

LATIHAN LINKED LIST PERTEMUAN 3

Nama: Haikal Faqih Aufa

NIM: J0403251021

Kelas: B

Kelas Praktikum: P1

1. LATIHAN 1

A. SS

```

1  #!/usr/bin/perl -w
2
3  # Modul untuk melakukan koneksi ke database MySQL
4  # Nama user: root, password: root
5  # Nama database: db_name
6  # Nama tabel: db_name
7
8  # Import modul DBI dan DBD::MySQL
9  use DBI;
10 use DBD::MySQL;
11
12 # Buat koneksi ke database
13 my $dbh = DBI->connect(
14     "dbi:mysql:database=db_name;host=localhost",
15     "root",
16     "root",
17     {
18         RaiseF => 1,
19     }
20 );
21
22 # Cek apakah koneksi berhasil
23 if ($dbh->ping) {
24     print "Koneksi berhasil!\n";
25 } else {
26     print "Gagal koneksi!\n";
27 }
28
29 # Buat tabel jika belum ada
30 my $sql = "CREATE TABLE IF NOT EXISTS users (
31     id INT(4) UNSIGNED ZEROFILL AUTO_INCREMENT PRIMARY KEY,
32     username VARCHAR(16) CHARACTER SET utf8 COLLATE utf8_general_ci NOT NULL,
33     password VARCHAR(16) CHARACTER SET utf8 COLLATE utf8_general_ci NOT NULL,
34     email VARCHAR(48) CHARACTER SET utf8 COLLATE utf8_general_ci NOT NULL,
35     register_time DATETIME NOT NULL
36 );";
37
38 # Eksekusi query
39 $dbh->do($sql);
40
41 # Tutup koneksi
42 $dbh->disconnect;
43
44 # Akhir program
45 exit 0;

```

B. OUTPUT

```
PS E:\ARSIP AKADEMIK IPB\SEMESTER 2 IPB\Algoritma dan Struktur Data\Praktik\Praktikk_J0403251021_HaikalFaqihAufa\Pertemu
an 3> py latihan1.py
Sebelum dihapus:
10 -> 20 -> 30 -> 40 -> None
Masukan data yang ingin dihapus: 10
Data berhasil dihapus.
Sesudah dihapus:
20 -> 30 -> 40 -> None
```

2. LATIHAN 3

A. SS

```
#-----
# JUDUL LINKED LIST LAJINAH 3
# Studi Kasus: Implementasikan Pencarian pada node tertentu
# Double Linked List.
#
# Nama : Haikal Faqih Aufa
# NPM : 30403251021
# Kelas : TPL-R1
#-----

#-----
# Class untuk menyimpan data(isi nilai) dan next(Petunjuk untuk node
selanjutnya)
#-----
class Node:
    def __init__(self, data):
        self.data = data
        self.next = None
        self.prev = None

#-----
# Class untuk menyimpan fungsi aksi yang akan dilakukan pada linkedis
t
#-----
class DoublyLinkedList:
    def __init__(self):
        self.head = None

    #-----
    # Fungsi untuk menambah data ke belakang (akhir list)
    #-----
    def append(self, data):
        new_node = Node(data)

        #-----
        # Kalau list masih kosong
        #-----
        if self.head is None:
            self.head = new_node
            return

        #-----
        # Kalau sudah ada isinya, cari node terakhir dulu
        #-----
        temp = self.head
        while temp.next:
            temp = temp.next

        #-----
        # Sambungkan node terakhir ke node baru
        #-----
        temp.next = new_node
        new_node.prev = temp

    #-----
    # Fungsi untuk mencari data tertentu
    #-----
    def search(self, key):
        temp = self.head

        #-----
        # Cek satu-satu sampai habis
        #-----
        while temp:
            if temp.data == key:
                return True
            temp = temp.next

        return False

    #-----
    # Fungsi untuk menampilkan semua isi list
    #-----
    def display(self):
        temp = self.head
        while temp:
            print(temp.data, end=" ")
            temp = temp.next
        print()

#-----
# Membuat object dari DoublyLinkedList
#-----
dll = DoublyLinkedList()

#-----
# Input data dari user (dipisahkan pakai koma)
#-----
data_input = input(
    "Masukkan elemen ke dalam Doubly Linked List (pisahkan dengan koma): "
)
data_list = data_input.split(",")

#-----
# Memasukkan input menjadi list
#-----
for item in data_list:
    dll.append(int(item.strip()))

#-----
# Input data yang ingin dicari
#-----
cari = int(input("Masukkan elemen yang ingin dicari: "))

#-----
# Cek apakah datanya ada atau tidak
#-----
if dll.search(cari):
    print(f"Elemen {cari} ditemukan dalam Doubly Linked List.")
else:
    print(f"Elemen {cari} tidak ditemukan dalam Doubly Linked List.")
```

