```
Stack >
#anclude < costream>
# include < stack>
 stack (int > st; 11 Initializing a stack
stack method
 St. push(10); pushes an element in the stack
 st. pop(); pops the top element of the stack
 st.top(); access the top element;
 St. empty(); checks "y the stack is empty
 St. Size(); Gret the Stack Size.
Stack Implementation>
#include <bits/stdc++.h>
using namespace std;
class stack {
public:
      vector (int > stack_;
      stack () { };
      void bush (int n) {
          stack_.pushback(n);
      int pop() {
          int res = Stack_[Stack_.size-1];
```

```
stack_.pop_back();
          resturn res;
→ practice (8th, April, 2025)
# include < luts/stdc++.h>
 using namespace std;
 # include < lits/stdc++.h>
 using namespace std;
  class stack {
  pullic:
          vector(int> stack-;
          Stack ()
          void push (int n) {
             Stack_.push_back(n);
            int bob(){
              int res = stack_[stack_. size()-1];
             Stack_.pop-back();
            return res;
```

```
protice 5 heet 12/4/2025
# include < lits/stdc++.h>
using namespace std;
class stack {
 pullic:
        vector (int) stack;
       Stack (){
        void push (int n) {
           stack. bush_back(n);
         int pop() {
           int rus = stack[stack. size()-1];
           stack. pop-back();
         roturn res;
```