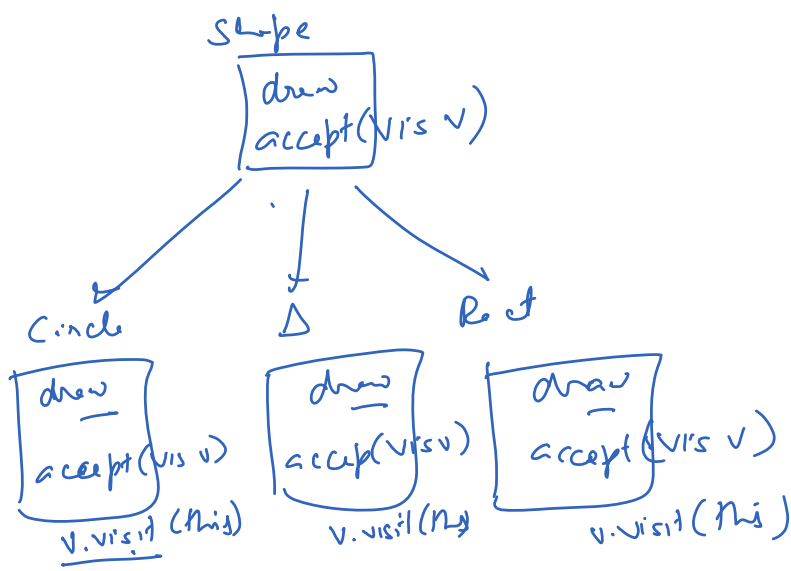
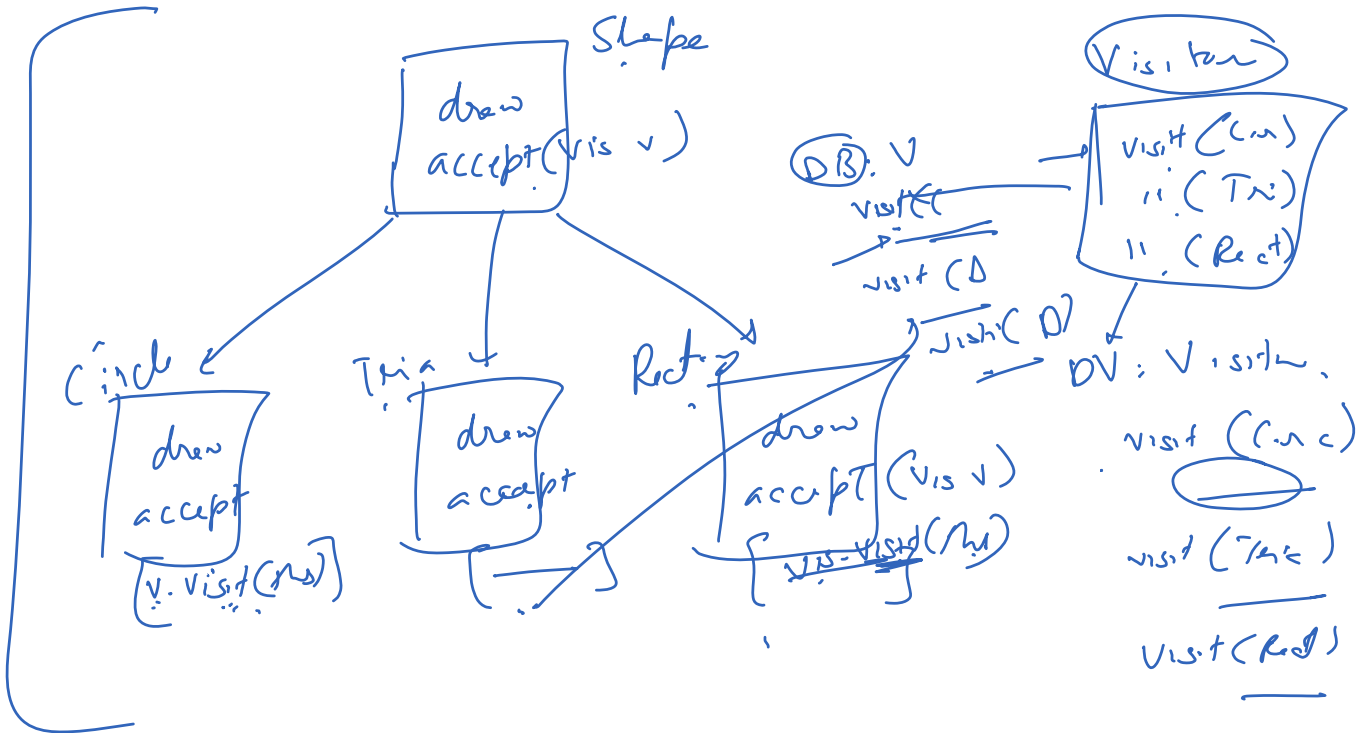


