**Practica 1 Tema 6**

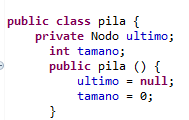
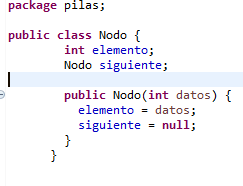
**Aplicación de las estructuras de almacenamiento**

**Índice**

1. **Lista enlazada pila………………………………………1**
2. **Métodos pila……………………………………………1-3**
   1. **Insertar datos……………………………………1-2**
   2. **Recuperar datos…………………………………….2**
   3. **Obtener datos………………………………………..2**
   4. **Devolver datos………………………………………2**
   5. **Más datos………………………………………….3-4**
3. **Listas enlazadas cola…………………………………….4**
   1. **Insertar datos……………………………………4-5**
   2. **Recuperar datos…………………………………….5**
   3. **Obtener datos………………………………………..5**
   4. **Devolver datos……………………………………5-6**
   5. **Más datos…………………………………………..6-7**
4. **Conclusión……………………………………………………7**

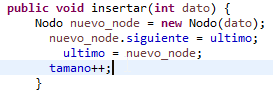
* **RESUELVE**

1. Implementa en Java una pila utilizando una lista enlazada como estructura de datos.



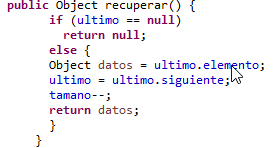
**La segunda diapositiva es la lista enlazada con la variable que almacena los datos y siguiente que apunta al próximo valor que por defecto es nulo y en la primera diapositiva es con lo que se trabajara principalmente para indicar cuáles serán los valores.**

1. Implementa el método necesario para:
2. Insertar un dato en la pila



**Lo que hace el método es almacenar el valor en el objeto creado y posteriormente en ultimo para luego aumentar el tamaño en 1**

1. Recuperar un dato de la pila



**Este método es un poco más complejo ya que lo que hace primero es comprobar si ultimo vale nulo, en caso de serlo que devuelva nulo y si no lo es, lo que se hace es crear un objeto que almacene el ultimo valor, que es el que se borrara, y luego pasa con el apuntador .Siguiente, a en vez de valer 4 y 5 por ejemplo, a valer 5 y nulo para almacenarlo en último, por lo cual ya se ha borrado el elemento, luego de esto el contador tamaño pasa a restar en 1.**

1. Obtener el dato de la cima de la pila

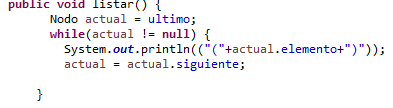


**El método lo que hace es devolver el elemento de la cima de la pila, último es lo que indica cual es y .elemento es el valor en sí, y como es tipo lifo el último en entrar es el primero en salir por lo cual se mostrara en la cima el ultimo introducido**

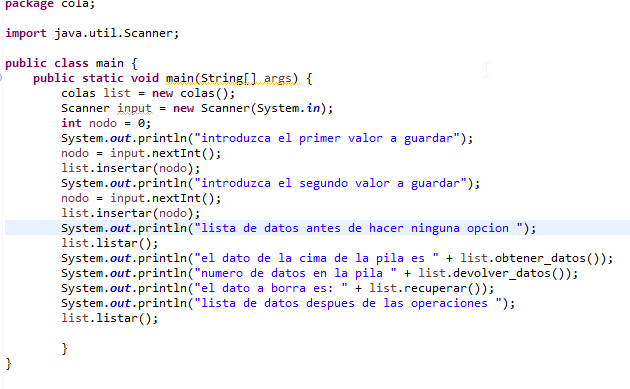
1. Devolver el dato de la cima de la pila



**El método devuelve la variable tamaño que es como un contador que disminuye o aumenta según qué método se haga como se ve en las diapositivas de más arriba. Se entiende que lo que se pedía en este apartado era devolver el número de elementos que hay, no el elemento de la cima, ya que en el anterior apartado ya se pedía.**

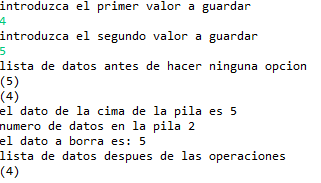


**También se ha implementado un método que lo que hace es listar los elementos de la cola o de la pila según en qué caso este, en este caso sería la pila. Lo que hace es, con el objeto creado se guarda el último elemento y con él .siguiente que sería el puntero, lo descarga, es decir, si valía 4 y nodo (5), pasa a valer 5 y nulo y en la siguiente pasada nulo en ambos, y ya que el bucle se rompe una vez el objeto valga nulo se romperá, parara. El system lo que imprime es el valor del objeto por eso el .elemento.**

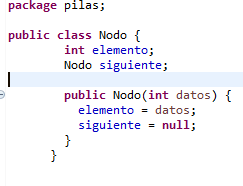
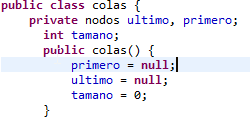
****

**Se ha creado un main a parte para clasear en ambas partes tanto en colas como en pilas, es el mismo main en ambos casos. Se ha puesto un scanner para que se puedan introducir los datos pidiéndolos, para que no sean predefinidos desde el mismo código y para que el valor introducido se almacene en el método correspondiente, se ha llamado al método colocándolo después del scanner y entre paréntesis el valor que se ha pedido (nodo), los demás métodos se han podido colocar en los system ya que son métodos no vacíos, excepto listar e insertar y por eso se colocaron afuera.**

