■ TypeScript Medium-Level Exercises

This document contains medium-level TypeScript exercises covering **Interfaces**, **Generics**, **Classes**, and **Nested Objects**. These exercises are designed for learners who have already covered the basics and want to practice applying concepts in more practical scenarios.

- **Exercise 1 – Interface & Array of Objects**
  Create an interface `Book` with fields `id`, `title`, and `author`. Then create an array of books and write a function that prints each book's details.
- **Exercise 2 – Nested Objects**
  Define an interface `User` that contains `id`, `name`, and a nested object `address` with `city` and `country`. Write a function to display a user's full information.
- **Exercise 3 – Posts & Comments**
  Create interfaces `Post` and `Comment`, where each post has multiple comments. Write a function that lists posts with their comments.
- **Exercise 4 – Class Blog Manager**
  Create a `Blog` class that allows adding posts, adding comments, deleting posts, and listing all posts with comments.
- **Exercise 5 – Generics with API Response**
  Create a generic type `ApiResponse` that wraps any type of data with `{ data: T, success: boolean }`. Use it with a `Post` and a `User`.
- **Exercise 6 – Updating Interfaces with Logic**
  Write a generic type that adds a new field `published: boolean` to any interface. Apply it to the `Post` interface.
- **Exercise 7 – Transform Interfaces**
  Using mapped types, create types that: (a) make all fields of `Post` optional, (b) make all fields readonly, (c) convert all fields into strings.

■ These exercises will help learners practice how to structure real-world applications using TypeScript's type system. Encourage them to write solutions in VSCode and run `tsc` to check for type errors.