

Nama : Ahmad Fadillah

NIM :121140173

Kelas :RC

Link source-code replit:

<https://replit.com/@AhmadFadillah16/Praktikum-Algoritma-dan-Struktur-Data>

Source-code:

```
#include <iostream>
using namespace std;

// beri nilai variabel currYear (current year)

struct info // buat record yang menyimpan info dari SLLDataMobil
{
    string pabrik, namaMobil;
    int tahunProduksi, ccMobil;
};

struct SLLDataMobil // buat single linked list
{
    info inf;
    SLLDataMobil *next;
} * head // deklarasi awal node
,
*tail // deklarasi akhir node
;

class ListFungsi // deklarasi class yang menampung kumpulan fungsi-fungsi
{
public: // semua fungsi dimasukkan ke dalam public
    void insert(string pabrik, string namaMobil, int tahunProduksi, int ccMobil)
    {
        // fungsi penambahan node baru
        SLLDataMobil *newNode; // deklarasi node yang ingin ditambahkan
        if (head == NULL)
        { // dicek apakah awal node SLLDataMobil sudah memiliki isinya atau belum
            head = new SLLDataMobil;
            head->inf.pabrik = pabrik;
            head->inf.namaMobil = namaMobil;
            head->inf.tahunProduksi = tahunProduksi;
```

```

        head->inf.ccMobil = ccMobil;
        head->next = NULL; // karena baru memiliki satu buah node maka elemen
next dari head akan menunjuk ke NULL
        tail = head;
    }
    else
    { // jika sudah memiliki isi maka node yang baru tidak disimpan dikepala
tapi disimpan ke bagian buntut
        newNode = new SLLDataMobil;
        newNode->inf.pabrikan = pabrikan;
        newNode->inf.namaMobil = namaMobil;
        newNode->inf.tahunProduksi = tahunProduksi;
        newNode->inf.ccMobil = ccMobil;
        newNode->next = NULL; // karena ini dimasukan ke akhir node maka node
yang baru akan menunjuk ke NULL
        tail->next = newNode;
        tail = newNode;
    }
}

void sortData()
{
    // fungsi untuk menyorting data
    berdasarkan tahun produksi dari yang terkecil menuju ke terbesar
    SLLDataMobil *current, *index; // deklarasikan variabel current dan
variabel index agar bisa dibandingkan nilainya
    // deklarasikan
    int tempCC, tempThn;
    string tempPabrikan, tempNamaMobil;
    current = head; // masukan nilai current sebagai head karena dicek dari
awal

    // gunakan prinsip bubble sort dengan menggunakan nested loop
    while (current != NULL)
    {
        // titik akhiran current adalah akhiran node
        index = current->next; // masukan nilai index sebagai node setelah
current untuk pengecekan
        while (index != NULL)
        { // titik akhiran index adalah akhiran node
            if (current->inf.tahunProduksi > index->inf.tahunProduksi)
            { // apabila current tahun produksi lebih besar daripada index
tahun produksi maka ditukar seluruh nilai nodenya keduanya
                // assign penukaran nilai nya dengan menggunakan temporari
data

                tempThn = current->inf.tahunProduksi;
                tempCC = current->inf.ccMobil;
                tempPabrikan = current->inf.pabrikan;
                tempNamaMobil = current->inf.namaMobil;
            }
        }
    }
}

```

```

        current->inf.ccMobil = index->inf.ccMobil;
        current->inf.namaMobil = index->inf.namaMobil;
        current->inf.tahunProduksi = index->inf.tahunProduksi;
        current->inf.pabrikan = index->inf.pabrikan;
        index->inf.tahunProduksi = tempThn;
        index->inf.ccMobil = tempCC;
        index->inf.pabrikan = tempPabrikan;
        index->inf.namaMobil = tempNamaMobil;
    }
    index = index->next; // index dimajukan nilainya
}
current = current->next; // current dimajukan nilainya
}

void deleteFirst()
{
    // fungsi untuk delete bagian awal node yang akan
    dipakai di fungsi hapusAneh
    SLLDataMobil *temp; // deklarasikan nilai temp sebagai node yang ingin
    dihapus
    temp = head; // masukan masukan nilai head ke temp
    head = head->next; // pindahkan head ke depan
    delete temp; // maka baru bisa dihapus head sebelumnya
}

void deleteLast()
{
    // fungsi untuk delete bagian akhir node yang
    akan dipakai di fungsi hapusAneh
    SLLDataMobil *temp, *curr; // deklarasikan dua node, curr sebagai travers
    nilai ke node sebelum bagian akhir node
    curr = head;
    temp = tail;
    while (curr->next != tail) // selama curr bukan diakhir node maka curr
    akan terus travers
    {
        curr = curr->next;
    }
    curr->next = NULL; // ubah curr next yang tadinya menunjuk tail ubah ke
    NULL
    tail = curr; // dan masukan nilai curr ke tail agar tailnya bukan
    yang mau dihapus lagi
    delete temp; // baru bisa dihapus nilai tail yang ingin dihapus
}

void insertMid(string pabrikan, string namaMobil, int tahunProduksi, int
ccMobil)

