

ببساطة الـ **time complexity** هي طريقة بنسخدمها عشان نقيس كفاءة الخوارزمية أو البرنامج من حيث الوقت اللي بيحتاجه عشان ينفذ. بنعبر عنها باستخدام رموز زي  $O(n)$  أو  $O(n^2)$  عشان نوضح إزاي الوقت بيزيد مع زيادة حجم البيانات اللي بنعالجها.

### أنواع الـ **Time Complexity**:

**$O(1)$** : الوقت ثابت، يعني مهما زاد حجم البيانات، الوقت مش هيزيد. مثال: الوصول لعناصر في مصفوفة باستخدام الفهرس.

**$O(n)$** : الوقت بيزيد بشكل مباشر مع زيادة حجم البيانات. مثال: البحث في مصفوفة غير مرتبة.

**$O(n^2)$** : الوقت بيزيد بشكل تربيعي مع زيادة حجم البيانات. مثال: خوارزمية الترتيب بالفقاعات (Bubble Sort).

**$O(\log n)$** : الوقت بيزيد بشكل أبطأ مع زيادة حجم البيانات. مثال: البحث الثنائي (Binary Search).

**$O(n \log n)$** : الوقت بيزيد بشكل أسرع من  $O(n)$  بس أبطأ من  $O(n^2)$ . مثال: خوارزمية الترتيب السريع (Quick Sort).