**import** pymysql

**class** ConectarBd():

**def** conectar(self):

**try**:

**global** cur,conn

d= dict()

d[**'host'**]=**'localhost'**

d[**'port'**]=3306

d[**'user'**]=**'root'**

d[**'passwd'**]=**'1234567'**

d[**'db'**]=**'proyectosocio'**

self.conn= pymysql.connect(\*\*d) *#Con self las transformo en propiedades de la clase conectarBD*

self.cur = self.conn.cursor()

**return** 0

**except** Exception **as** f:

**return** 1 *#Si no conecta retorna 1 para que otro metodo sepa que la base de datos no conecto*

**class** Socio ():

**def** \_\_init\_\_(self,idSocio,dni,nombre,apellido):

self.idSocio=idSocio

self.dni=dni

self.nombre=nombre

self.apellido=apellido

**class** CDSocio ():

**def** \_\_init\_\_(self): *#Instancio una conexion*

self.conexion = ConectarBd()

a = self.conexion.conectar()

print(a)

**def** alta(self,socioAlta):

consul1=**"insert into socio (dni,nombre,apellido) values({0},{1},{2})"**.format(repr(socioAlta.dni), repr(socioAlta.nombre),repr(socioAlta.apellido))

self.conexion.cur.execute(consul1)

self.conexion.conn.commit()

**return** self.conexion.cur.lastrowid *#DEVULVE EL ULTIMO ID, NO UN OBJETO socioAlta!!!*

**def** borrar(self,socioBorrar):

consul2=**"delete from socio where idsocio = %d"** %socioBorrar.idSocio

self.conexion.cur.execute(consul2)

self.conexion.conn.commit()

**if** self.conexion.cur.rowcount == 0:

**return False**

**else**:

**return True**

**def** modificar(self,socioModificar): *#Revisar que pasa cuando no se modifico*

consul3=**'update socio set dni={0}, nombre={1}, apellido={2} where idsocio = {3}'**.format(repr(socioModificar.dni), repr(socioModificar.nombre), repr(socioModificar.apellido), socioModificar.idSocio)

self.conexion.cur.execute(consul3)

self.conexion.conn.commit()

**return** socioModificar

**def** buscarxId(self,idSocio):

consul4=**'select \* from socio where idsocio=%d'** %idSocio

self.conexion.cur.execute(consul4)

socioEnco= self.conexion.cur.fetchone()

**return** socioEnco

**def** todos(self):

consul5 = **"SELECT \* FROM socio"**

self.conexion.cur.execute(consul5)

socios = self.conexion.cur.fetchall()

**return** socios

**def** anterior(self, id):

consul6 = **"SELECT \* FROM socio WHERE idsocio=(SELECT MAX(idsocio) FROM socio WHERE idsocio < {0})"**.format(id)

self.conexion.cur.execute(consul6)

socEnco = self.conexion.cur.fetchone()

**if** (self.conexion.cur.rowcount == 0):

**return False** *#Esto es malo. No lo encuentra*

**else**:

**return** socEnco

**def** proximo(self, id):

consul7 = **"SELECT \* FROM socio WHERE idsocio=(SELECT MIN(idSocio) FROM socio WHERE idsocio > {0})"**.format(id)

self.conexion.cur.execute(consul7)

socEnco = self.conexion.cur.fetchone()

**if** self.conexion.cur.rowcount == 0:

**return False** *#Esto es malo*

**else**:

**return** socEnco

**def** primero(self):

consul8 = **"SELECT \* FROM socio WHERE idsocio=(SELECT MIN(idsocio) FROM socio)"**

self.conexion.cur.execute(consul8)

socEnco = self.conexion.cur.fetchone()

**if** self.conexion.cur.rowcount == 0:

**return False**

**else**:

**return** socEnco

**def** ultimo(self):

consul9 = **"SELECT \* FROM socio WHERE idsocio=(SELECT MAX(idsocio) FROM socio)"**

self.conexion.cur.execute(consul9)

socEnco = self.conexion.cur.fetchone()

**if** self.conexion.cur.rowcount == 0:

**return False**

**else**:

**return** socEnco

**def** buscaDni(self,dni):

consul10 = **'select count(\*) from socio where dni = {0}'**.format(dni)

self.conexion.cur.execute(consul10)

cant = self.conexion.cur.fetchone()

**return** cant[0]

**def** validaCantidad(self):

consul11 = **'select count(\*) from socio'**

self.conexion.cur.execute(consul11)

cant = self.conexion.cur.fetchone()

**return** cant[0]