```

```

{ // fungsi untuk menyelipkan
node ke tengah nodes
    SLLDataMobil *lastPtr, *midPtr, *newNode; // deklarasikan beberapa node
yang ingin di tranvers. lastPtr akan ditravers ke akhir node, midPtr akan
ditravers ke tengah node, newNode adalah node yang ingin diinputkan
    newNode = new SLLDataMobil; // alokasikan newNode
    // berikan nilai head ke lastPtr dan midPtr
    if (head == NULL)
    { // dicek apakah user langsung memilih opt no2 daripada opt no1, jika
iya maka akan mengisi di data head
        head = new SLLDataMobil;
        head->inf.pabrikan = pabrikan;
        head->inf.namaMobil = namaMobil;
        head->inf.ccMobil = ccMobil;
        head->inf.tahunProduksi = tahunProduksi;
        head->next = NULL;
        tail = head;
    }
    else
    {
        lastPtr = head;
        midPtr = head;

        while (lastPtr->next != NULL && lastPtr->next->next != NULL)
        { // perulangan akan terus terjadi
sampai lastPtr sampai ke akhir node
            lastPtr = lastPtr->next->next; // lastPtr dimajukan duakali
kedepan
            midPtr = midPtr->next; // lalu midPtr dimajukan satu kali
        }
        newNode->inf.pabrikan = pabrikan;
        newNode->inf.namaMobil = namaMobil;
        newNode->inf.tahunProduksi = tahunProduksi;
        newNode->inf.ccMobil = ccMobil;
        newNode->next = midPtr->next; // ubah nilai next newNode ke node
setelah midPtr
        midPtr->next = newNode; // ubah nilai next midPtr ke newNode
    }
}

void cariDataMobil(string namaMobil)
{ // fungsi untuk menampilkan data mobil sesuai
inputan user berdasarkan namaMobil
    SLLDataMobil *curr; // deklarasikan node untuk travers
    curr = head; // berikan nilai head ke curr sebagai awalan
    while (curr != NULL) // looping akan terus berjalan sampai akhir node

```

```

{
    if (curr->inf.namaMobil == namaMobil)
    { // jika curr namaMobil sesuai dengan yang dicari maka akan
      ditampilkan data dari mobil tersebut
        cout << "Pabrikan : " << curr->inf.pabrikan << "\n";
        cout << "Nama Mobil : " << curr->inf.namaMobil << "\n";
        cout << "Tahun Produksi : " << curr->inf.tahunProduksi << "\n";
        cout << "CC Mobil : " << curr->inf.ccMobil << "\n";
        break; // dan keluar dari perulangan
    }
    curr = curr->next;
    if (curr->inf.namaMobil != namaMobil && curr->next == NULL)
    { // jika tidak ada data yang sesuai maka masuk ke kondisi dibawah
        cout << "\nTidak ada nama mobil tersebut dalam data\n";
    }
}
}

void hapusAneh()
{
    // fungsi hapusAneh yang ada di tugas
    SLLDataMobil *curr; // deklarasikan node untuk travers node
    bool konfirmasiThn = false, konfirmasiCC = false;
    int rentangThn = 0, currYear = 2022; // nilai rentang tahun (currentYear
- curr->inf.tahunProduksi)
    curr = head; // masukan nilai head ke curr untuk
travers dari awal node
    while (curr != NULL) // looping akan terus terjadi sampai
curr di akhir node
    {
        rentangThn = currYear - curr->inf.tahunProduksi; // berikan nilai
rentangThn
        if (rentangThn > 20)
        { // jika rentangThn lebih besar daripada 20 maka konfirmasiThn akan
bernilai true
            konfirmasiThn = true;
        }
        if (curr->inf.ccMobil % 160 == 0)
        { // jika cc mobil merupakan kelipatan dari 160 maka konfirmasiCC
akan bernilai true
            konfirmasiCC = true;
        }
        curr = curr->next;
    }
    if (konfirmasiThn)
    { // jika konfirmasiThn bernilai true maka node awal akan dihapus
      deleteFirst();
    }
}