**Resultados:**

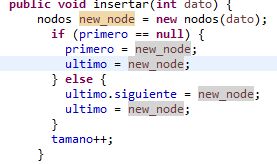


1. Implementar todo lo anterior para una cola.



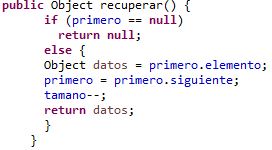
**La segunda diapositiva es la lista enlazada con la variable que almacena los datos y siguiente que apunta al próximo valor que por defecto es nulo y en la primera diapositiva es con lo que se trabajara principalmente para indicar cuáles serán los valores.**

1. Insertar un dato en la cola



**Este método es más complejo que el de pila, lo que hace es: comprobar que primero valga nulo, al valer nulo almacena el valor introducido en primero y ultimo pero con el puntero siguiente en nulo, esto es así porque al ser solo un valor claramente es el primero y el ultimo hasta que se introduzca un nuevo valor, en primero estará en elemento el primero y el apuntador el nodo y en el último estará en elemento el valor ultimo y en siguiente nulo por lo cual donde se almacena realmente todo es en las listas enlazadas, en este caso elemento y siguiente el apuntador que marca, como su propio nombre indica, al siguiente valor ya sea nulo o no, al introducir un nuevo valor realizara else que almacena el ultimo valor.**

1. Recuperar un dato de la cola



**Este método es un poco más complejo ya que lo que hace primero es comprobar si primero vale nulo, en caso de serlo que devuelva nulo y si no lo es, lo que se hace es crear un objeto que almacene el primer valor, que es el que se borrara, y luego pasa con el apuntador .Siguiente a en vez de valer 4 y 5 por ejemplo a valer 5 y nulo para almacenarlo en primero, por lo cual ya se ha borrado el elemento, luego de esto el contador tamaño pasa a restar en 1.**

1. Obtener el dato de la cima de la cola

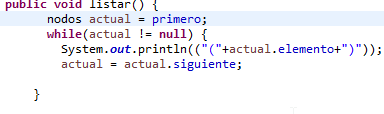


**El método lo que hace es devolver el primer elemento de la cola, de ahí el primero que indica que es el primero como su propio nombre indica y elemento es el valor en sí, si se quisiera el último sería: ultimo.elemento. Al ser FIFO que el primer dato introducido es el primero que sale, por lo cual en la cima estará el primer dato que se introdujo.**

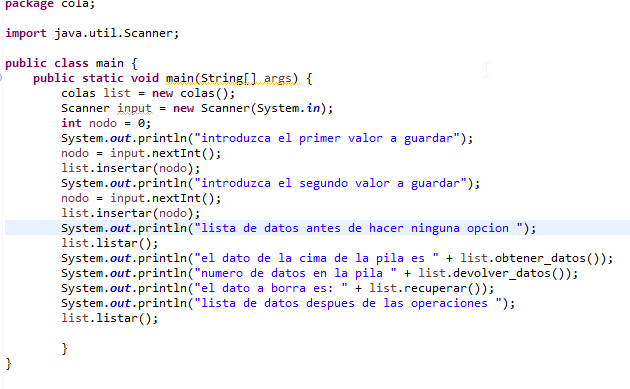
1. Devolver el dato de la cima de la cola



**El método devuelve la variable tamaño que es como un contador que disminuye o aumenta según qué método se haga como se ve en las diapositivas de más arriba. Se entiende que lo que se pedía en este apartado era devolver el número de elementos que hay, no el elemento de la cima, ya que en el anterior apartado ya se pedía.**

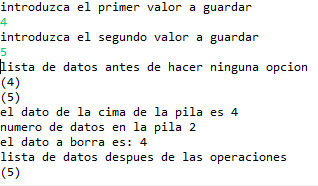


**También se ha implementado un método que lo que hace es listar los elementos de la cola o de la pila según en qué caso este, en este caso sería la cola. Lo que hace es, con el objeto creado se guarda el primer elemento y con él .siguiente que sería el puntero, lo descarga, es decir, si valía 4 y nodo (5), pasa a valer 5 y nulo y en la siguiente pasada nulo en ambos, y ya que el bucle se rompe una vez el objeto valga nulo se romperá. El system lo que imprime es el valor del objeto por eso el .elemento.**



**Se ha creado un main a parte para clasear en ambas partes tanto en colas como en pilas, es el mismo main en ambos casos. Se ha puesto un scanner para que se puedan introducir los datos pidiéndolos, para que no sean predefinidos desde el mismo código y para que el valor introducido se almacene en el método correspondiente, se ha llamado al método colocándolo después del scanner y entre paréntesis el valor que se ha pedido (nodo), los demás métodos se han podido colocar en los system ya que son métodos no vacíos, excepto listar e insertar y por eso se colocaron afuera.**

**Resultados:**



**Conclusión**

**El ejercicio ha sido entretenido, aunque estas clases y métodos han sido un poco difíciles de entender, se ha tenido algún que otro problema a la hora de la estructura ya que como se ha dicho ya, era un poco compleja y hasta que no se entendió del todo hubo problemas, una vez entendido el concepto y como funciona un hubo ningún problema. Se ha utilizado el debugg para una mayor comprensión de estas clases y métodos además de los videos del temario entre otros.**