

Pokedex Guesser

By: Louis Garcia

Project Description

A game created with Python that displays a pokedex entry from Pokemon Scarlet / Violet and asks the user to provide the name of the Pokemon that has that description.

Users

Users	Descriptions	Objectives
Pokemon Enthusiast	A an avid Pokemon fan who will play this game to test their knowledge.	To play the game without any issue.
		To receive feedback whenever their answer is correct or not.
		To have their answer accepted properly.
		To know the points they currently have.
		To know the questions they got wrong.
		To know the questions they got correct.

User Journey

1. User starts the program.
2. User sees the main menu.
 - a. User chooses the Exit option
 - i. The program counts down from 3 then closes itself.
 - b. User chooses the Instructions option
 - i. The program explains the game.

- ii. The program goes back to the main menu.
- c. User chooses an option not available in the system.
 - i. The program prompts the user to choose a valid option.
 - ii. The program goes back to the main menu.
- d. User chooses the Start Game option.
 - i. The program initiates the score counter.
 - ii. The program initiates the question counter.
 - iii. The program splashes “Are you ready?”
 - iv. The program gets a random pokédex entry from a dictionary.
 - 1. The program checks whether the question has already been asked.
 - 2. If it has been asked already, the system will draw another question.
 - 3. If it has not been asked yet, it will be removed from the list of questions.
 - v. The program prompts the user to provide an answer.
 - vi. The program evaluates the answer provided.
 - vii. If the answer is correct:
 - 1. The user is notified their answer is correct.
 - 2. The user is provided some trivia.
 - 3. A point gets added to the score counter.
 - 4. The question counter increases by 1.
 - 5. The program checks whether there have been 20 questions already.
 - a. If there are already 20 questions, the ending phase is initiated.
 - b. If there are less than 20 questions, the next question is picked.
 - viii. If the answer is wrong.
 - 1. The user is notified their answer is wrong.
 - 2. The user is provided the correct answer.
 - 3. The score counter remains the same.
 - 4. The question counter increases by 1.
 - 5. The program checks whether there have been 20 questions already.
 - a. If there are already 20 questions, the ending phase is initiated.

- b. If there are less than 20 questions, the next question is picked.
- ix. User reaches the ending phase.
- x. The program notifies the user their score.
- xi. The program checks the user score in a range of values and returns the remarks for that value.
- xii. The program asks the user if they want to play again.
 - 1. If the user chooses to play again, the game restarts.
 - 2. If the user chooses not to play, the game starts from the main menu.