```

```

    }
    else
    {
        cout << "\n Tidak ada mobil umur 20 Tahun keatas \n";
    }
    if (konfirmasiAnehCC)
    { // jika konfirmasiAnehCC bernilai true maka node akhir akan dihapus
        deleteLast();
    }
    else
    {
        cout << "\n Tidak ada mobil yang cc nya kelipatan 160\n";
    }
}

void print()
{
    // fungsi untuk menampilkan semua node
    SLLDataMobil *curr; // deklarasi curr sebagai node untuk travers
    curr = head;        // berikan nilai head ke curr untuk travers dari awal
node
    while (curr != NULL)
    { // perulangan akan terus terjadi sampai curr berada diakhir node
        cout << "Pabrik : " << curr->inf.pabrik << "\n";
        cout << "Nama Mobil : " << curr->inf.namaMobil << "\n";
        cout << "Tahun Produksi : " << curr->inf.tahunProduksi << "\n";
        cout << "CC Mobil : " << curr->inf.ccMobil << "\n";
        curr = curr->next;
    }
}
} x; // deklarasi nama variabel classnya;

int main()
{
    char menu; // deklarasi variabel menu untuk navigasi pengisian
data
    bool checkSLL = false; // variabel untuk mengecek apakah user sudah mengisi
data atau belum, karena menu no 3,4,5 membutuhkan data terlebih dahulu
start: // deklarasi checkpoint untuk kembali ke menu utama
    cout << "SELAMAT DATANG DI MENU DATABASE MOBIL!\n";
    cout << "(1) Masukkan Data Mobil\n";
    cout << "(2) Masukkan Data Mobil Ditengah Single Linked List\n";
    cout << "(3) Pencarian Data Mobil\n";
    cout << "(4) Operasikan fungsi hapusAneh pada Data Mobil\n";
    cout << "(5) Tampilkan Data Mobil Yang Sudah Tersorting\n";
    cout << "(6) Keluar\n";

```

```

cin >> menu; // user menginputkan pilihan

string pabrik, namaMobil;
int tahunProduksi, CCMobil; // variabel-variabel yang digunakan untuk
dilempar lewat parameter

switch (menu) // memakai percabangan switch-case sebagai menu utama
{
case '1': // jika user menginputkan '1'
    cout << "Masukan Pabrik Mobil : ";
    cin.ignore(1); // gunakan cin.ignore karena saat memakai
getline, ini akan memasukan nilai ENTER juga ke dalam variabel maka digunakan
cin.ignore(1) maksudnya getline akan mengabaikan 1 character dibelakang inputan
yaitu ENTER
    getline(cin, pabrik); // memakai getline karena jika memakai cin biasa,
spasi nya tidak akan terbaca dan akan terjadi error.
    cout << "Masukan Nama Mobil : ";
    getline(cin, namaMobil); // sama kasusnya seperti diatas
    cout << "Masukan Tahun Produksi Mobil : ";
    cin >> tahunProduksi;
    cout << "Masukan CC Mobil : ";
    cin >> CCMobil;
    x.insert(pabrik, namaMobil, tahunProduksi, CCMobil);
    checkSLL = true; // berarti user telah setidaknya menginputkan 1 data
mobil
    goto start; // kembali ke checkpoint utama
case '2': // jika user menginputkan '2'
    cout << "Masukan Pabrik Mobil : ";
    cin.ignore(1); // kasusnya sama seperti diatas
    getline(cin, pabrik);
    cout << "Masukan Nama Mobil : ";
    getline(cin, namaMobil);
    cout << "Masukan Tahun Produksi Mobil : ";
    cin >> tahunProduksi;
    cout << "Masukan CC Mobil : ";
    cin >> CCMobil;
    cout << "Data Sebelum ditambah pada bagian tengah :\n";
    x.print();
    x.insertMid(pabrik, namaMobil, tahunProduksi, CCMobil);
    cout << "Data Setelah ditambah pada bagian tengah :\n";
    x.print();
    checkSLL = true; // berarti user telah setidaknya menginputkan 1 data
mobil
    goto start; // kembali ke checkpoint utama
case '3': // jika user menginputkan '3'

