

كلية الدراسات المتوسطة - الأزهر
College of Intermediate Studies-Alazhar



تراكيب البيانات

مدرس المساق / أ. م. رائد خضير

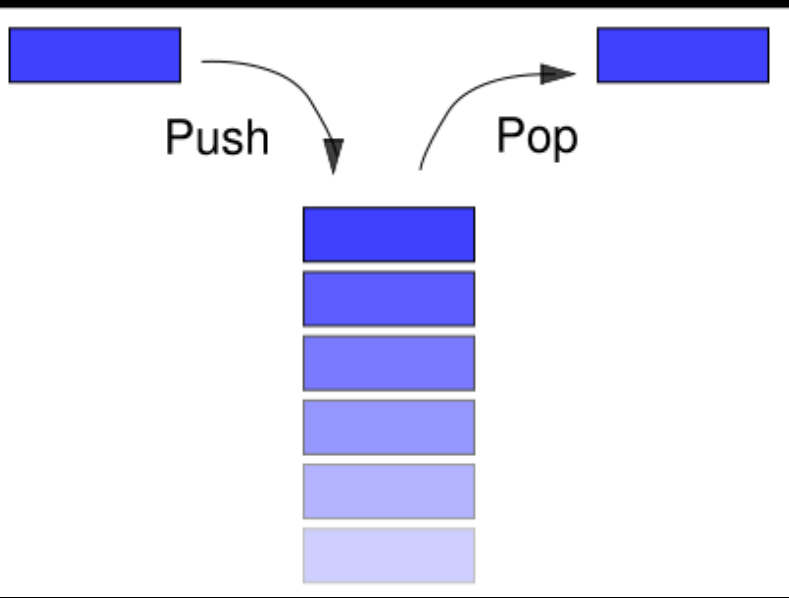
هيكل البيانات المكس (Stack)

محاور المحاضرة

- ما هو المكس ؟ (Stack)
- عملية الإضافة Push operation
- عملية الحذف Pop operation
- عرض بيانات عنصر ما

ما هو المكّس ؟ Stack

- المكّس هو أحد هياكل البيانات التي تسمح بالتفاعل معها (حذف و إضافة) من جهة واحدة فقط و هي تحاكي مفهوم (الرص الرأسية) مثل مجموعة من الكتب الرأسية أو مجموعة من الثمار كما يظهر في الصور أدناه.

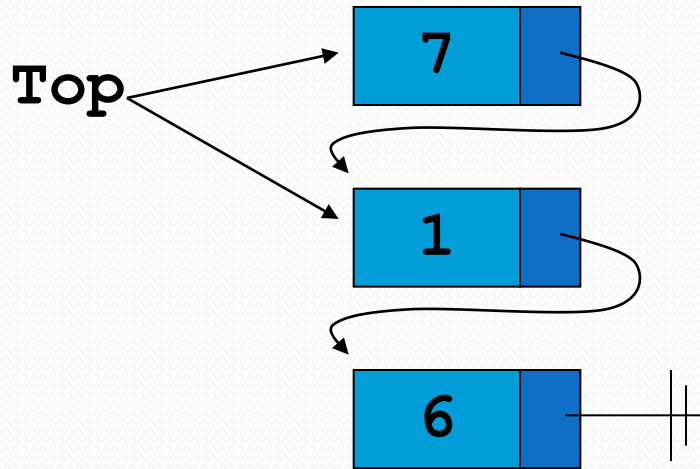


ما هو المكس ؟ Stack

- و إن كانت القائمة تسمح بالحذف و الإضافة في أي موضع و من أي جهة, فإنّ المكس لا يسمح إلا بثلاث عمليات أساسية فقط تتم من أعلى وهي: الإضافة Push و الحذف Pop العرض Peek.
- و تتعامل العمليات مع المكس بمبدأ **من يدخل أولاً يخرج أخيراً** FILO (First in Lirst Out)
- هذا لا يعني أننا لا نستطيع القيام بعمليات أخرى, و لكن أي عملية أخرى لابد أن تعتمد على هذه العمليات.

