**Queue LinkedList**

**Interface** **QueueLL**

|  |
| --- |
| **interface** QueueLL  {  **public** **void** enqueue(String str);  **public** String peek();  **public** String dequeue();  **public** Boolean isEmpty();  } |

**Clas Node**

|  |
| --- |
| **class** Node  {  **public** String value;  **public** Node next;    **public** Node(String val, Node n)  {  value = val;  next = n;  }  } |

**Class StringQueue implements QueueArr interface**

|  |
| --- |
| **class** StringQueue **implements** QueueArr  {  **private** Node front = **null**;  **private** Node rear = **null**;  >>>>> *THE REST OF THE METHODS GO HERE* <<<<<  } |

**The isEmpty() Method**

|  |
| --- |
| **Public Boolean isEmpty**  {  **Return** (front == null);  } |

**The enqueue() Method**

|  |
| --- |
| **public** **void** enqueue(String str)  {  **if**(rear != **null**)  {  rear.next = **new** Node(str, **null**);  rear = rear.next;  }  **else**  {  rear = **new** Node(str, **null**);  front = rear;  }  } |

**The peek() Method**

|  |
| --- |
| **public** **String peek**()  {  String temp = null;  **if**(isEmpty())  **temp =** **null**; // throw new QueueEmptyException();  **else**  **temp =** front.value;  **return** temp;  } |

**The dequeue() Method**

|  |
| --- |
| **public** String dequeue()  {  String temp = null;  **if**(isEmpty())  **temp =** **null**; // throw new QueueEmptyException();  **else**  **{**  temp = front.value;  front = front.next;  **if**(front == **null**)  rear = **null**;  }  **return** temp;  } |

**The isEmpty() Method**

|  |
| --- |
| **public** **boolean** isEmpty()  {  **return** (front == **null**);  } |

**The toString() Method**

|  |
| --- |
| **public** String toString()  {  StringBuilder builder = **new** StringBuilder();  **for**(Node ref = front; ref != **null**; ref = ref.next)  builder.append(ref.value + " ");  **return** builder.toString();  } |

**The main() Method**

|  |
| --- |
| **public** **class** PointApp  {  **public** **static** **void** main(String[] args)  {  StringQueue theQueue = **new** StringQueue();    String[] names = {"One", "Two", "Three", "Four"};  **for**(String s : names)  theQueue.enqueue(s);  System.*out*.println(theQueue); // One Two Three Four    theQueue.dequeue(); theQueue.dequeue();  System.*out*.println(theQueue); // Three Four    theQueue.enqueue("Five");  System.*out*.println(theQueue); // Three Four Five  }  } |