About Cards Against Humanity

Note: This is a party game can get pretty crude/offensive. However, it's a lot fun.

Cards Against Humanity is a card Mad Libs based party game. Every person starts off with a deck of 5 response cards where each card has a word or a phrase(ie: "natural selection", "The petty troubles of the landed gentry", "A disappointing birthday party.", etc.). One person starts off as the Card Czar and picks a prompt card. This prompt card has a sentence with a fill in the blank or some type of question(ie: "Due to a PR fiasco, Walmart no longer offers _____."). The other players put down a response card that fills in the blank of the prompt card. The Card Czar choose his favorite(most often the funniest) response card that matches the prompt and whoever's card that was gets a point. Then the Card Czar position rotates. This continues until a person gets a certain number of points(you decide how many) and wins.

How to Run Program:

I wrote this program in **Python 3**. Please run this program where **Python 3** is installed.

Enter the CardsAgainstHumanity directory and run the "main.py" file. I usually run it by typing "python main.py" into the terminal. Then follow the instructions for the game setup that appear on the terminal.

Game Steps:

The terminal will ask you how many players you want to add, how many points you want to play till, and the names of the players.

The game will start by displaying the scoreboard and selecting the Card Czar by stating "Card Czar Rotation". A player will be listed as the Card Czar and it will show the prompt sentence for the round.

The game will indicate which player is next to select his/her response cards. That player should come to the terminal and hit "Enter". The terminal will display the player's deck of cards and he/she will choose a response that is fitting for the prompt. After the card is put down the player automatically picks up a new random card from the response card deck. The terminal will state which player is up next. This continues till all the players have put down their cards.

The Card Czar then decides which response card that was put down is his favorite. This step occurs when the terminal states "Card Czar Selection". Once he selects his favorite response card, a point is given to the one who put down that card. The score is updated and a new round begins with a new Card Czar. This continues until someone reaches the points needed to win.

Implementation:

The Cards:

I found an excel sheet on Reddit with all the Cards Against Humanity cards. I parsed that excel sheet using pandas to get all the prompt cards and response cards. I then saved them to binary files and .txt files(in case you wanted to see them). The excel sheet, python file, and card files are in the "DataClean" directory.

Game Setup:

For the initial setup of the game (How many players?, How many points?) I created a method that would print out questions and ask for user input. This input is fed into the "Game" class.

The Game:

I used object oriented design. I created a Player object which would store information about the player. Each player object had various instance variables such as their name, their score, their prompt deck, etc. The player class would also contain methods which would get and set a player's attributes.

I created a Game object as well. The Game object contained information about the Card Czar, the prompt card and response card decks, the points needed to win, and the players playing the game. There were various methods in the Game Object. Many methods called other methods in the same class. Some methods rotated the player selection and rotated the card czar. Other methods would ask the user for their input. Other methods would display and calculate the score. The **`start()`** method would run the game and would continue until **`isWinner()`** would return true, indicating that someone had reached the necessary points.

I used a wide variety of arrays and dictionaries to store information about the players, their response decks, their points, and the cards they've selected. Many of the algorithms I wrote focused on looping through the possible players, showing their response deck, and letting them pick a card. Every time they selected a card to put down, a new one had to popped from the main response card deck into their hand. I also wrote algorithms to rotate the card czar every round.

Error Checking:

I wrote a method to check every user input and see if it was a number and if it was in range of the possible options I presented. If it did not meet the standards, and error message would display and the user would have to prompted again to input a value.

External Libraries:

pandas, pickle, random, numpy

Improvement:

If I had to redo this game, I'd try to make it more object oriented by having classes for the scoreboard, list of players, and for the Card Czar.