

◆ Class 1.

- Reference type (يعني بيتخزن في ال heap، والمتغير ييمسك في العنوان مش النسخة نفسها).
- يدعم الوراثة (Inheritance).
- مناسب للكائنات اللي فيها سلوكيات (methods) وحياة طويلة في البرنامج.
- استخدم class لما:
- الكائن كبير أو معقد.
- محتاج تورث منه classes ثانية.
- بيتغير حاله (state) بشكل مستمر.
- بتتعامل مع OOP concepts زي الوراثة و ال polymorphism.

◆ Struct 2.

- Value type (يعني بيتخزن في ال stack، والنسخة بتتنسخ بالكامل عند الإسناد).
- أخف وأسرع من ال class في الكائنات الصغيرة.
- مش بيدعم الوراثة (لكن بيدعم ال interfaces).
- يستخدم للـ data container البسيط.

• استخدم struct لما:

- عندك بيانات صغيرة الحجم (زي Point, Color, Date).
- مش محتاج تورث منه.
- محتاج سرعة وأداء أعلى (لأن القيمة بتتخزن مباشرة).
- الكائن immutable أو بيتغير نادرًا.

◆ 3. Record (C# 9+)

- زي ال class بس معمول للـ data models.
- بييجي افتراضياً بـ immutability (قيم ثابتة) ويعمل override للـ Equals () و GetHashCode () و ToString () تلقائياً.

- مناسب للـ DTOs / Models اللي أهم حاجة فيهم القيم مش الهوية.

• استخدم record لما:

- بتخزن بيانات ثابتة (immutable).
- عايز مقارنة بالقيمة value equality بدل المرجع reference equality.
- بتتعامل مع data transfer بين طبقات (مثال: API response, database entity).
- بتكتب أكواد models كتير وعايز تختصر (records بتوفر boilerplate code).

✓ ملخص سريع:

- Class: للكائنات المعقدة واللي ليها سلوك وحياة طويلة.
- Struct: للبيانات الصغيرة، value type، أسرع وأخف.
- Record: للـ data models اللي أهم حاجة فيهم القيم وثباتها.