IMPLEMENTACIÓN LISTA DOBLEMENTE ENLAZADA

PRESENTADO POR:

MIGUEL ANGEL SALAMANCA

JUAN CAMILO BAZURTO ARIAS

PRESENTADO A:

SEBASTIAN CAMILO MARTINEZ REYES

ESCUELA COLOMBIANA DE INGENIERÍA JULIO GARAVITO

ALGORITMOS Y ESTRUCTURAS DE DATOS

PROGRAMA DE INGENIERÍA DE SISTEMAS

BOGOTÁ D.C.

2021 – 1

1. **Contexto**

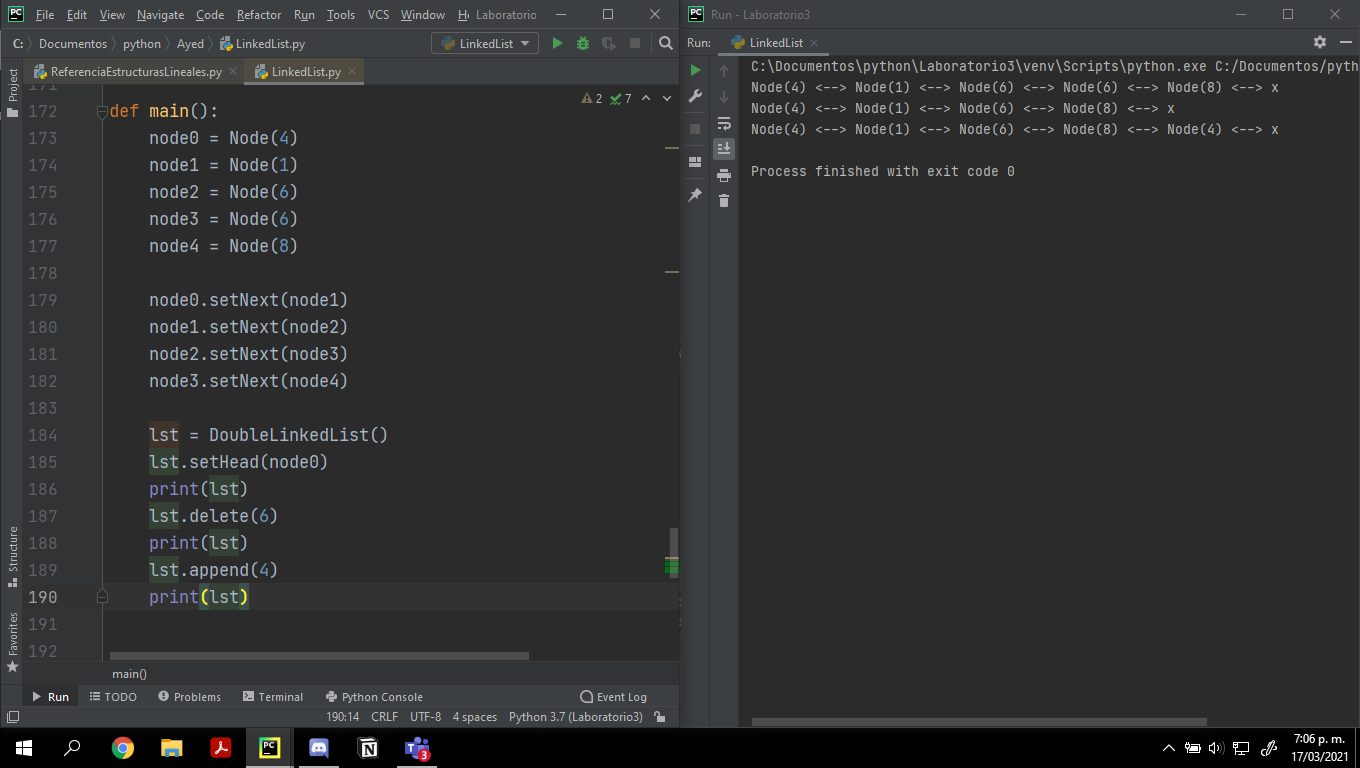
Realizar la implementación de la lista doblemente enlazada usando como referencia el código adjunto ( DoubleLinkedList ) extendiendo la definición de Nodo a tener un apuntador a aun nodo previo

1. **Diseño**

Para implementar la lista doblemente enlazada, agregamos un nuevo atributo al constructor el cual es el nodo previo, y se agregaron dos nuevos métodos los cuales tienen como objetivo obtener el valor previo (get), y asignar el valor previo (set).