# SOLID

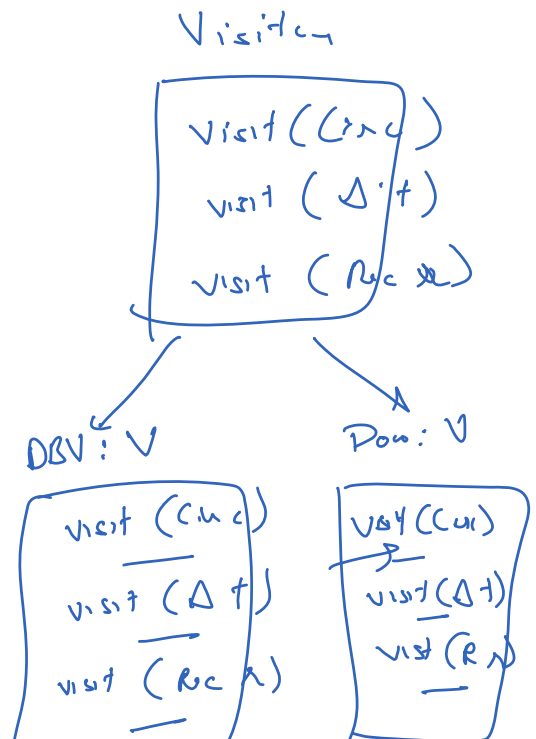
Visitor

style      open-closed

## Open-closed principle



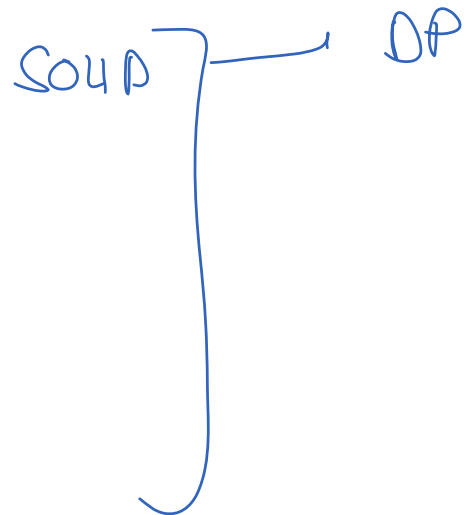
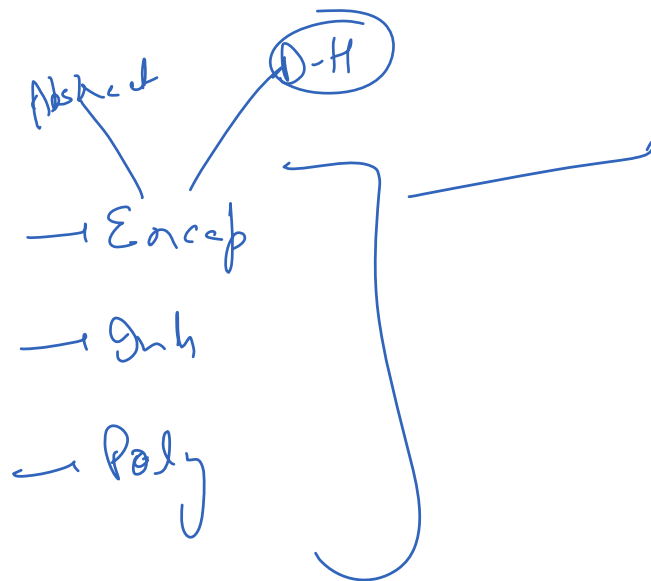
c.draw()  
c.accept(new DV\_V())

