //findTotalInterest
                SimpleInterest = ic.calculateInterest(PrincipalAmount,
INDIAN_BANK.getHousingLoan(),
INDIAN_BANK.getHousingLoanTimePeriod());
                INDIAN_BANK.getBankDetails();
                di.displayAmount("Housing Loan", PrincipalAmount,
INDIAN_BANK.getHousingLoan(),
INDIAN_BANK.getHousingLoanTimePeriod(), SimpleInterest);
            } else if (typeOfLoan == 4) {//Educational Loan
                //findTotalInterest
                SimpleInterest = ic.calculateInterest(PrincipalAmount,
INDIAN_BANK.getEducationalLoan(),
INDIAN_BANK.getEducationalLoanTimePeriod());
                INDIAN_BANK.getBankDetails();
                di.displayAmount("Educational Loan", PrincipalAmount,
INDIAN_BANK.getEducationalLoan(),
INDIAN_BANK.getEducationalLoanTimePeriod(), SimpleInterest);
```

```
                }
            }
            //AXIS BANK
            case 3 -> {
                if (typeOfLoan == 1) {//Gold Loan
                    //findTotalInterest
                    SimpleInterest = ic.calculateInterest(noOfGramsOfGold,
AXIS.getGoldRate(), AXIS.getGoldLoan(), AXIS.getGoldLoanTimePeriod());
                    AXIS.getBankDetails();
                    di.displayAmount("Gold Loan", noOfGramsOfGold,
AXIS.getGoldRate(), AXIS.getGoldLoan(), AXIS.getGoldLoanTimePeriod(),
SimpleInterest);
                } else if (typeOfLoan == 2) {//Personal Loan
                    //findTotalInterest
                    SimpleInterest = ic.calculateInterest(PrincipalAmount,
AXIS.getPersonalLoan(), AXIS.getPersonalLoanTimePeriod());
                    AXIS.getBankDetails();
                    di.displayAmount("Personal Loan", PrincipalAmount,
AXIS.getPersonalLoan(), AXIS.getPersonalLoanTimePeriod(),
SimpleInterest);
                } else if (typeOfLoan == 3) {//Housing Loan
                    //findTotalInterest
                    SimpleInterest = ic.calculateInterest(PrincipalAmount,
AXIS.getHousingLoan(), AXIS.getHousingLoanTimePeriod());
                    AXIS.getBankDetails();
                    di.displayAmount("Housing Loan", PrincipalAmount,
AXIS.getHousingLoan(), AXIS.getHousingLoanTimePeriod(), SimpleInterest);
                } else if (typeOfLoan == 4) {//Educational Loan
                    //findTotalInterest
```

```
                    SimpleInterest = ic.calculateInterest(PrincipalAmount,
AXIS.getEducationalLoan(), AXIS.getEducationalLoanTimePeriod());
                    AXIS.getBankDetails();
                    di.displayAmount("Educational Loan", PrincipalAmount,
AXIS.getEducationalLoan(), AXIS.getEducationalLoanTimePeriod(),
SimpleInterest);
                }
            }
            //HDFC
            case 4 -> {
                if (typeOfLoan == 1) {//Gold Loan
                    //findTotalInterest
                    SimpleInterest = ic.calculateInterest(noOfGramsOfGold,
HDFC.getGoldRate(), HDFC.getGoldLoan(),
HDFC.getGoldLoanTimePeriod());
                    HDFC.getBankDetails();
                    di.displayAmount("Gold Loan", noOfGramsOfGold,
HDFC.getGoldRate(), HDFC.getGoldLoan(),
HDFC.getGoldLoanTimePeriod(), SimpleInterest);
                } else if (typeOfLoan == 2) {//Personal Loan
                    //findTotalInterest
                    SimpleInterest = ic.calculateInterest(PrincipalAmount,
HDFC.getPersonalLoan(), HDFC.getPersonalLoanTimePeriod());
                    HDFC.getBankDetails();
                    di.displayAmount("Personal Loan", PrincipalAmount,
HDFC.getPersonalLoan(), HDFC.getPersonalLoanTimePeriod(),
SimpleInterest);
                } else if (typeOfLoan == 3) {//Housing Loan
                    //findTotalInterest
```

