

//

ArrayList : generic type (أي نوع) ~~Objects~~   
 ← ~~Objects~~   
 ~~Objects~~   
 ~~Objects~~

ArrayList<String> list = new ArrayList<> ();  
↓   
String   
↓   
String   
↓   
String

ex list.add("sadi");  
list.add("ahmed");  
list.add(1, "salma");  
↓   
index is to change

for (int i=0; i<list.size(); i++)  
print(list.get(i));

output

sadi  
salma  
ahmed



## ArrayList methods

```
ArrayList<String> list = new ArrayList ();
```

(1)

للاضافة

```
list.add("Hi");
```

```
list.add("Bye");
```

(2)

للاضافة عند index معين

← جعل Shift للـ index بعد

```
list.add(1, "Hello");
```

(3)

الحجم الكلي

```
int length = list.size();
```

تجميع القائمة الآن

```
for(int i=0; i<list.size(); i++)
```

```
println(list.get(i));
```

output

Hi

Hello

Bye

(4)

التالي، لاخذ الـ object الموجود في 'index' معين

```
list.get(i);
```

STUDENTS HUB.com

(5)

حذف الـ object الموجود في 'index' معين

```
list.remove("Hi");
```

(6)

حذف الـ object الموجود في 'index' معين

```
list.remove(i);
```



(7)

هل تحتوي على ...  
 --- print( list.contains("Hi"));  
 true

(8)

لتغيير العنصر في index معين  
 list.set(i, "Hello");

(9)

هل هما متاثلان؟

list.equals(object);  
 [يمكن استعمال طريقة انهل list1 تساي list2]  
 print( list.get(0).equals("Hi")); او  
 true

(10)

كيف استخامها؟ replaceAll()

(11)

كيف استخامها؟ to Array()  
 [في الصفحة التالية]



#

# Converting between Array / ArrayList

## 1/ From Array to ArrayList

```
String [] Words = {"one", "two", "three"};
```

```
ArrayList<String> list = new ArrayList<>(Arrays.asList(words));
```

## 2/ From ArrayList to Array

```
String [] Words = new String(list.size());
```

```
list.toArray(Words);
```

// Collections.sort/min/max/---  
needs to be comparable

ex Suppose our list contains, "ahmed", "Sadi", "helmi"

```
Collections.sort(list);
```

output →

ahmed  
helmi  
Sadi

max is "Sadi"  
min is "ahmed"

ex

Suppose our list contains "1", "3", "2"

```
Collections.sort(list);
```

output →

output →

1  
2  
3

max is "3" , min is "1"



(\*) كل نوع معين يكون له مكتوب له "بناء" على ماذا سوف يرتب  
 فمثلاً Strings ، سواء برأها احرف او ارقام  
 ← ترتب الجديا

Note كل الاعمال السابقة ، يتوجب على أي نوع من ال Objects

ex ArrayList < Person > list = new ArrayList<> ();

list.add(new Person (1, "Ali"));

list.add(new Person (5, "Rami"));

← لا يمكن استعمال sort/max... إلا لو حددت له على ماذا يرتب... (لا بد)

for(int i=0; i < list.size(); i++)

System.out.println(list.get(i));

output → if Person ~~contains~~ contains toString

→ ( id=1 name=Ali  
 id=5 name=Rami )

else

Prints an address

System.out.println(list.get(0).id);

STUDENTS-HUB.com

1

Capital الذي اول حرف Capital

ثم يرتب الذي اول حرف small

ex Ahmed → Rami → ahmed → rami

يمكن أن أعدد Person او subtype

Student , Worker على



تنبيه: (int, double) Simple type → ArrayList

الحل: ArrayList<Integer> list = new ArrayList<>();

list.add(new Integer("4"));  
list.add(new Integer(4));  
list.add(8);

Wrapper type

System.out.println(~~list.max()~~ Collections.sort(list));

→ 8

لكن هذا غير محبب استخدام.

الحل: Static method | override Jan, 2<sup>nd</sup> 2020