



C- Singly linked list

C - Doubly linked list

Leen Mohammed Alsaleh

LINKED LIST :

◆ Flexible size
حجم مرن

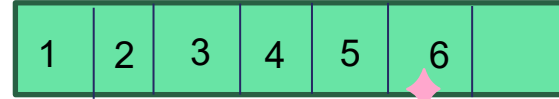
ARRAY VS LINKED LIST

ARRAY :
Fixed size
حجم ثابت

Add 7



1	2	3	4	5	6
---	---	---	---	---	---

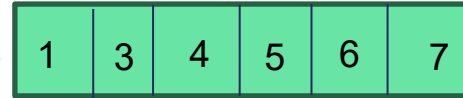


1	2	3	4	5	6	
---	---	---	---	---	---	--

Delete
number:
2



1	2	3	4	5	6
---	---	---	---	---	---



1	3	4	5	6	7
---	---	---	---	---	---

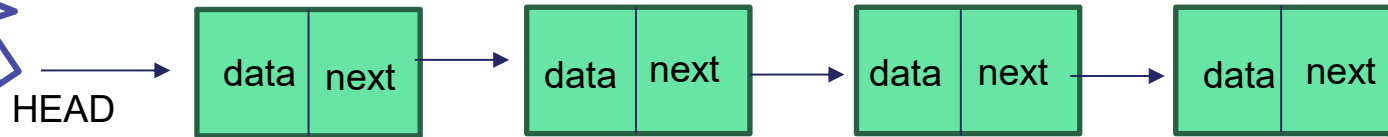
ARRAY VS LINKED LIST

LINKED LIST	ARRAY
Flexible size حجم مرن	Fixed size حجم ثابت
مو لازم تحجز أماكن جنب بعض	لازم تحجز أماكن جنب بعض
عيوب: الوصول لأي عنصر يكون في زمن غير ثابت وبطيئ	ميزة : توصل لأي عنصر يكون في زمن ثابت وسريع

{ 5 , 10 , 15 , 7 } ♦



كل مربع يحمل قيمتين :
• قيمة او البيانات (رقم او نص)
• مؤشر يشير الى النود التالي



{ 5 , 10 , 15 , 7 }

HEAD

5

10

15

7

Head
0003

data
5

next
0025

data
15

next
0042

data
10

next
0018

data
7

next
null

Singly Linked List

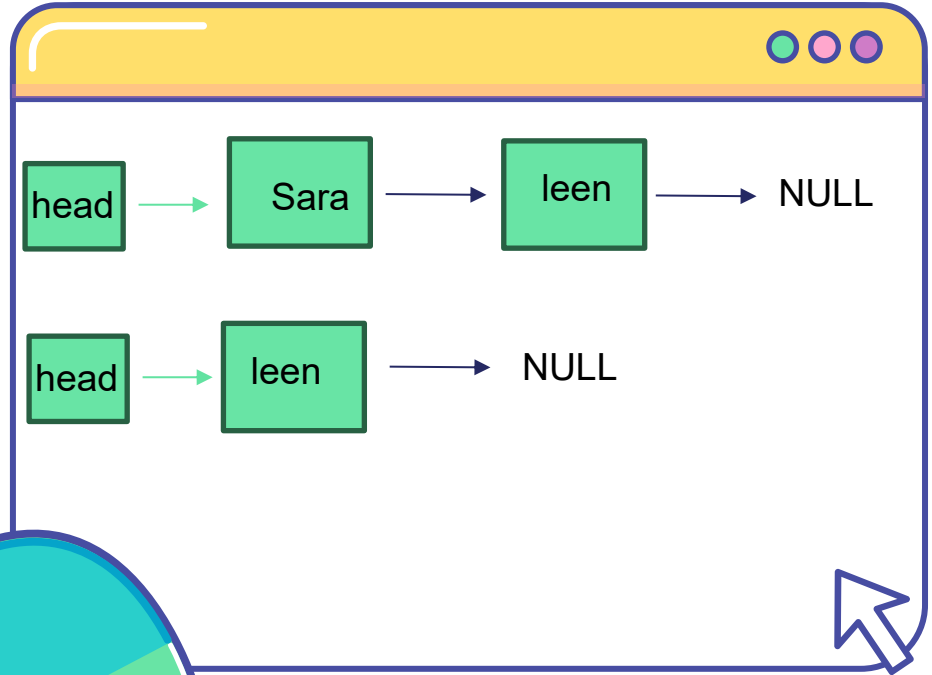
```
typedef struct list_s
{
    char *str;
    unsigned int len;
    struct list_s *next;
}
List_t;
```

```
list_t *node1 = malloc(sizeof(list_t));
node1->str = strdup("Leen");
node1->len = 4;
node1->next = NULL;
```

```
head = node1;
```

