```java
package com.kce.fitness.bean;

class Trainer extends Member
{
public int trainer_id;
public String trainer_name;
public String t trainer_mobno;
public int year_of_experience;

Trainer(int trainer_id, String trainer_name,String trainer_mobno, int
year_of_experience)
{
this.trainer_id=trainer_id;
this.trainer_name=trainer_name;
this.trainer_mobno=trainer_mobno;
this.year_of_experience=year_of_experience;
}

public int  gettrainer_id()
{
return trainer_id;
}

public String gettrainer_name()
{
return trainer_name;
}

public String gettrainer_mobno()
{
return trainer_mobno;
}

public int getyear_of_experience()
{
return year_of_experience
}
}

Class Member extends Trainer
{
public int member_id;
public String member_name;
public String member_mobno;
public String member_Email;
public String date_of_join;
public String member_training_time;

Member(int member_id, String member_name, String member_mobno, String member_Email,
String date_of_join, String member_training_time,int tainer_id)
```

```java
{
this.member_id=member_id;
this.member_name=member_name;
this.member_mobno=member_mobno;
this.member_Email=member_Email;
this.date_of_join=date_of_join;
this.member_training_time=member_training_time;
super(trainer_id);
}

public int getmember_id()
{
return member_id;
}

public String getmember_name()
{
return member_name;
}

public String getmember_mobno()
{
return member_mobno;
}

public String getmember_Email()
{
return member_Email;
}

public String getdate_of_join()
{
return date_of_join;
}

public String getmember_training_time()
{
return member_training_time;
}

public int gettrainer_id()
{
return trainer_id;
}

}

package com.kce.trainer.util;
import java.sql.*;
import java.kce.fitness.bean;
```

```java
class FittnessTrainer
{

Trainer trainer=null;


String driverName="com.mysql.jdbc.Driver";
String url="jdbc:mysql://localhost:3306/mysql";
String userName="root";
String password="root";

class.forName(driverName).newInstance();        //linking and loading the driver
connection=DriverManager.getConnection(url,username,password);           //establish
connection

public Trainer createtable()
{

String sql1="create table Trainer(trainer_id integer,trainer_name
varchar(20),,trainer_mobno varchar(10),year_of_experience integer)";

PreparedStatement ps1=null;

try{

ps1=connection.prepareStatement(sql1);                                    //prepare query1
boolean val1=ps1.execute();                                               //call exectute
to perforn operation
System.out.println("table created"+val1);
}
catch(Exception e)
{
e.printStackTrace();
}
finally{
if(ps1!=null)
ps1.close();
if(connection!=null)
connection.close();
}catch(SQLException | EmptyException e)
{
e.printStackTrace();
}
}


public Trainer insertintotable()
{

String sql2="insert into Trainer values(?,?,?,?)";
```

```java
PreparedStatement ps2=null;


try{

ps2=connection.prepareStatement(sql2);
ps2.setInt(1,trainer.gettrainer_id());
ps2.setString(2,trainer.gettrainer_name();
ps2.setString(3,trainer.gettrainer_mobno());
ps2.setInt(4,trainer.getyear_of_experience());
ps2.exectute();

}
catch(Exception e)
{
e.printStackTrace();
}
finally{
if(ps2!=null)
ps2.close();
if(connection!=null)
connection.close();
}catch(SQLException | EmptyException e)
{
e.printStackTrace();
}
}


public Trainer deletecolumn(int trainer_id)
{
String sql3="delete fom Trainer where trainer_id=?";
PreparedStatement ps3=null;

ps3=connection.preparedStatement(sql13);

ps.setInt(1,trainer_id);
ps3.execute();

}


public Trainer retrivetable(int trainer_id)
{
String sql4="select * from Trainer where trainer_id=?";
preparedStatement ps4=null;

ps4=connection.prepareStatement(sql4);
ps4.setInt(1,trainer_id);
ResultSet rs=ps.executeQuery();
```

```java
while(rs.next())
{
int trainer_id=rs.getInt(1);
String trainer_name=rs.getString(2);
String trainer_mobno=rs.getString(3);
int year_of_experience=rs.getInt(4);


System.out.println("trainer_id \t trainer_name \t trainer_mobno \t
year_of_experience);
System.out.println(trainer.gettrainer_id+"\t" +trainer.gettrainer_name+ "\t"
+trainer.gettrainer_mobno+ "\t" +trainer.getyear_of_experience);
}
}

}
```

```java
package com.kce.member.util;
import java.sql.*;
import java.kce.fitness.bean;

class Fitness Menmber