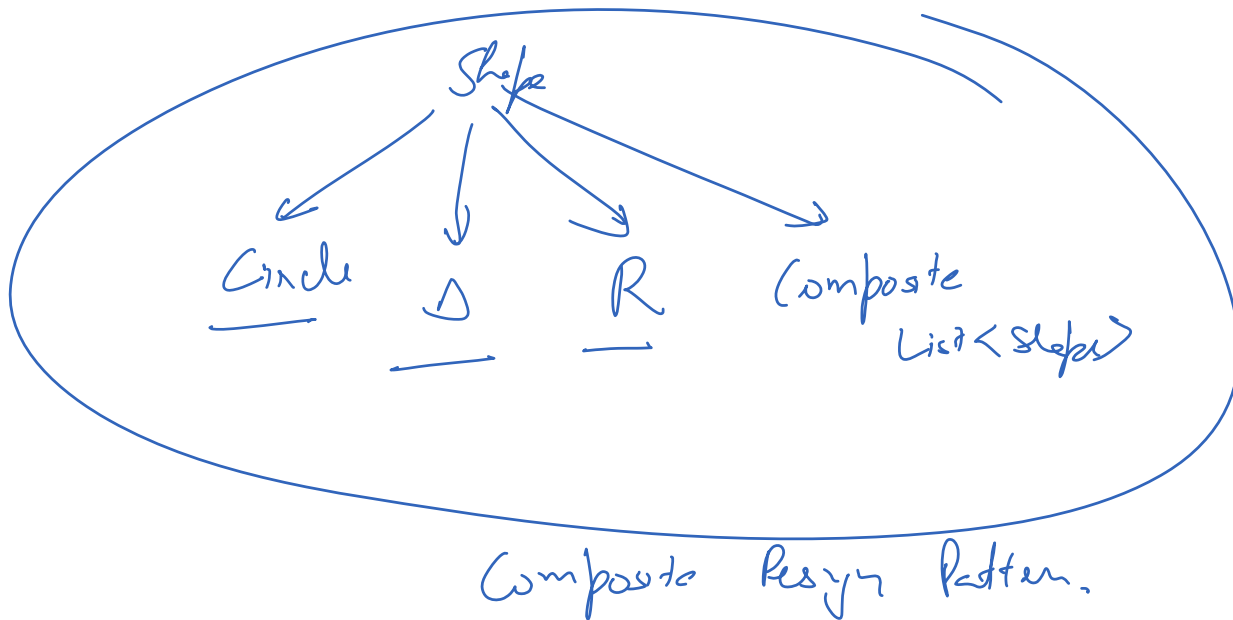


c: draw;  
c: accept ( new DonV( ) )

visit (Rec h)

vis (R, D)





9:03-9:08

#### 844. Backspace String Compare

Easy 5466 244 Add to List Share

Given two strings *s* and *t*, return *true* if they are equal when both are typed into empty text editors. '#' means a backspace character.

Note that after backspacing an empty text, the text will continue empty.

##### Example 1:

Input: *s* = "ab#c", *t* = "ad#c"

Output: true

Explanation: Both *s* and *t* become "ac".

##### Example 2:

Input: *s* = "ab##", *t* = "c#d#"

Output: true

Explanation: Both *s* and *t* become "".

✓✓✓✓  
a b # c

✓✓✓✓  
a d # c

✓✓✓✓  
s b # #  
(b)  
(x)

~~a~~  
~~b~~  
~~#~~

C =

C  
~~d~~  
#

✓✓✓✓  
(#) d (#)

(d)  
(#)