* 1. Casos de Prueba

Creamos una lista doblemente enlazada mediante código.



1. **Análisis**
   1. Temporal

* Los métodos get y set de nuestra implementación tiene complejidad O(1).
* El método delete tiene complejidad O(n)
* El método para eliminar duplicados tiene complejidad O(n).
* El método para combinar dos DoubleLinkedList, tiene complejidad O(1).
* El método append tiene complejidad O(n).
* El método reverse tiene complejidad O(n).
* El método length tiene complejidad O(n).

IMPLEMENTACIÓN DE PILA Y COLA

PRESENTADO POR:

MIGUEL ANGEL SALAMANCA

JUAN CAMILO BAZURTO ARIAS

PRESENTADO A:

SEBASTIAN CAMILO MARTINEZ REYES

ESCUELA COLOMBIANA DE INGENIERÍA JULIO GARAVITO

ALGORITMOS Y ESTRUCTURAS DE DATOS

PROGRAMA DE INGENIERÍA DE SISTEMAS

BOGOTÁ D.C.

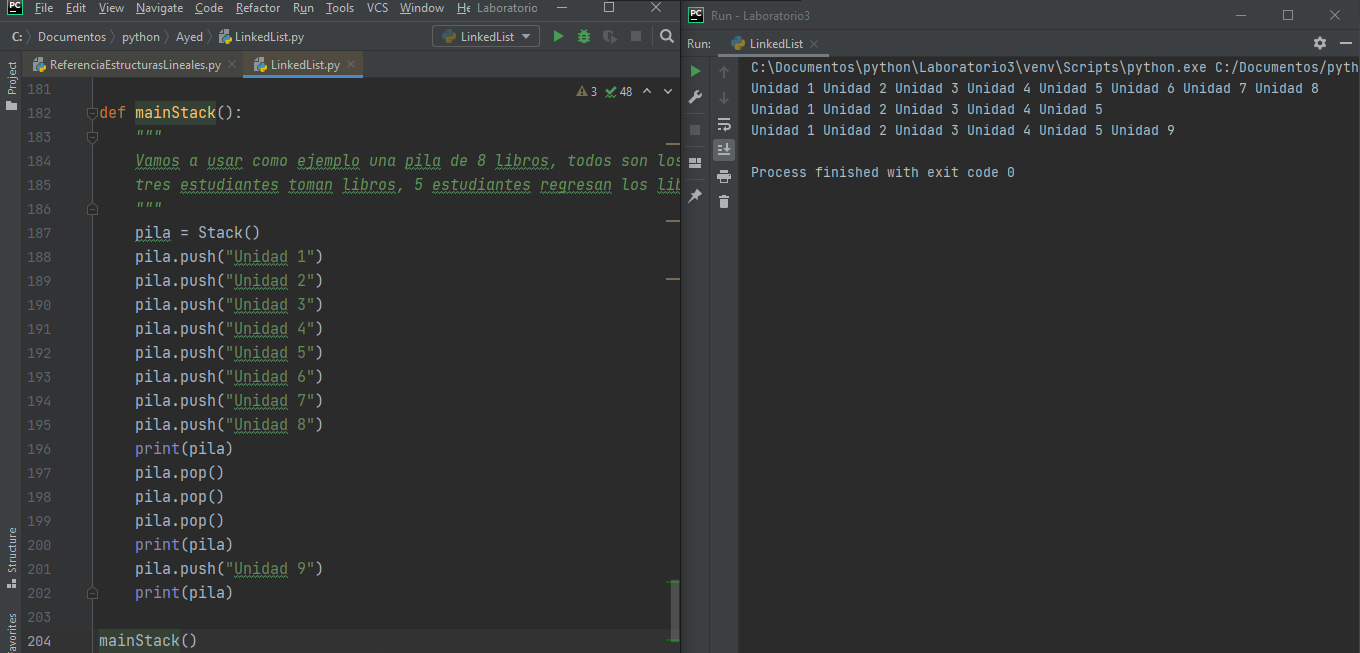
2021 – 1

1. **Contexto**

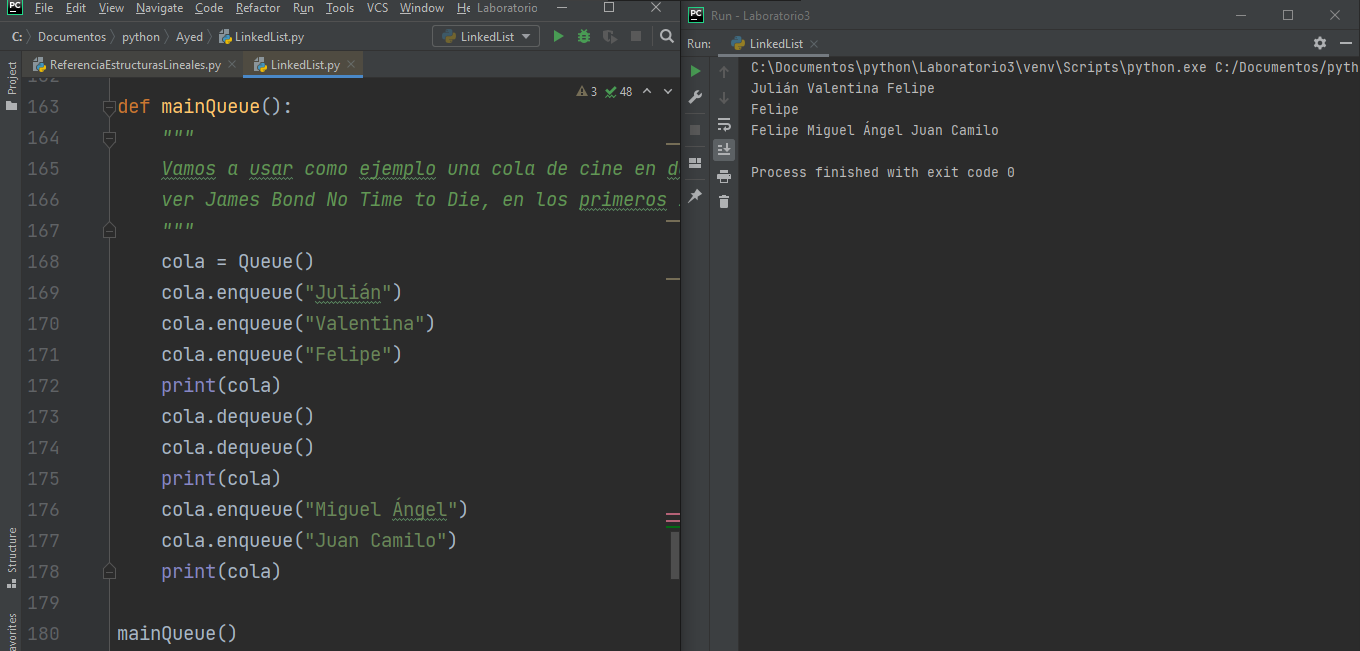
Realizar la implementación de Pila y Cola (Queue, Stack) en donde en su definición estructural se reúse la definición realizada en el punto 1. En las operaciones CRUD recuerde mantener la política de cada estructura (LIFO - FIFO).

1. **Diseño**
2. Para implementación de Pila, esta clase stack hereda todos los atributos y métodos de DoubleLinkedList, lo que cambia es la forma en que se agregar y eliminan valores, dado que las Pilas tienen como política LIFO para agregar nuevos elementos y LIFO para eliminar.
3. Para implementación de Cola, esta clase queue hereda todos los atributos y métodos de DoubleLinkedList, lo que cambia es la forma en que se agregar y eliminan valores, dado que las Colas tienen como política LIFO para agregar nuevos elementos y FIFO para eliminar.
   1. Casos de Prueba

Como caso de prueba se utilizó la simulación de una pila de libros donde ingresan 5 libros, se prestan 3 de estos y devuelven uno.



Como caso de prueba se utilizó la simulación de una cola de cine donde dos personas son atendidas y llegan otras dos.



1. **Análisis**
   1. Temporal

* El método push tiene complejidad O(n).
* El método pop tiene complejidad O(n).
* El método enqueue tiene complejidad O(n).
* El método dequeue tiene complejidad O(1).