{

Member member=null;

String driverName="com.mysql.jdbc.Driver";
String url="jdbc:mysql://localhost:3306/mysql";
String userName="root";
String password="root";

class.forName(driverName).newInstance();       //linking and loading the driver
connection=DriverManager.getConnection(url,username,password);           //establish
connection

public Member createtable()
{

String sql1="create table Member(member_id integer,member_name
varchar(20),member_mobno varchar(20),member_Email varchar(20),date_of_join
varchar(10),member_training_time varchar(20),trainer_id integer)";
```

```java
PreparedStatement ps1=null;

try{

ps=connection.prepareStatement(sql1);                              //prepare query
boolean val=ps1.execute();                                         //call exectute to
perforn operation
System.out.println("table created"+val1);
}
catch(Exception e)
{
e.printStackTrace();
}
finally{
if(ps1!=null)
ps1.close();
if(connection!=null)
connection.close();
}catch(SQLException | EmptyException e)
{
e.printStackTrace();
}
}


public Member insertintotable()
{

String sql2="insert into Member values(?,?,?,?,?,?,?)";
PreparedStatement ps2=null;

try{

ps2=connection.prepareStatement(sql2);
ps2.setInt(1,member.getmember_id());
ps2.setString(2,member.getmember_name())
ps2.setString(3,member.getmember_mobno());
ps2.setString(4,member.getmember_Email());
ps2.setString(5,member.getdate_of_join());
ps2.setString(6,member.getmember_training_time());
ps2.setInt(7,member.gettrainer_id());
ps2.exectute();

}
catch(Exception e)
{
e.printStackTrace();
}
finally{
if(ps2!=null)
```

```java
ps2.close();
if(connection!=null)
connection.close();
}catch(SQLException | EmptyException e)
{
e.printStackTrace();
}
}


public Member deletecolumn(int member_id)
{
String sql3="delete fom Member where member_id=?";
PreparedStatement ps3=null;

ps3=connection.preparedStatement(sql13);

ps.setInt(1,member_id);
ps3.execute();

}


public Member retrivetable(int member_id)
{
String sql4="select * from Member where member_id=?";
preparedStatement ps4=null;

ps4=connection.prepareStatement(sql4);
ps4.setInt(1,member_id);
ResultSet rs=ps.executeQuery();
while(rs.next())
{
int member_id=rs.getInt(1);
String member_name=rs.getString(2);
String member_mobno=rs.getString(3);
String member_Email=rs.getString(4);
String date_of_join=rs.getString(5);
String member_training_time=rs.getString(6);
int trainer_id=rs.getInt(7);

System.out.println("member_id \t member_name \t member_mobno \t member_Email \t
date_of_join \t member_training_time \t trainer_id);


System.out.println(member.getmember_id+ "\t" +member.getmember_name+ "\t"
+member.getmember_mobno+ "\t" +member.getmember_Email+ "\t" +member.getdate_of_join+
"\t" +member.getmember_training_time+ "\t" +member.gettrainer_id);
}
}
```

```java
}


import java.util.*;
import com.kce.trainer.util;
import com.kce.member.util;

class FitnesCenter
{
public static void main(String [] args)
{
Scanner sc=new Scanner(System.in);
System.out.println(1.Trainer\n2.Member);
System.out.println("ENTER YOUR CHOICE:");
int choice=sc.nextInt();
switch(choice)
{
case 1:
    FitnessTrainer ftrainer=new FitnessTrainer();
    Trainer trainer=ftrainer.createtable();
    ftrainer.insertintotable();
    int del_id=sc.nextInt();
    ftrainer.deletecolumn(del_id);
    int retrive_id=sc.nextInt();
    ftrainer.createtable(retrive_id);
    break;

case 2:
    FitnessMember fmember=new FitnessMember();
    Member member=fmember.createtable();
    fmember.insertintotable();
    int del_id=sc.nextInt();
    fmember.deletecolumn(del_id);
    int retrive_id=sc.nextInt();
    fmember.createtable(retrive_id);
    break;

default:
    System.out.println("invalid choice");
}
}
}



output table:
```

Member

member_id	member_name	member_mobno	member_Email	date_of_join	member_training_time	trainer_id
10101m	sam	9861571364	sam0258@gmail.com	15-02-2023	eve-6to8	7890t
10102m	thiru	8751729165	thirupath89@gmail.com	25-01-2023	eve-7to8:30	7896t
10103m	ranjani	9987624571	ranju8456@gmail.com	05-03-2023	mor-7to9	7892t
10104m	nakul	7694135254	nakulnath87@gmail.com	24-02-2023	mor-5to7	7890t

Trainer

trainer_id	trainer_name	trainer_mobno	year_of_experience
7890t	joshwa	8792485712	4
7896t	rajesh	7894424884	2
7892t	ramya	7971856541	2