Mars Lapierre-Furtado

Project 1 - Guess the Air!

For this project, I created a minigame called *Guess the Air!*, where the goal is to guess which city has cleaner air between two options. The data comes from the World Air Quality Index API, so every round uses real, live information. The idea was to make something that felt fun and interactive while still focusing on Python and practicing how to use APIs.

When the program starts, players are greeted with a small intro screen and prompted to select a difficulty level. The difficulty determines how many cities are in the pool that the game randomly pulls from. *Easy* has 9 cities, *Medium* has 23, and *Hard* has 90. Once the difficulty is chosen, the game displays two random cities, and the player has to pick which one they think has cleaner air.

If the player is correct, they earn a point. If they're wrong, they lose a life. Everyone starts with three lives, and the program keeps track of both score and remaining lives. The output uses a mix of text colors, emojis, and ASCII-style formatting to make the terminal interface feel a bit more lively.

At the end of each round, the program displays which city was cleaner and shows your updated score. Once you lose all your lives, the game shows a final results screen that includes your total score, your accuracy percentage, and a little message that changes depending on how well you did. I added the accuracy section to make it more satisfying for players to see their performance and maybe want to replay the game to improve their score.

The main goal for *Guess the Air!* was to explore how data and code can be used in a playful, expressive way, even inside something as minimal as the terminal. I wanted to make it interactive and fun, while still practicing technical concepts like working with live data, using functions, and handling user input.

At first, though, I actually made a completely different version by accident. I turned it into a full website. I got carried away adding animations, visuals, and sound effects before realizing everything was supposed to happen directly in the terminal. Even though it wasn't the right format, I think making both versions helped me understand how presentation affects interaction. The web version focused more on visuals and effects, while the terminal version pushed me to make the experience engaging through text alone, which was a really fun challenge.

Overall, this project was about finding creative ways to make something simple feel interactive and alive. Once the main logic was in place, I enjoyed adding all the little text effects and details that make the game feel more personal and playful.