

Celebrity Dogs Game

A program needs to be created that allows the user to play Celebrity Dogs, a round based card comparison game (a bit like Top Trumps).

There is a pack of cards with each card in the pack representing a famous dog. The following details about the dog appear on the card:

- the name of the dog
- a value between 1 and 10 for the fitness score of the dog, with 1 the lowest and 10 the highest
- a value between 1 and 10 for the intelligence of the dog, with 1 the lowest and 10 the highest
- a value between 1 and 10 for the friendliness of the dog, with 1 the least friendly and 10 the most friendly
- a value between 1 and 10 for the "drool rating" of the dog, with 1 the least amount of drool and 10 the most amount of drool

How the game works:

The game is played against the computer.

The player is asked to enter the number of cards in the pack. The whole pack is then divided between the player and the computer, creating two stacks.

The player is shown the first card on their stack. They choose one of the categories: fitness, friendliness, intelligence or drool.

The computer's card is then revealed and the value for the category chosen compared with the value on the player's card. The highest value wins. The two cards are then placed at the bottom of the winners stack.

The game continues until one of the player or computer have no cards.

Resources:

You have been provided with:

- a text file called **dogsdata.txt** (there are 30 dogs in the pack)
- a program which will read in the data file for you and create a 2-dimensional array for you. This needs to be copied into IDLE and the file saved

IMPORTANT: these files *MUST* be saved in the same folder

Things to do to start with

1. run the program to see what it produces
2. comment out the `printList(dogsList)` line of the program
3. begin to write your program

The program should work in this way:

1. A title is displayed and the player is asked their name. The program should then greet them by name
2. A menu is then displayed allowing the user to select from the following options:
 - play game
 - see a random dog card
 - quit
3. If the user selects the quit option then a suitable message should be displayed and the program should stop
4. If the user selects "see a random dog card" then a random card should be displayed to the user. The program should then return to the menu
5. If the user selects "play game" they are asked to enter the number of cards to be used in the pack. The number should be between 4 and 30.
 - if the number is not between 4 and 30 an error message should be displayed
 - if an odd number is entered an error message should be displayed
 - in either case the user is returned to the menu (step 2)
6. If a valid number is entered at step 5, the cards are divided into two equal stacks. You could shuffle the cards first if you want to
7. The game should then be played

Your program should make use of subroutines for each of the main steps.