

CONTOH SOURCE CODE STACK SEDERHANA

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

int main() {
    // create a stack of strings
    stack<string> makanan;

    // add element to the Stack
    makanan.push("Nasi Goreng");
    makanan.push("Nasi Rawon");
    makanan.push("Nasi Jagung");

    // print top element
    cout << makanan.top();

    return 0;
}
```

CONTOH SOURCE CODE PENAMBAHAN PADA STACK

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

int main() {

    // create a stack of strings
    stack<string> buah;

    // push elements into the stack
    buah.push("Anggur");
    buah.push("Jeruk");
    buah.push("Salak");
    buah.push("Leci");

    cout << "Stack: ";

    // print elements of stack
    while(!buah.empty()) {
        cout << buah.top() << ", ";
        buah.pop();
    }

    return 0;
}
```

CONTOH SOURCE CODE MENGHAPUS PADA STACK

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

// function prototype for display_stack utility
void display_stack(stack<string> st);

int main() {

    // create a stack of strings
    stack<string> colors;

    // push elements into the stack
    colors.push("Red");
    colors.push("Orange");
    colors.push("Blue");

    cout << "Initial Stack: ";
    // print elements of stack
    display_stack(colors);

    // removes "Blue" as it was inserted last
    colors.pop();

    cout << "Final Stack: ";

    // print elements of stack
    display_stack(colors);

    return 0;
}

// utility function to display stack elements
void display_stack(stack<string> st) {

    while(!st.empty()) {
        cout << st.top() << ", ";
        st.pop();
    }

    cout << endl;
}
```

CONTOH SOURCE CODE MENGAkses TOP ELEMENT PADA STACK

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

int main() {

    // create a stack of strings
    stack<string> minuman;

    // push element into the stack
    minuman.push("Es Teh");
    minuman.push("Es Degan");
    minuman.push("Es Campur");

    // get top element
    string top = minuman.top();

    cout << "Top Element: " << top;

    return 0;
}
```

CONTOH SOURCE CODE UNTUK MENGHITUNG JUMLAH ELEMENT PADA STACK

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

int main() {
    // Create a stack of strings called cars
    stack<string> Mobil;

    // Add elements to the stack
    Mobil.push("Pajero");
    Mobil.push("Inova");
    Mobil.push("Brio");
    Mobil.push("Ayla");
    Mobil.push("Agya");

    // Get the size of the stack
    cout << Mobil.size();
    return 0;
}
```

CONTOH SOURCE CODE UNTUK CEK STACK KOSONG/TIDAK

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

int main() {

    // create a stack of double
    stack<double> nums;

    cout << "Is the stack empty? ";

    // check if the stack is empty
    if (nums.empty()) {
        cout << "Yes" << endl;
    }
    else {
        cout << "No" << endl;
    }

    cout << "Pushing elements..." << endl;

    // push element into the stack
    nums.push(1);
    nums.push(2);

    cout << "Is the stack empty? ";

    // check if the stack is empty
    if (nums.empty()) {
        cout << "Yes";
    }
    else {
        cout << "No";
    }

    return 0;
}
```