

Instruction to candidates:

Your program code and output for each of Task 1 to N should be saved in a single . ipynb file using Jupyter Notebook. For example, your program code and output for Task 1 should be saved as:

Task1_<your name>_<centre number>_<index number>.ipynb

Make sure that each of your . ipynb files shows the required output in Jupyter Notebook.

1 Name your Jupyter Notebook as

Task1_<your name>_<centre number>_<index number>.ipynb

A data structure is required to store nodes. A linked list is maintained of all the nodes. A node contains a data value and a pointer, which is initially set to `None`. Subsequently, items in the list are linked using the pointer.

Each node is implemented as an instance of the class `ConnectionNode`. The class `ConnectionNode` has the following properties:

Class: <code>ConnectionNode</code>		
Attributes		
Identifier	Data Type	Description
<code>data</code>	<code>STRING</code>	The node data
<code>next</code>	<code>CLASS</code>	The node pointer

The structure for the linked list `LinkedList` is implemented as follows:

Identifier	Data Type	Description
<code>head</code>	<code>CLASS</code>	Initially set to <code>None</code> , this points to the first node in the list.

`LinkedList` also has the following methods:

Identifier	Method Type	Description
<code>isEmpty</code>	<code>FUNCTION</code> <code>RETURNS BOOLEAN</code>	Returns whether the list is empty.
<code>length</code>	<code>FUNCTION</code> <code>RETURNS INTEGER</code>	Returns the number of nodes in the list.
<code>insertEnd</code>	<code>PROCEDURE</code>	Inserts a node at the end of the list.
<code>insertHead</code>	<code>PROCEDURE</code>	Inserts a node before the first node of the list.
<code>insertAt</code>	<code>PROCEDURE</code>	Inserts a node at the specified position. The node at the head of the list has position 0.
<code>deleteAt</code>	<code>PROCEDURE</code>	Deletes a node at the specified position.
<code>deleteEnd</code>	<code>PROCEDURE</code>	Deletes the last node.
<code>printList</code>	<code>PROCEDURE</code>	Prints the data value of each node in the list.

For each of the sub-tasks, add a comment statement at the beginning of the code using the hash symbol '#', to indicate the sub-task the program code belongs to, for example:

```
In [1] : # Task 1.1
Program code
```

Output:

Task 1.1

Write program code that implements `ConnectionNode` and `LinkedList`.

[20]

Task 1.2

Test your implementation of `LinkedList` by performing in sequential order the following actions:

- creating a `LinkedList` object, `linkedlist`
- inserting at the end of `linkedlist` first the node 'network', then the node 'transmission' and lastly the node 'receiver'
- printing `linkedlist`
- deleting the third node of `linkedlist`
- inserting at the start of `linkedlist` the node 'wireless'
- inserting the node 'data' as the third node of `linkedlist`
- inserting the node 'router' as the third node `linkedlist`
- deleting the last node of `linkedlist`
- printing `linkedlist`.

[10]

Save your Jupyter Notebook for Task 1.