```

```

    if (checkSLL)
    {
        cin.ignore(1); // kasusnya sama seperti diatas
        getline(cin, namaMobil);
        x.cariDataMobil(namaMobil);
    }
    else
    {
        cout << "Masukan Data Terlebih dahulu!\n";
    }
    goto start; // kembali ke checkpoint utama
case '4':      // jika user menginputkan '4'
    if (checkSLL)
    {
        x.hapusAneh();
    }
    else
    {
        cout << "Masukan Data Terlebih dahulu!\n";
    }
    goto start; // kembali ke checkpoint utama
case '5':      // jika user menginputkan '5'
    if (checkSLL)
    {
        x.sortData();
        x.print();
    }
    else
    {
        cout << "Masukan Data Terlebih dahulu!\n";
    }
    goto start; // kembali ke checkpoint utama
case '6':      // jika user menginputkan '6'
    break;      // kenapa cuma ini yang memakai break? karena yang lain
                // memakai fitur goto yang mana akan langsung mengeluarkan program dari percabangan
                // ini dan langsung menuju ke checkpoint. dan case '6' ini merupakan jalan keluar
                // dan karena masih ada case default dibawahnya, makanya diberi break disini
default:       // jika user menginputkan selain 1-6
    cout << "Silahkan Masukan Angka Yang Sesuai Dengan Menu Diatas\n";
    goto start; // kembali ke checkpoint utama
}
cout << "TERIMA KASIH TELAH MAMPIR";
return 0;
}