✓✓✓✓

✓✓✓✓

✓✓✓✓

✓ ✓ ✓ ✓ ✓  
~~a~~ b # # #

✓ ✓  
~~c~~ #

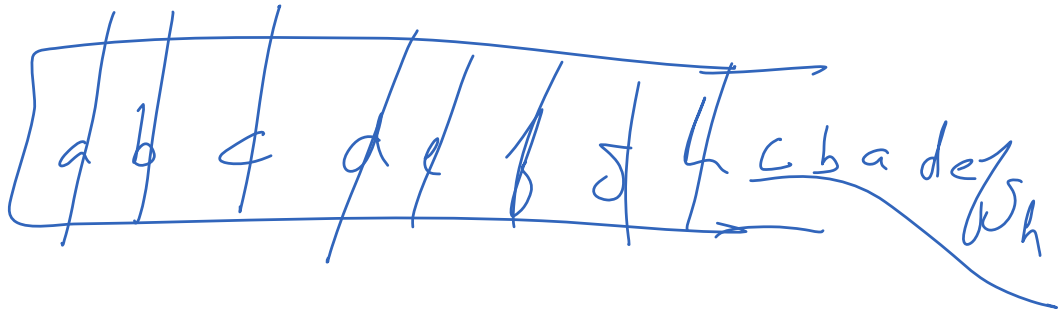
~~a~~

~~c~~

→ a b # c # d e # f g h # #

→ c # # # a d f

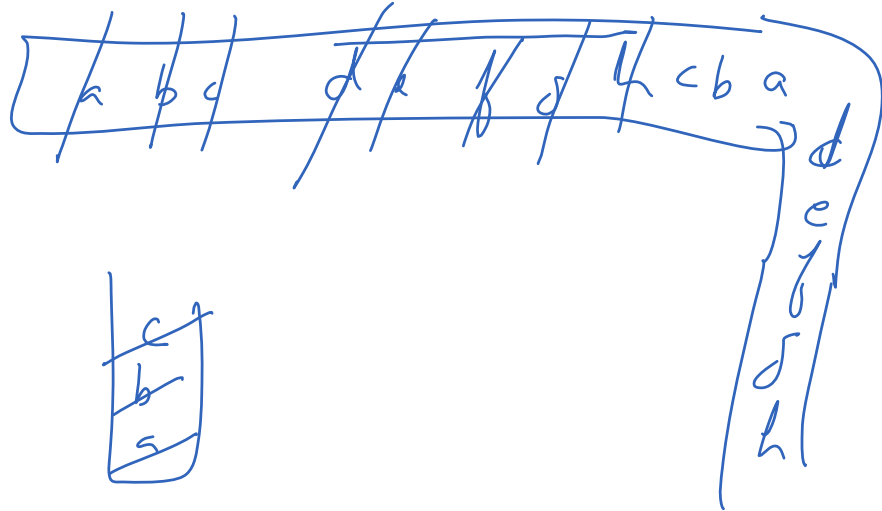
k = 3



~~c~~  
~~b~~  
~~a~~

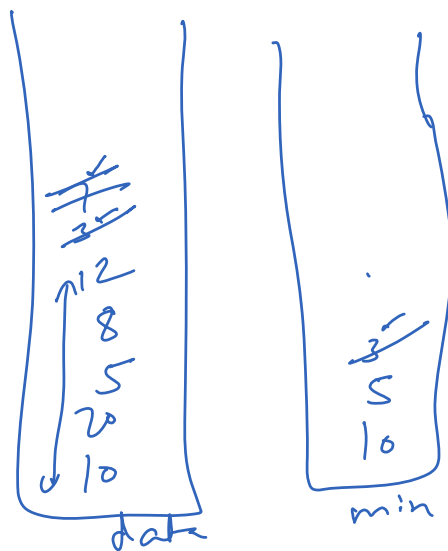
k = 3

```
// add code here.
int sz = q.size();
Stack<Integer> st = new Stack<>();
for(int i = 0; i < k; i++){
    st.push(q.remove());
}
while(st.size() > 0){
    q.add(st.pop());
}
for(int i = 0; i < sz - k; i++){
    q.add(q.remove());
}
return q;
```



9:35 - 9:45

10✓ 20✓ 5✓ 8✓ 12✓ 3✓ 7✓

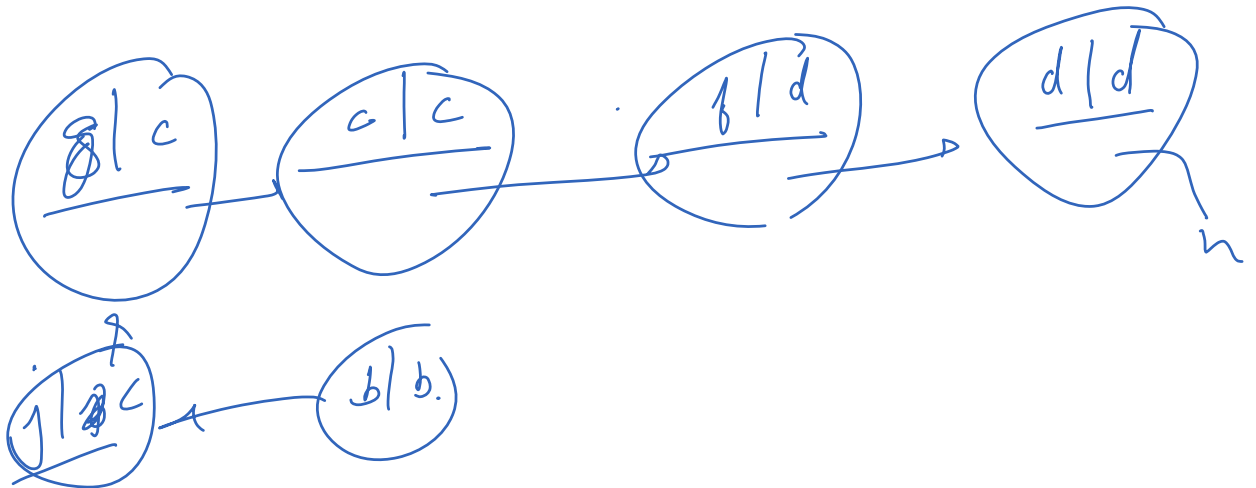


Push → data ✓  
min

Pop → data ✓  
min ✓

top → data  
min → min

✓ d ✓ f ✓ c ✓ g ✓ j ✓ b e a



✓ d ✓ c ✓ b ✓ e

~~A~~  $\left. \begin{array}{c} e + e - d \\ b \\ c \\ d. \end{array} \right\}$   $\begin{array}{c} e \\ \cancel{d} \end{array}$