```
                    SimpleInterest = ic.calculateInterest(PrincipalAmount,
HDFC.getHousingLoan(), HDFC.getHousingLoanTimePeriod());
                    HDFC.getBankDetails();
                    di.displayAmount("Housing Loan", PrincipalAmount,
HDFC.getHousingLoan(), HDFC.getHousingLoanTimePeriod(),
SimpleInterest);
                } else if (typeOfLoan == 4) {//Educational Loan
                    //findTotalInterest
                    SimpleInterest = ic.calculateInterest(PrincipalAmount,
HDFC.getEducationalLoan(), HDFC.getEducationalLoanTimePeriod());
                    HDFC.getBankDetails();
                    di.displayAmount("Educational Loan", PrincipalAmount,
HDFC.getEducationalLoan(), HDFC.getEducationalLoanTimePeriod(),
SimpleInterest);
                }
            }
            //ICICI
            case 5 -> {
                if (typeOfLoan == 1) {//Gold Loan
                    //findTotalInterest
                    SimpleInterest = ic.calculateInterest(noOfGramsOfGold,
ICICI.getGoldRate(), ICICI.getGoldLoan(), ICICI.getGoldLoanTimePeriod());
                    ICICI.getBankDetails();
                    di.displayAmount("Gold Loan", noOfGramsOfGold,
ICICI.getGoldRate(), ICICI.getGoldLoan(), ICICI.getGoldLoanTimePeriod(),
SimpleInterest);
                } else if (typeOfLoan == 2) {//Personal Loan
                    //findTotalInterest
                    SimpleInterest = ic.calculateInterest(PrincipalAmount,
ICICI.getPersonalLoan(), ICICI.getPersonalLoanTimePeriod());
```

```
                    ICICI.getBankDetails();
                    di.displayAmount("Personal Loan", PrincipalAmount,
ICICI.getPersonalLoan(), ICICI.getPersonalLoanTimePeriod(),
SimpleInterest);
                } else if (typeOfLoan == 3) {//Housing Loan
                    //findTotalInterest
                    SimpleInterest = ic.calculateInterest(PrincipalAmount,
ICICI.getHousingLoan(), ICICI.getHousingLoanTimePeriod());
                    ICICI.getBankDetails();
                    di.displayAmount("Housing Loan", PrincipalAmount,
ICICI.getHousingLoan(), ICICI.getHousingLoanTimePeriod(), SimpleInterest);
                } else if (typeOfLoan == 4) {//Educational Loan
                    //findTotalInterest
                    SimpleInterest = ic.calculateInterest(PrincipalAmount,
ICICI.getEducationalLoan(), ICICI.getEducationalLoanTimePeriod());
                    ICICI.getBankDetails();
                    di.displayAmount("Educational Loan", PrincipalAmount,
ICICI.getEducationalLoan(), ICICI.getEducationalLoanTimePeriod(),
SimpleInterest);
                }
            }
            //Wrong option
            default -> System.out.println("Invalid Choice. Please Try Again");
        }
        //ask the user if he/she needs to calculate more interests
        System.out.print("Wish to view results of other banks(yes/no)?");
        youChose = sc.next();
    }while(youChose.equals("yes"));
    System.out.println("Thank you:)");
  }
```

```
}
```

UserDetails.java

```
package com.InterestCalculator;

import java.util.Scanner;

public class UserDetails {
    public String userName;
    public long accountNumber;

    //Get user details
    public void getUserDetails(Scanner sc){
        System.out.print("Enter your Name: ");
        userName = sc.nextLine();
        System.out.print("Enter your account Number: ");
        accountNumber = sc.nextLong();
    }
    //Display user details
    public void displayUserDetails(){
        System.out.println(".....User Details:.....");
        System.out.println("Account Holder:"+userName);
        System.out.println("Account Number:"+accountNumber);
    }
}
```

DisplayInterest.java

```
package com.InterestCalculator;
```

```
public class DisplayInterest {
   //Display the loan details with interest for gold loans
   public void displayAmount(String category,double noOfGrams, double
goldRate, double rate, int timePeriod,double simpleInterest){
       System.out.println("............Loan Details.............");
       System.out.println("Category:"+category);
       System.out.println("No of grams:"+noOfGrams);
       System.out.format("Today's cost of gold(Fixed by the
bank):%.2f",goldRate);
       System.out.println("\nInterest Rate:"+rate);
       double principal = noOfGrams*goldRate;
       System.out.format("Principal:%.2f",principal);
       System.out.println("\nTime period:"+(timePeriod*12)+" months");
       System.out.format("Overall Interest:%.2f",simpleInterest);
       double monthlyPay =
(principal/(timePeriod*12))+(simpleInterest/(timePeriod*12));
       System.out.format("\nTo Pay(monthly):%.2f",monthlyPay);

