Task 01: Retrieve Basic Character Information

Objective

In this task, you will retrieve and display basic information about your D&D character using the API functions. This task will introduce you to Rust syntax for function calls and printing values to the console.

Instructions

1. **Call the API Functions**: Use the functions provided in the D&D Character Simulator API to retrieve character details. 2. **Store Values in Variables**: Assign the retrieved values to variables so that you can use them later in your code. 3. **Print the Character Information**: Display the character's name, class, ancestry, and a couple of attributes in the terminal.

Example Code

Here's some example code to help you get started. This code retrieves and prints the character's name, class, ancestry, and a few attributes.

```
fn main() {
    let name = get_character_name();
    let class = get_character_class();
    let ancestry = get_character_ancestry();
    let strength = get_strength();
    let intelligence = get_intelligence();

    println!("Character Name: {}", name);
    println!("Class: {}", class);
    println!("Ancestry: {}", ancestry);
    println!("Strength: {}", strength);
    println!("Intelligence: {}", intelligence);
}
```

Running the Code

To test your code, make sure you're in the correct project directory. Run the following command in the terminal to see the output:

```
cargo run
```

You should see output similar to:

```
Character Name: Arin
Class: Fighter
Ancestry: Elf
Strength: 16
Intelligence: 14
```

Additional Exercise

After successfully displaying the basic information, try the following:

- Retrieve and print additional attributes, such as `dexterity` and `charisma`.
- Write a function to calculate and print the modifier for each attribute.

Next Steps

This task introduces basic function calls and printing in Rust. Once you're comfortable with these steps, you'll be ready to explore more advanced interactions with the D&D Character API.