

مقارنة بين أنواع القوائم المرتبطة (Linked Lists)

1. القائمة المرتبطة الأحادية (Singly Linked List)

- الاستخدامات: تُستخدم في بناء هيكل البيانات مثل "المكدس" (Stack) و"الطابور" (Queue)، وفي الأنظمة التي تتطلب مروراً في اتجاه واحد فقط لتوفير الذاكرة.
- الإيجابيات:
 - تستهلك مساحة ذاكرة أقل لأن كل عقدة تحتوي على مؤشر واحد فقط للعنوان التالي.
 - سهولة الإضافة والحذف من بداية القائمة.
- السلبيات:
 - لا يمكن المرور فيها إلا في اتجاه واحد (للأمام فقط).
 - صعوبة الوصول إلى العقدة السابقة (يجب البدء من البداية دائمًا).

2. القائمة المرتبطة الدائرية (Circular Linked List)

- الاستخدامات: تُستخدم في أنظمة التشغيل لجدولة العمليات (Round Robin Scheduling)، وفي تطبيقات الوسائط المتعددة (مثل تشغيل قائمة الأغاني بشكل متكرر).
- الإيجابيات:
 - يمكن الوصول إلى بداية القائمة من نهايتها مباشرة (حلقة مستمرة).
 - تسمح بالمرور على جميع العناصر بغض النظر عن نقطة البداية.
- السلبيات:
 - تعقيد أكبر في البرمجة لتجنب الدخول في "حلقة لا نهاية" (Infinite Loop).
 - صعوبة في تمثيل نهاية القائمة إذا لم يتم تتبع المؤشر بدقة.

3. القائمة المرتبطة المزدوجة (Doubly Linked List)

- الاستخدامات: تُستخدم في متصفحات الويب (للانتقال بين الصفحة السابقة والتالية) وفي البرامج التي تتطلب خاصية "التراجع" (Undo/Redo).
- الإيجابيات:
 - يمكن المرور في اتجاهين (للأمام والخلف).
 - الحذف أكثر سهولة وفعالية لأنك تملك عنوان العقدة السابقة مباشرة.
- السلبيات:
 - تستهلك ذاكرة أكبر (بسبب وجود مؤشرين في كل عقدة: **Prev** و **Next**).
 - تعقيد الكود البرمجي عند الإضافة أو الحذف لتحديث جميع الروابط.