System.out.println("\n-----------------------------------------------------------
-----------------");
   }
   //Display the loan details with interest for other loans except gold loans
   public void displayAmount(String category,double principal, double rate,
int timePeriod,double simpleInterest){
       System.out.println("............Loan Details.............");
       System.out.println("Category:"+category);
       System.out.format("Principal:%.2f",principal);
       System.out.println("\nInterest Rate:"+rate);
       System.out.println("Time period:"+(timePeriod*12)+" months");
```

```
        System.out.println("Interest Rate:"+rate);
        System.out.format("Overall Interest():%.2f",simpleInterest);
        double monthlyPay =
(principal/(timePeriod*12))+(simpleInterest/(timePeriod*12));
        System.out.format("\nTo Pay(monthly):%.2f",monthlyPay);

System.out.println("\n-----------------------------------------------------------
----------------");
    }
}
```

OUTPUT:

```
Enter your Name: MO
Enter your account Number: 6383664520
Select a bank:---------------------------------------------------
Enter 1 for State Bank of India
Enter 2 for Indian Bank
Enter 3 for Axis Bank
Enter 4 for HDFC Bank
Enter 5 for ICICI Bank
Your choice?1
Enter 1 for gold loan
Enter 2 for Personal Loan
Enter 3 for Housing Loan
Enter 4 for Educational Loan
Your Choice?1
Enter Gold weight (in grams):10
--------------------------------------------------------------------------------
```

.....User Details:.....
Account Holder:MO
Account Number:6383664520
.....Bank Details Fetched.....
Bank Name:SBI
Bank Id:85SAD
IFSC Code:SBIB0085623
MICR Code:964563215
Bank Location:Attibele
Phone Number:04244875362
.....Loan Details.....
Category:Gold Loan
No of grams:10.0
Today's cost of gold(Fixed by the bank):2500.56
Interest Rate:3.45
Principal:25005.60
Time period:12 months
Overall Interest:862.69
To Pay(monthly):2155.69

Wish to view results of other banks(yes/no)?yes
Select a bank:----------
Enter 1 for State Bank of India
Enter 2 for Indian Bank
Enter 3 for Axis Bank
Enter 4 for HDFC Bank
Enter 5 for ICICI Bank
Your choice?4
Enter 1 for gold loan
Enter 2 for Personal Loan

```
Enter 3 for Housing Loan
Enter 4 for Educational Loan
Your Choice?3
Enter the Principal Amount (in rupees):5000000
--------------------------------------------------------------------------------
.............User Details:.............
Account Holder:MO
Account Number:6383664520
..........Bank Details Fetched..........
Bank Name:HDFC
Bank Id:7WE86
IFSC Code:HDF986S232
MICR Code:8662105753
Bank Location:BTM Layout
Phone Number:04244153642
.............Loan Details.............
Category:Housing Loan
Principal:5000000.00
Interest Rate:5.8
Time period:156 months
Interest Rate:5.8
Overall Interest():3770000.00
To Pay(monthly):56217.95
--------------------------------------------------------------------------------
Wish to view results of other banks(yes/no)?no
Thank you:)
```

College Database

```
//create a table students with columns id,age,mobile number,registration number,year of batch
>>> create table students(id integer PRIMARY key AUTOINCREMENT,name varchar(50) NOT NULL,age INTEGER NOT NULL, mobile_number VARCHAR(11), registration_number VARCHAR(10) NOT NULL, year_of_batch INTEGER, UNIQUE(registration_number));
//create a table teachers with columns name,domain, department
>>> create table teachers(id integer PRIMARY key AUTOINCREMENT,name varchar(50) NOT NULL, domain varchar(40) not NULL,department varchar(20) not NULL);

