حاليا بدور ع الفرق بين اني استخدم Dynamic و generic <T> في الفانكشن

فقابلني ان الـ Dynamic دي حاجه بتـ bypasses static type checking

و ده معناه انه مش بيتشك ع نوعه في ال compile-time لا هو بيشوفه فال run-time

الاختلافات ما بينهم

* **Type Safety:** Generics are **statically typed, ensuring type safety at compile-time**. Dynamics are **dynamically typed, leading to potential runtime errors** if the type is not handled properly.
* **Performance: Generics** offer **better** performance since the type information is resolved at compile-time. Dynamics incur a slight **performance overhead** as type information is resolved at runtime.
* **Compile-time vs. Runtime:** Generics resolve type information at compile-time, allowing for **early error detection**. Dynamics resolve type information at runtime, offering flexibility but sacrificing some safety.
* **Code Reusability:** Generics promote code reusability by creating flexible and strongly-typed components. Dynamics are more suitable for scenarios **where the type is unknown until runtime**, but they may lead to less reusable code.

1. Use Cases:

* **Generics:** Collections such as List<T>, Dictionary<T, U>, etc.Algorithm implementations that work with different data types.
* **Dynamics:** Interacting with dynamic languages or COM objects.Situations where the type is not known until runtime.