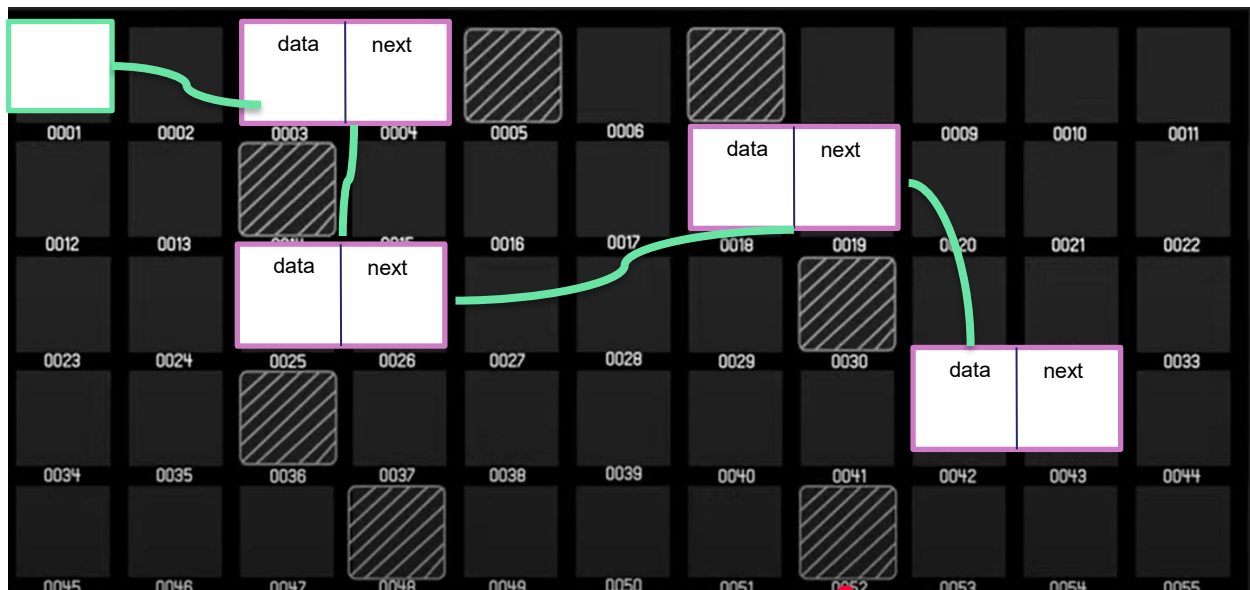
```
list_t *new = malloc(sizeof(list_t));
new->str = strdup("Sara");
new->len = 4;
new->next = &person1;
```

اقدر أقول بدال
node1->str
(*node1).str





{ 2 , 3 , 4 }



Double-Linked List



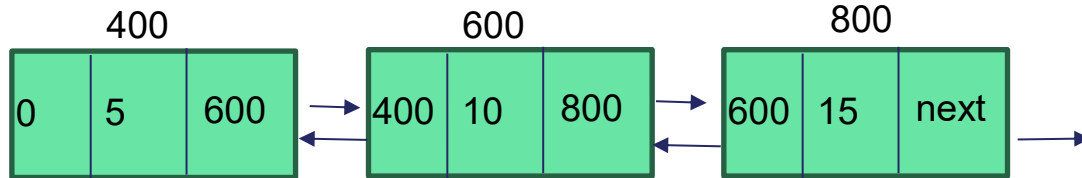
Linked List

Singly Linked List	Double Linked List
كل عنصر يعرف اللي بعده بس مايعرف مين قبله	كل عنصر يعرف مين اللي قبله ومين اللي بعده
[3] -> [7] -> [9] -> NULL	NULL <- [3] <-> [7] <-> [9] -> NULL