ما هي عناصر المكس؟

- يتكون المكس من عناصر مماثلة تماما لعناصر القائمة, و لا يوجد فروق أبدا بينهما سوى العمليات التي يسمح بالقيام بها على كلاهما.



```
class Items
{
    double Number;
}
```

عناصر Stack من الموظفين

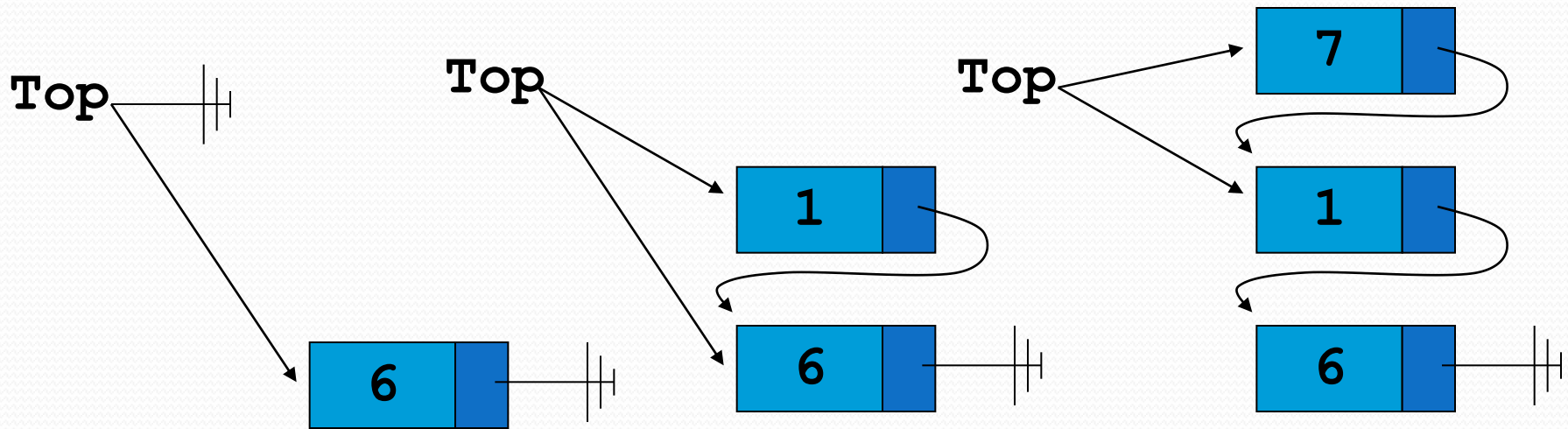
```
1. class Employee {  
2.     public int salary;  
3.     public String name;  
4.     public Employee next;  
5.     public Employee()  
6.     {  
7.         salary = 300;  
8.         name = "no name";  
9.     }  
10.    public Employee(int salary, String name)  
11.    {  
12.        this.salary = salary;  
13.        this.name = name;  
14.    } }
```

بناء Stack من الموظفين

```
1. class EmployeeStack
2. {
3.     Employee Top = null;
4.     int length = 0;
5.     // هنا تضاف العمليات المختلفة
6. }
```

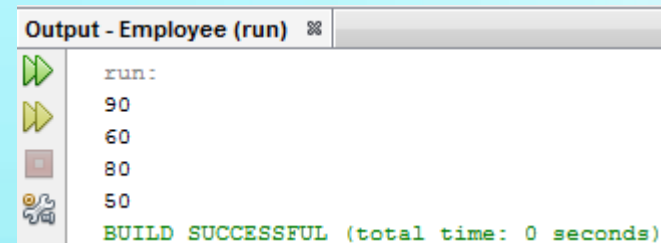
عملية الإضافة Push operation

- إضافة أي عنصر يتم من خلال التالي:
- جعل مؤشر العنصر الجديد يشير إلى ما يشير له المؤشر Top
- جعل المؤشر Top يشير إلى العنصر الجديد.



Push operation عملية الإضافة

```
1. public class Employee {
2.     private int [] rk = new int[4];
3.     private int top = -1;
4.     public void push (int y){
5.         if (top == 3)
6.             System.out.println("Stack is Full");
7.         else
8.             top ++;
9.         rk[top]=y;}
10.    public void pop (){
11.        if (top == -1)
12.            System.out.println("Stack is empty");
13.        else
14.            System.out.println(rk[top]);
15.        top --; }
16.    public static void main(String[] args) {
17.        Employee em = new Employee ();
18.        em.push(50);
19.        em.push(80);
20.        em.push(60);
21.        em.push(90);
22.        em.pop();
23.        em.pop();
24.        em.pop();
25.        em.pop()
26.    }}
```

