

Leonel Antonio González García 201709088

```
nodo.py × lista_doble.py app.py Persona.py
nodo.py > nodo > _init_
1 class nodo:
2     def __init__(self, persona = None, siguiente = None, anterior = None):
3         self.persona = persona
4         self.siguiente = siguiente
5         self.anterior = anterior
```

```
lista_doble.py > ...
1 from nodo import nodo
2
3 class lista_doble:
4
5     def __init__(self):
6         self.primeros = None
7
8
9     def recorrer(self):
10        if self.primeros is None:
11            print("La lista se encuentra vacía")
12            return
13        actual=self.primeros
14
15        print("ID: ", actual.persona.id, "Nombre: ", actual.persona.nombre, "Edad: ", actual.persona.edad)
16        while actual.siguiente:
17            actual=actual.siguiente
18            print("ID: ", actual.persona.id, "Nombre: ", actual.persona.nombre, "Edad: ", actual.persona.edad)
19
20
21    def insertar_ordenado(self, persona):
22        nuevo_nodo = nodo(persona = persona)
23        if self.primeros is None:
24            self.primeros = nuevo_nodo
25            return
26        if persona.edad < self.primeros.persona.edad:
27            nuevo_nodo.siguiente = self.primeros
28            self.primeros.anterior = nuevo_nodo
29            self.primeros = nuevo_nodo
30            return
31        actual = self.primeros
32        while actual.siguiente and actual.siguiente.persona.edad < persona.edad:
33            actual = actual.siguiente
34        nuevo_nodo.siguiente = actual.siguiente
35        if actual.siguiente:
36            actual.siguiente.anterior = nuevo_nodo
37        actual.siguiente = nuevo_nodo
38        nuevo_nodo.anterior = actual
39
```

```
nodo.py  lista_doble.py  app.py  X  Persona.py
app.py > ...
1  from Persona import Persona
2  from lista_doble import lista_doble
3
4  persona1= Persona(202001534,"Lionel Messi", 36)
5  persona2= Persona(202100000,"Cristiano Ronaldo", 38)
6  persona3= Persona(202200000,"Ronaldinho Gaucho", 43)
7
8  lista_doble = lista_doble()
9
10 lista_doble.insertar_ordenado(persona1)
11 lista_doble.insertar_ordenado(persona2)
12 lista_doble.insertar_ordenado(persona3)
13 lista_doble.recorrer()
```

```
nodo.py  lista_doble.py  app.py  Persona.py X
Persona.py > ...
1  |
2  class Persona:
3
4      def __init__(self,id,nombre,edad,):
5          self.id = id
6          self.nombre = nombre
7          self.edad = edad
8
```