```



## Dokumentasi hasil running Section Coba-coba:

```
File Edit Selection View Go Run Terminal Help
prak.cpp - prakmg1 - Visual Studio Code

PROBLEMS OUTPUT TERMINAL JUPYTER
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\ernik\Documents\algoritma struktur data\prakmg1> cd "c:\Users\ernik\Documents\algoritma struktur data\prakmg1"; if ($?) { g++ prak.cpp -o prak }; if ($?) { .\prak }
SELAMAT DATANG DI MENU DATABASE MOBIL!
(1) Masukan Data Mobil
(2) Masukan Data Mobil Ditengah Single Linked List
(3) Pencarian Data Mobil
(4) Operasikan fungsi hapusAneh pada Data Mobil
(5) Tampilkan Data Mobil Yang Sudah Tersorting
(6) Keluar
1
Masukan Pabrikkan Mobil : Mazda
Masukan Nama Mobil : MKS Miata
Masukan Tahun Produksi Mobil : 1995
Masukan CC Mobil : 1600
SELAMAT DATANG DI MENU DATABASE MOBIL!
(1) Masukan Data Mobil
(2) Masukan Data Mobil Ditengah Single Linked List
(3) Pencarian Data Mobil
(4) Operasikan fungsi hapusAneh pada Data Mobil
(5) Tampilkan Data Mobil Yang Sudah Tersorting
(6) Keluar
1
Masukan Pabrikkan Mobil : Lamborghini
Masukan Nama Mobil : Galardo
Masukan Tahun Produksi Mobil : 2014
Masukan CC Mobil : 2500
SELAMAT DATANG DI MENU DATABASE MOBIL!
(1) Masukan Data Mobil
(2) Masukan Data Mobil Ditengah Single Linked List
(3) Pencarian Data Mobil
(4) Operasikan fungsi hapusAneh pada Data Mobil
(5) Tampilkan Data Mobil Yang Sudah Tersorting
Ln 270, Col 2 Spaces: 4 UTF-8 CRLF C++ Go Live Win32 Prettier
25°C
Hujan akan berh...
```

```
File Edit Selection View Go Run Terminal Help
prak.cpp - prakmg1 - Visual Studio Code

PROBLEMS OUTPUT TERMINAL JUPYTER
1
Masukan Pabrikkan Mobil : Porche
Masukan Nama Mobil : Carenra GT
Masukan Tahun Produksi Mobil : 2019
Masukan CC Mobil : 3300
SELAMAT DATANG DI MENU DATABASE MOBIL!
(1) Masukan Data Mobil
(2) Masukan Data Mobil Ditengah Single Linked List
(3) Pencarian Data Mobil
(4) Operasikan fungsi hapusAneh pada Data Mobil
(5) Tampilkan Data Mobil Yang Sudah Tersorting
(6) Keluar
1
Masukan Pabrikkan Mobil : Honda
Masukan Nama Mobil : Supra GTR
Masukan Tahun Produksi Mobil : 2011
Masukan CC Mobil : 2200
SELAMAT DATANG DI MENU DATABASE MOBIL!
(1) Masukan Data Mobil
(2) Masukan Data Mobil Ditengah Single Linked List
(3) Pencarian Data Mobil
(4) Operasikan fungsi hapusAneh pada Data Mobil
(5) Tampilkan Data Mobil Yang Sudah Tersorting
(6) Keluar
1
Masukan Pabrikkan Mobil : BMW
Masukan Nama Mobil : BMW Razor
Masukan Tahun Produksi Mobil : 2004
Masukan CC Mobil : 3000
SELAMAT DATANG DI MENU DATABASE MOBIL!
(1) Masukan Data Mobil
(2) Masukan Data Mobil Ditengah Single Linked List
(3) Pencarian Data Mobil
(4) Operasikan fungsi hapusAneh pada Data Mobil
(5) Tampilkan Data Mobil Yang Sudah Tersorting
(6) Keluar
2
Ln 270, Col 2 Spaces: 4 UTF-8 CRLF C++ Go Live Win32 Prettier
25°C
Hujan akan berh...
```

```
File Edit Selection View Go Run Terminal Help
prak.cpp - prakmg1 - Visual Studio Code

PROBLEMS OUTPUT TERMINAL JUPYTER
2
Masukan Pabrikkan Mobil : Tesla
Masukan Nama Mobil : Tipe x
Masukan Tahun Produksi Mobil : 2018
Masukan CC Mobil : 2600
Data Sebelum ditambah pada bagian tengah :
Pabrikan : Mazda
Nama Mobil : MX5 Miata
Tahun Produksi : 1995
CC Mobil : 1600
Pabrikan : Lamborghini
Nama Mobil : Galardo
Tahun Produksi : 2014
CC Mobil : 2500
Pabrikan : Porche
Nama Mobil : Carrera GT
Tahun Produksi : 2019
CC Mobil : 3300
Pabrikan : Honda
Nama Mobil : Supra GTR
Tahun Produksi : 2011
CC Mobil : 2200
Pabrikan : BMW
Nama Mobil : BMW Razor
Tahun Produksi : 2004
CC Mobil : 3000
Data Setelah ditambah pada bagian tengah :
Pabrikan : Mazda
Nama Mobil : MX5 Miata
Tahun Produksi : 1995
CC Mobil : 1600
Pabrikan : Lamborghini
