

Nama :Ahmad wahyu abadi setiawan

Kelas : IX RPL

No absen : 02

Jawaban Soal Teori

Jawaban Soal Teori

1. Perbedaan Mendasar Penambahan Data di Tengah (Array vs. Linked List):

- **Array:** Untuk menyisipkan data di tengah array, semua elemen yang berada setelah posisi sisipan harus **digeser** satu per satu ke kanan untuk menciptakan ruang kosong. Proses penggeseran ini bisa sangat lambat dan tidak efisien (memakan waktu komputasi), terutama jika array sangat besar.
- **Linked List:** Untuk menyisipkan data di tengah, kita hanya perlu mengubah **pointer** atau "sambungan". Kita temukan node sebelumnya, lalu kita atur pointer dari node tersebut untuk menunjuk ke node baru, dan pointer dari node baru kita arahkan ke node berikutnya. Tidak ada proses penggeseran data sama sekali, sehingga jauh lebih cepat dan efisien.

2. Fungsi dan Pentingnya **head**: **head** adalah **pintu gerbang** atau titik awal dari sebuah Linked List. Fungsinya adalah sebagai referensi atau alamat dari node (simpul) pertama dalam rantai. Ia sangat penting karena tanpa **head**, kita akan kehilangan akses ke seluruh rantai data. Jika kita tidak tahu di mana node pertama berada, kita tidak akan pernah bisa menemukan node kedua, ketiga, dan seterusnya, karena satu-satunya cara menelusuri Linked List adalah dengan mengikuti petunjuk (**next**) dari satu node ke node berikutnya, dimulai dari **head**

Studi Kasus Praktik

```
class Node:
```

```
    def __init__(self, data):
```

```
        self.data = data
```

```
        self.next = None
```

```
class LinkedList:

    def __init__(self):

        self.head = None

    def print_list(self):

        print("Isi Playlist:")

        current_node = self.head

        if current_node is None:

            print("(Playlist kosong)")

            return

        while current_node is not None:

            print(f"- {current_node.data}")

            current_node = current_node.next

    def append(self, data):

        new_node = Node(data)

        if self.head is None:

            self.head = new_node

            return

        last_node = self.head

        while last_node.next:

            last_node = last_node.next

        last_node.next = new_node
```

```
def prepend(self, data):  
    new_node = Node(data)  
    new_node.next = self.head  
    self.head = new_node  
  
def delete_node(self, key):  
    current_node = self.head  
    if current_node and current_node.data == key:  
        self.head = current_node.next  
        current_node = None  
        return  
    prev = None  
    while current_node and current_node.data != key:  
        prev = current_node  
        current_node = current_node.next  
    if current_node is None:  
        print(f"Error: Data '{key}' tidak ditemukan.")  
        return  
    prev.next = current_node.next  
    current_node = None  
  
def insert_after(self, prev_node_data, new_data):  
    current_node = self.head
```

```
while current_node:
    if current_node.data == prev_node_data:
        new_node = Node(new_data)
        new_node.next = current_node.next
        current_node.next = new_node
        print(f"Sukses: '{new_data}' disisipkan setelah '{prev_node_data}'")
        return
    current_node = current_node.next
print(f"Error: Data pendahulu '{prev_node_data}' tidak ditemukan.")
```

```
playlist = LinkedList()
playlist.append("Lagu Pop - A")
playlist.append("Lagu Rock - B")
playlist.append("Lagu Jazz - C")
playlist.print_list()
```

```
print("\nMenambah lagu intro...")
playlist.prepend("Intro Musik")
playlist.print_list()
```

```
print("\nMenyisipkan 'Lagu Iklan' setelah 'Intro Musik'...")
playlist.insert_after("Intro Musik", "Lagu Iklan")
playlist.print_list()
```

```
print("\nMenghapus 'Lagu Rock - B'...")
```

```
playlist.delete_node("Lagu Rock - B")
```

```
playlist.print_list()
```

```
print("\nMencoba menghapus lagu yang salah...")
```

```
playlist.delete_node("Lagu Dangdut")
```

```
playlist.print_list()
```

```
print("\nMenambah lagu penutup...")
```

```
playlist.append("Outro Musik")
```

```
playlist.print_list()
```

```
print("\nMembuat playlist baru dan mengeceknya...")
```

```
playlist_kosong = LinkedList()
```

```
playlist_kosong.print_list()
```