```
class MyLL:
    class mynodes:
        def __init__(self,data):
            self.data = data
            self.next = None
    def __init__(self):
        self.head=None
    def insertb(self,data):
        new_node =MyLL.mynodes(data)
        new_node.next = self.head
        self.head = new_node
    def inserted(self,data):
        new_node=MyLL.mynodes(data)
        if self.head == None:
            self.head = new_node
            return
            current = self.head
        while current.next:
            current =current.next
            current.next=new_node
    def insertaft(self,data,prev_node):
        new_node = MyLL.mynodes(data)
        new_node.next = prev_node.next
        prev_node.next = new_node
    def print_LL(self):
        current=self.head
        while current:
            print(current.data, end='-->')
            current = current.next
    def search(self,val):
        current=self.head
        while current:
            if current.data ==val:
                return True
                current = current.next
            return False
    def deletefirst(self):
        if self.head == None:
            return
            current=self.head
            while current.next.next:
                current=current.next
                current.next=None
                print(current)
    def del_by_val(self,x):
        if self.head.data == x:
            self.head = self.head.next
            current = self.head
        while current.next:
            if current.next.data!=x:
                current = current.next
            else:
                break
        if current.next:
                current.next = current.next.next
        else:
            return 'not found'
    def reverse(self):
        prev = None
        current = self.head
        while current:
            next_node = current.next
            current.next = prev

j =MyLL()
j.insertb(55)
j.insertb(40)
j.insertb(90)
j.print_LL()
```

```
90-->40-->55-->
```

```
j.search(40)
```

```
False
```

```
j.del_by_val(90)
j.print_LL()
```

```
40-->55-->
```

```
def move_hyphens_to_front(str):
    hyphens = " "
    others = " "
    for char in str:
        if char =='_':
            hyphens=hyphens+char
        else:
            others=others+char
    return hyphens+others


input_str = "hello-hi-sir"
output_str = move_hyphens_to_front(input_str)
print(output_str)
```

```
  hello-hi-sir
```