TypeScript Week 1 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 character information from the *Wizarding World*. The application should demonstrate your understanding of **TypeScript fundamentals**, including **type annotations**, **enums**, **arrays**, **tuples**, **union types**, **readonly structures**, and **basic functions**.

The goal is to create a fun and educational project that reinforces your understanding of TypeScript through hands-on coding and logic building.

The application can:

1. Store characters with specific data types and structures.
2. Display and filter characters based on specific criteria.
3. Handle read-only and type-safe data using tuples and arrays.
4. Destructure tuples and arrays wherever appropriate.
5. Showcase the usage of optional and union types.

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

Problem Statement:

1. **Define Character Structure:** Define a character using type or interface, with properties: name (string), age (number), isWizard (boolean), house (enum), spells (string array), wand (tuple).
2. **Use Enums and Tuples:** Create a House enum and a Wand tuple with named types: [core: string, length: number, material: string].
3. **Add Character Function:** Implement a function addCharacter to store a new character in an array.
4. **Display Characters Function:** Write a function displayCharacters to iterate over the array and show each character's details including spell list and wand details.

5. **Filtering by House:** Implement a function filterByHouse(house: House) that returns all characters from a given house.
6. **Count Wizards and Muggles:** Implement a function to count and return the number of wizards vs. muggles in the list.