TypeScript Week 2 Guided Project

Introduction:

You are hired as a junior developer at a wizard-themed fan portal. Your task is to build a small TypeScript application to manage and display information about magical items in the Wizarding World. The application should demonstrate your understanding of TypeScript's object-oriented features, including interfaces, functions, classes, generics, and the keyof utility. The goal is to create a fun and educational project that reinforces your knowledge of TypeScript through hands-on coding and logic building.

The application can:

1. Store magical items using objects and interfaces.
2. Use classes to manage item data and behavior.
3. Apply generic programming for reusable and type-safe containers.
4. Use functions to process and compare item properties.
5. Dynamically access properties using the keyof operator.

You will be graded based on how well you implement each of these features, as described below.

Problem Statement:

1. **Define Magical Item Structure:** Define a magical item using an interface IMagicalItem with properties: name (string), type (string), powerLevel (number), isRare (boolean).
2. **Use Classes and Methods:** Create a class MagicalItem that implements IMagicalItem. Include a method displayInfo() to log the item's details.
3. **Compare Items with a Function:** Create a function comparePower(item1, item2) that returns the name of the item with the higher power level.
4. **Generic Inventory Manager:** Create a generic class Inventory<T> with methods to add(item: T) and getAll() items from an internal array.
5. **Access Properties Using keyof:** Add a method getProperty<T, K extends keyof T>(item: T, key: K): T[K] to the Inventory class that returns the value of the given key for a given item.