**from** CapaDatosPersona **import** CDSocio

**from** Entidades **import** s

**class** CNSocio():

**global** cd

cd = CDSocio()

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

**if**(self.validaDni(s)== **True and** self.validaNombre(s)==**True and** cd.validaCantidad()<=200 **and** cd.buscaDni(s.dni)==0):

cd.alta(s)

**return True**

**else**:

**return False**

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

**if**(self.validaDni(s)==**True and** cd.buscaDni(s.dni)==1):

**if**(cd.borrar(s)==**True**):

**return True**

**else**:

**return False** *#No pudo borrar*

**else**:

**return False** *#No cumple con las condiciones*

**def** modificar(self,s):

**if**(self.validaDni(s)== **True and** self.validaNombre(s)==**True and** cd.buscaDni == 0):

socioModif = cd.modificar(s)

**return** socioModif

**else**:

**return False**

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

**if** (id != **''**):

socEnco = cd.buscarxId(id)

**if** socEnco == **None**:

**return False** *#No existe es Id.*

**else**: **return** socEnco

**def** todos(self):

**return** cd.todos()

**def** validaDni(self, s):

**if**(s.dni != **'' and** isinstance(s.dni,int)):

**return True** *#Esto es bueno*

**else**: **return False**

**def** validaNombre(self,s):

**if**(isinstance(s.nombre,str) **and** isinstance(s.apellido,str) **and** len(s.nombre)<25 **and** len(s.apellido)<25 **and** len(s.nombre)>3 **and** len(s.apellido)>3):

**return True** *#Esto es bueno*

**else**:

**return False**

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

socEnco = cd.anterior(id)

**if** socEnco == **None**:

**return** self.primero() *#No se si afecta o no que la funcion primero esté despues de esta llamada*

**else**:

**return** socEnco

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

socEnco = cd.proximo(id)

**if** socEnco == **None**:

**return** self.ultimo()

**else**:

**return** socEnco *#No se si afecta o no que la funcion ultimo esté despues de esta llamada*

**def** primero(self):

socEnco = cd.primero()

**if** socEnco == **None**:

**return False** *#No existen socios entonces no puede devolver el primero*

**else**:

**return** socEnco

**def** ultimo(self):

socEnco = cd.ultimo

**if** socEnco == **None**:

**return False** *#"No existe" el ultimo socio, no habría ninguno*

**else**:

**return** socEnco

cn2 = CNSocio()

print(cn2.modificar(s))