HEAD

400

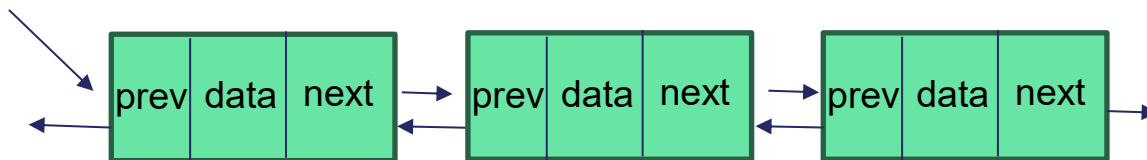
{ 5 , 10 , 15 }



كل مربع يحمل :

- قيمة او البيانات (رقم او نص)
- مؤشر يشير الى النود التالي
- مؤشر لنود اللي قبله

HEAD

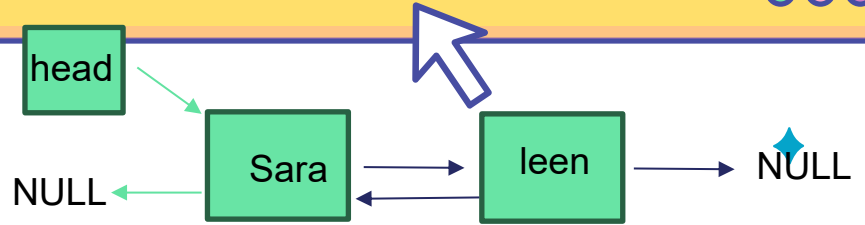


```
typedef struct dlistint_s
{
    char *str;
    struct dlistint_s *prev;
    struct dlistint_s *next;
} dlistint_t;
```

```
node1->str = strdup("Leen");
node1->len = 4;
node1->prev = NULL; // لأنه أول عنصر في القائمة، ماعندنا عنصر قبله
node1->next = NULL;
```

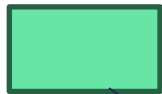
```
node2->str = strdup("Sara");
node2->len = 4;
node2->prev = NULL;
node2->next = node1;
```

```
// ربط النودين ببعض
node1->prev = node2;
head = node2;
```

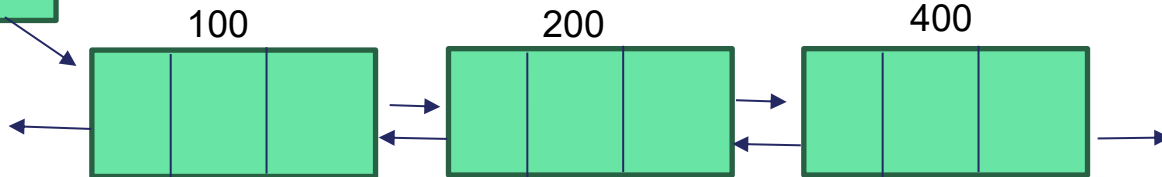




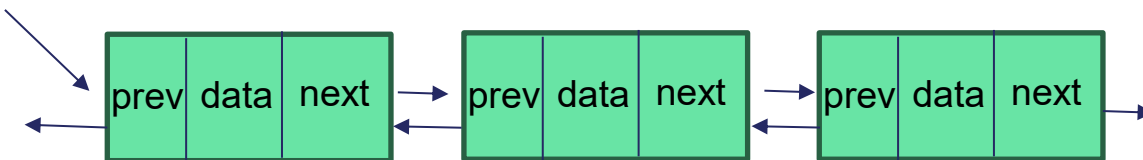
HEAD



{ 1 , 12 , 20 }



HEAD





**Thank
you**