

Clase y constructor de un nodo.

```
C: > Users > Garcia > Desktop > IPC2 > nudo.py
1 class Nodo():
2     #Constructor de Nodod para los datos de la lista
3     #Nodo siguiente y Nodo aterior
4     def __init__(self,elemento):
5         self.elemento = elemento
6         #En una lista Enlazada doble Es necesario un nodo siguiente y un
7         #nodo anterior como apuntadores.
8         self.siguiente = None
9         self.anterior = None
10
```

Clase y constructor de una lista doblemente enlazada.

```
C: > Users > Garcia > Desktop > IPC2 > lista_doble_enlazada.py
1 from nodo import Nodo
2
3 class ListaDoblementeEnlazada:
4     def __init__(self):
5         #Apuntadores de la lista
6         self.primeros = None
7         self.ultimo = None
8         self.size = 0
9
```

Función para insertar nodo a la lista vacía.

```
#Metodo de la lista cuando esta Vacía
#Insertando datos a la lista doblemente enlazada
def listavacia(self, elemento):
    #Sí la lista está vacía se crea un nuevo nodo
    if self.root = None:
        nuevoNodo = Nodo (elemento)
        # El root de la lista apunta al nuevo nodo
        self.root = nuevoNodo
    else:
        print("La lista está vacia")
```

Función para insertar nodo al inicio de la lista

```
#Insertando datos al inicio de la lista
def agregar_inicio(self, elemento):
    if self.root is None:
        #Usamos la funcion listavacia
        self.listavacia(elemento)
        return
    else:
        nuevoNodo = Nodo(elemento)
        # Se asigna como siguiente nodo de el nuevo nodo al root
        nuevoNodo.siguiente = self.root
        self.root.anterior = nuevoNodo
        self.root = nuevoNodo
```

Función para insertar nodo al final de la lista

```
# Metodo para Agregar un Elemento Al Final de la Lista
def agregar_final(self, elemento):
    # Se Verifica si la lista esta Vacía
    if self.root is None:
        nuevoNodo = Nodo(elemento)
        self.root = nuevoNodo
        return
    #Creamos un apuntador al root si la lista no esta vacia
    apuntador = self.root
    # Se mueve a travez de la lista hasta que el ultimo nodo sea none
    while apuntador.siguiente is not None:
        apuntador = apuntador.siguiente
    # Se crea un nuevo nodo
    nuevoNodo = Nodo(elemento)
    # Se agrega el nodo al final de la fila
    apuntador.siguiente = nuevoNodo
    #Se conecta el nuevo nodo al ultimo que existia
    nuevoNodo.anterior = apuntador
```