B. OUTPUT

```
PS E:\ARSIP AKADEMIK IPB\SEMESTER 2 IPB\Algoritma dan Struktur Data\Praktik\Praktikk_J0403251021_HaikalFaqihAufa\Pertemuan 3> py latihan3.py
Masukkan elemen ke dalam Doubly Linked List (pisahkan dengan koma): 9,8,7,6,5
Masukkan elemen yang ingin dicari: 9
Elemen 9 ditemukan dalam Doubly Linked List.
```

3. LATIHAN 5

A. SS

```
#####
# TUGAS LINKED LIST LATIHAN 3
# Studi Kasus: Implementasikan Pencarian pada node tertentu
# Double Linked List.
#
# Nama : Haikal Faqih Aufa
# NIM : 30403251021
# Kelas : TPL-B1
#####

# Class untuk menyimpan data(isi nilai) dan next(Pentunjuk untuk node selanjutnya)
#####
class Node:
    def __init__(self, data):
        self.data = data
        self.next = None

# Class untuk menyimpan fungsi aksi yang akan dilakukan pada linkedlist
#####
class LinkedList:
    def __init__(self):
        self.head = None

    # Fungsi untuk menambah data ke belakang (akhir list)
    # =====
    def append(self, data):
        new_node = Node(data)

        if not self.head:
            self.head = new_node
            return

        temp = self.head
        while temp.next:
            temp = temp.next

        temp.next = new_node

    # Fungsi untuk menampilkan semua isi list
    # =====
    def display(self):
        temp = self.head
        while temp:
            print(temp.data, end=" -> ")
            temp = temp.next
        print("null")

    # Fungsi untuk membalik isi list
    # =====
    def reverse(self):
        prev = None
        current = self.head

        while current:
            next_node = current.next
            current.next = prev
            prev = current
            current = next_node

        self.head = prev

# =====
# Membuat object linkedlist
# =====
ll = LinkedList()

# Input dari user (pisahkan pakai koma)
# =====
data_input = input("Masukkan elemen untuk Linked List: ")

# Ubah input jadi list
# =====
data_list = data_input.split(",")

# Masukkan satu per satu ke linked list
# =====
for item in data_list:
    ll.append(int(item.strip()))

print("Linked list sebelum dibalik: ", end="")
ll.display()

ll.reverse()

print("Linked list setelah dibalik: ", end="")
ll.display()
```

B. OUTPUT

```
PS E:\ARSIP AKADEMIK IPB\SEMESTER 2 IPB\Algoritma dan Struktur Data\Praktik\Praktikk_J0403251021_HaikalFaqihAufa\Pertemu
an 3> py latihan5.py
Masukkan elemen untuk Linked List: 1,2,3,4,5,6,7
Linked List sebelum dibalik: 1 -> 2 -> 3 -> 4 -> 5 -> 6 -> 7 -> null
Linked List setelah dibalik: 7 -> 6 -> 5 -> 4 -> 3 -> 2 -> 1 -> null
```