**1. Project Overview - Guess the Dog Breed**

**This Java project generates a random dog URL from the** [**Dog API**](https://dog.ceo/dog-api/?ref=public_apis&utm_medium=website)**, and challenges the user to guess the breed of the dog. They are given 5 guesses. It then checks the user's answer with the actual breed. When the user reaches one guess left, they are provided with a hint. After the round is completed, they are given their winrate and given an option to replay.**

**2. Code Overview**

**main(String[] args)**

* **Used to start the program**
* **Invokes the App class to run**

**getURL()**

* **Returns the URL of the generated dog image**
* **Used in getting the dog breed**

**getHint()**

* **Returns the hint created from the dog breed provided**
* **Used in the handleGuess method**

**App()**

* **Sets up the GUI features, such as JLabel (where the image is displayed), JPanel (Where the GUI dimensions are created), JTextField (Where the user guesses), and JButton (Where the user can click to replay).**
* **Allows for the user to be able to play outside the terminal.**
* **Used in order for the program to run.**

**startNewRound()**

* **Starts the round for the user**
* **Gets a random dog image**
* **Uses ImageIO in order to display the image in the GUI**
* **Updates the UI after every guess, including hints, number of guesses, and the guess entered**

**handleGuess(ActionEvent e)**

* **Used in the startNewRound() method**
* **Sets the guess to lowercase**
* **If the guess is correct, increments win by 1. If the guess is wrong, allows the user to retry and keep going until the guesses run out**
* **At the last guess, the user is provided with a hint**
* **When they run out of guesses, they lose**

**showResult(boolean won)**

* **Used in the handleGuess method**
* **Displays if the user won or not based on the boolean provided**
* **Have a popup showing if they won or lost and asked them if they want to play again**
* **Updates the statistics of the user's winrate**

**getRandomDog(String imageURL)**

* **Gets the dog breed from the URL itself**
* **Filters out the type of the dog breed(Australian, Polish, etc)**
* **Returns the dog breed**
* **Used in the parameter of getHint()**

**3. Features Implemented (Rubric Aligned)**

**Base Project (88%)**

* **Uses an external API (Dog API)**
* **Uses multiple Java methods and logic**
* **Parses JSON response using basic string matching**
* **Provides interactive user experience via GUI and dialog**

**Statistics/ Machine Learning/Basic Computation (+6%)**

* **Calculates the users average winrate while playing the game**

**GUI (Swing) (+2%)**

* **JPanel creates the dimensions of the GUI**
* **JLabel creates the area that the image is displayed**
* **JTextField creates the area the user can guess in**
* **JButton creates the buttons the user can click**

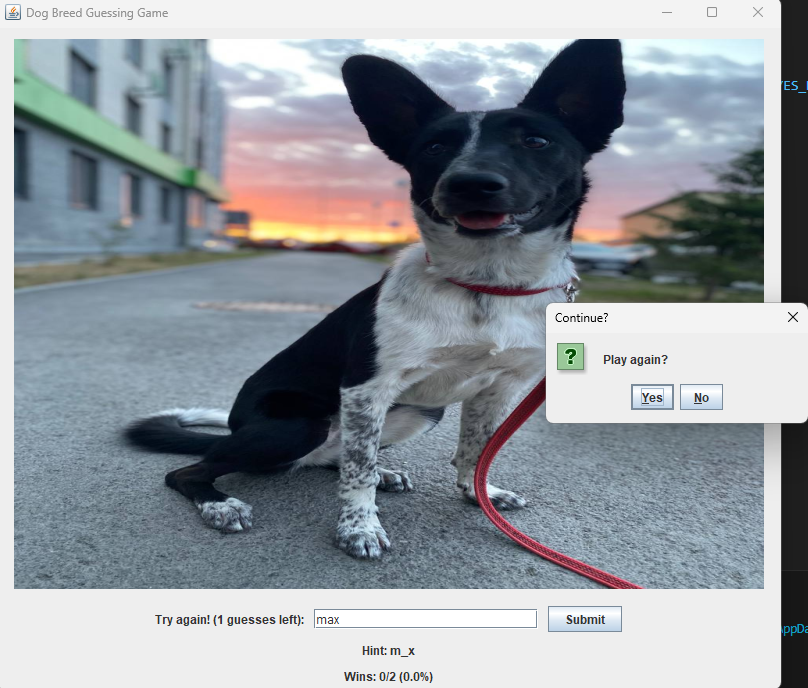
**Save/Load Data (+2%)**

* **The JSONObjectProcessing gets the data from the API and stores it**

**Filter/Sort Data (+2%)**

* **Filters out the type of dog breed it is in the getRandomDog method**

**4. Output Example**

****

**5. What I Learned**

* **How to connect a public API in java using API.getData()**
* **How to parse JSON using string methods**
* **How to use Java Swing for displaying images, handling input, and creating an interactive GUI**
* **How to handle exceptions and utilize try/catch**