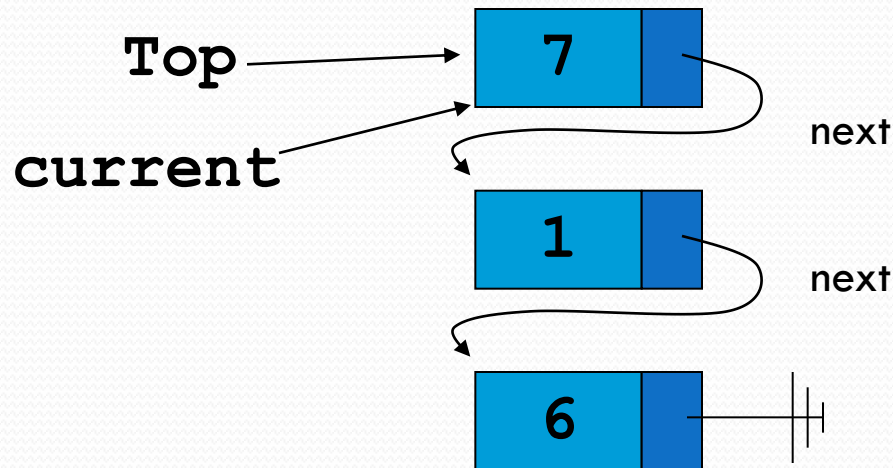


Output - Employee (run) ✖

```
run:
90
60
80
50
BUILD SUCCESSFUL (total time: 0 seconds)
```

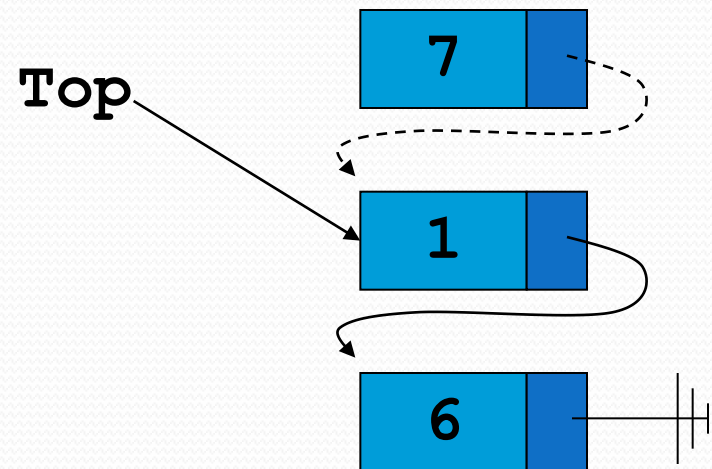
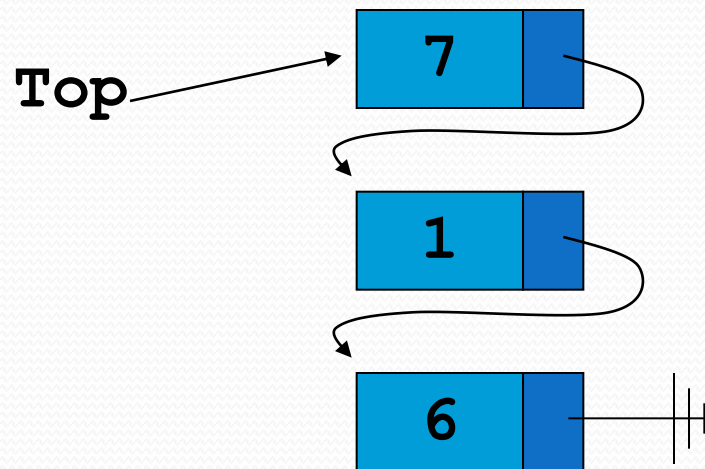
عملية طباعة عناصر ال Stack

- عملية الطباعة تتم من خلال التنقل على عناصر المكس و طباعة البيانات و هذا يتم من تعريف متغير Current يبدأ من حيث يشير المتغير Top.



عملية الحذف Pop operation

- حذف العنصر يتم من خلال التالي:
- جعل Top يشير إلى العنصر التالي للعنصر الأول.



عملية الاسترجاع مع الحذف Pop operation

```
1. public static void main(String[] args) {  
2.     Employee em = new Employee ();  
3.     em.push(50);  
4.     em.push(80);  
5.     em.push(60);  
6.     em.push(90) ;  
7.     em.pop();  
8. }
```