//insert 10 rows into the table students
>>> INSERT into
students(name,age,mobile_number,registration_number,year_of_batch)
VALUES ("Peyton List",19,"7853694127","22IT047",2022);
>>> INSERT into
students(name,age,mobile_number,registration_number,year_of_batch)
VALUES ("Harry",18,"08642359186","21CS049",2021);
>>> INSERT into
students(name,age,mobile_number,registration_number,year_of_batch)
VALUES ("Bebe Rexha",17,"6383452127","22EE015",2023);
>>> INSERT into
students(name,age,mobile_number,registration_number,year_of_batch)
VALUES ("Halsey",18,"06425896153","21ME196",2021);
>>> INSERT into
students(name,age,mobile_number,registration_number,year_of_batch)
VALUES ("Shawn",19,"07536241053","22BM075",2022);
```

```
>>> INSERT into
students(name,age,mobile_number,registration_number,year_of_batch)
VALUES ("Bieber",19,"08634529753","21IS045",2021);
>>> INSERT into
students(name,age,mobile_number,registration_number,year_of_batch)
VALUES ("Alex",19,"09562067538","22RA035",2021);
>>> INSERT into
students(name,age,mobile_number,registration_number,year_of_batch)
VALUES ("Khalid",19,"7568234862","21AE123",2021);
>>> INSERT into
students(name,age,mobile_number,registration_number,year_of_batch)
VALUES ("Drake",19,"9997586754","22CI121",2022);
>>> INSERT into
students(name,age,mobile_number,registration_number,year_of_batch)
VALUES ("Weekend",18,"6583275862","21EC111",2023);


//insert 10 rows into the table teachers
>>> INSERT into teachers(name,domain,department) values ("Robert","Web
technology","IT");
>>> INSERT into teachers(name,domain,department) values ("Mary","Digital
Logics","ECE");
>>> INSERT into teachers(name,domain,department) values ("John","Software
Engineering","CSE");
>>> INSERT into teachers(name,domain,department) values
("Shawn","Automation and machine vision","EI");
>>> INSERT into teachers(name,domain,department) values ("Riley","Android
app technology","CSE");
>>> INSERT into teachers(name,domain,department) values ("Susan","Nano
technology","Physics");
```

```
>>> INSERT into teachers(name,domain,department) values
("Mathew","Circuits","ECE");
>>> INSERT into teachers(name,domain,department) values
("Michael","HealthCare Technologies","BME");
>>> INSERT into teachers(name,domain,department) values ("Michaela","Artificial
Intelligence","IT");
>>> INSERT into teachers(name,domain,department) values ("William","Data
Science","IT");

//get all the records of students with year_of_bacth as 2022
>>> select * from students where year_of_batch = 2022
```

id	name	age	mobile_number	registration_number	year_of_batch
1	Peyton List	19	7853694127	22IT047	2022
5	Shawn	19	07536241053	22BM075	2022
9	Drake	19	9997586754	22CI121	2022

```
//get all the records of teachers where department is CSE
>>> select * from teachers where department = "CSE"
```

id	name	domain	department
3	John	Software Engineering	CSE
5	Riley	Android app technology	CSE

```
//updating mobileNumber of student where registration number is 22IT047
>>> UPDATE students set mobile_number = "8563429654" where
registration_number = "22IT047";
//updating age of student where registration number is 22BM075
>>> UPDATE students set age = 20 where registration_number = "22BM075";
//updating year_of_batch of student where registration number is 21IS045
>>> UPDATE students set year_of_batch = 2022 where registration_number =
"21IS045";
```

```
//deleting teacher records whose names are Robert,Mary and William
>>> delete from teachers where name = "Robert";
>>> delete from teachers where name = "Mary";
>>> delete from teachers where name = "William";
//display all students
>>> select * from students;
```

id	name	age	mobile_number	registration_number	year_of_batch
1	Peyton List	19	8563429654	22IT047	2022
2	Harry	18	08642359186	21CS049	2021
3	Bebe Rexha	17	6383452127	22EE015	2023
4	Halsey	18	06425896153	21ME196	2021
5	Shawn	20	07536241053	22BM075	2022
6	Bieber	19	08634529753	21IS045	2022
7	Alex	19	09562067538	22RA035	2021
8	Khalid	19	7568234862	21AE123	2021
9	Drake	19	9997586754	22CI121	2022
10	Weekend	18	6583275862	21EC111	2023

```
//display all teachers
>>> select * from teachers;
```

id	name	domain	department
3	John	Software Engineering	CSE
4	Shawn	Automation and machine vision	EI
5	Riley	Android app technology	CSE
6	Susan	Nano technology	Physics
7	Mathew	Circuits	ECE
8	Michael	HealthCare Technologies	BME
9	Michaela	Artificial Intelligence	IT