Nama Mobil : Galardo
Tahun Produksi : 2014
CC Mobil : 2500
Pabrikan : Porche
Nama Mobil : Carrera GT

Ln 270, Col 2 Spaces: 4 UTF-8 CRLF C++ Go Live Win32 Prettier
25°C Hujan akan berh...
```

```
File Edit Selection View Go Run Terminal Help
prak.cpp - prakmg1 - Visual Studio Code

PROBLEMS OUTPUT TERMINAL JUPYTER
Tahun Produksi : 2019
CC Mobil : 3300
Pabrikan : Tesla
Nama Mobil : Tipe x
Tahun Produksi : 2018
CC Mobil : 2600
Pabrikan : Honda
Nama Mobil : Supra GTR
Tahun Produksi : 2011
CC Mobil : 2200
Pabrikan : BMW
Nama Mobil : BMW Razor
Tahun Produksi : 2004
CC Mobil : 3000
SELAMAT DATANG DI MENU DATABASE MOBIL!
(1) Masukan Data Mobil
(2) Masukan Data Mobil Ditengah Single Linked List
(3) Pencarian Data Mobil
(4) Operasikan fungsi hapusAneh pada Data Mobil
(5) Tampilkan Data Mobil Yang Sudah Tersorting
(6) Keluar
3
MX5 Miata
Pabrikan : Mazda
Nama Mobil : MX5 Miata
Tahun Produksi : 1995
CC Mobil : 1600
SELAMAT DATANG DI MENU DATABASE MOBIL!
(1) Masukan Data Mobil
(2) Masukan Data Mobil Ditengah Single Linked List
(3) Pencarian Data Mobil
(4) Operasikan fungsi hapusAneh pada Data Mobil
(5) Tampilkan Data Mobil Yang Sudah Tersorting
(6) Keluar
5
Pabrikan : Mazda
Nama Mobil : MX5 Miata

Ln 270, Col 2 Spaces: 4 UTF-8 CRLF C++ Go Live Win32 Prettier
25°C Hujan akan berh...
```

```
File Edit Selection View Go Run Terminal Help      prak.cpp - prakmg1 - Visual Studio Code

PROBLEMS  OUTPUT  TERMINAL  JUPYTER
(2) Masukan Data Mobil Ditengah Single Linked List
(3) Pencarian Data Mobil
(4) Operasikan fungsi hapusAneh pada Data Mobil
(5) Tampilkan Data Mobil Yang Sudah Tersorting
(6) Keluar
5
Pabrikan : Mazda
Nama Mobil : MX5 Miata
Tahun Produksi : 1995
CC Mobil : 1600
Pabrikan : BMW
Nama Mobil : BMW Razor
Tahun Produksi : 2004
CC Mobil : 3800
Pabrikan : Honda
Nama Mobil : Supra GTR
Tahun Produksi : 2011
CC Mobil : 2200
Pabrikan : Lamborghini
Nama Mobil : Galardo
Tahun Produksi : 2014
CC Mobil : 2500
Pabrikan : Tesla
Nama Mobil : Tipe x
Tahun Produksi : 2018
CC Mobil : 2600
Pabrikan : Porsche
Nama Mobil : Canerra GT
Tahun Produksi : 2019
CC Mobil : 3300
SELAMAT DATANG DI MENU DATABASE MOBIL!
(1) Masukan Data Mobil
(2) Masukan Data Mobil Ditengah Single Linked List
(3) Pencarian Data Mobil
(4) Operasikan fungsi hapusAneh pada Data Mobil
(5) Tampilkan Data Mobil Yang Sudah Tersorting
(6) Keluar

Ln 270, Col 2  Spaces: 4  UTF-8  CRLF  C++  Go Live  Win32  Prettier  2314
25°C  Hujan akan berh...  02/10/2022
```

```
File Edit Selection View Go Run Terminal Help      prak.cpp - prakmg1 - Visual Studio Code

PROBLEMS  OUTPUT  TERMINAL  JUPYTER
SELAMAT DATANG DI MENU DATABASE MOBIL!
(1) Masukan Data Mobil
(2) Masukan Data Mobil Ditengah Single Linked List
(3) Pencarian Data Mobil
(4) Operasikan fungsi hapusAneh pada Data Mobil
(5) Tampilkan Data Mobil Yang Sudah Tersorting
(6) Keluar
4
(6) Keluar
5
Pabrikan : BMW
Nama Mobil : BMW Razor
Tahun Produksi : 2004
CC Mobil : 3800
Pabrikan : Honda
Nama Mobil : Supra GTR
Tahun Produksi : 2011
CC Mobil : 2200
Pabrikan : Lamborghini
Nama Mobil : Galardo
Tahun Produksi : 2014
CC Mobil : 2500
Pabrikan : Tesla
Nama Mobil : Tipe x
Tahun Produksi : 2018
CC Mobil : 2600
SELAMAT DATANG DI MENU DATABASE MOBIL!
(1) Masukan Data Mobil
(2) Masukan Data Mobil Ditengah Single Linked List
(3) Pencarian Data Mobil
(4) Operasikan fungsi hapusAneh pada Data Mobil
(5) Tampilkan Data Mobil Yang Sudah Tersorting
(6) Keluar
6
TERIMA KASIH TELAH MAMPIR
PS C:\Users\ernik\Documents\algoritma struktur data\prakmg1>

Ln 270, Col 2  Spaces: 4  UTF-8  CRLF  C++  Go Live  Win32  Prettier  2315
25°C  Hujan akan berh...  02/10/2022
```