CS 111 - Final Project

A Roulette Simulator written in Java

Robert Edmonston 5/19/2021

Roulette

Overview

- Roulette is a classic casino game where players place one or more bets on a table and then watch as the roulette wheel is spun causing a marble to land in a colored numeric slot which determines if they win.
- Players may bet on a single number, various groupings of numbers, colors red and black, whether the numbers are odd or even, or if the numbers are high (19-36) or low (1-18).

Roulette Simulator

Features

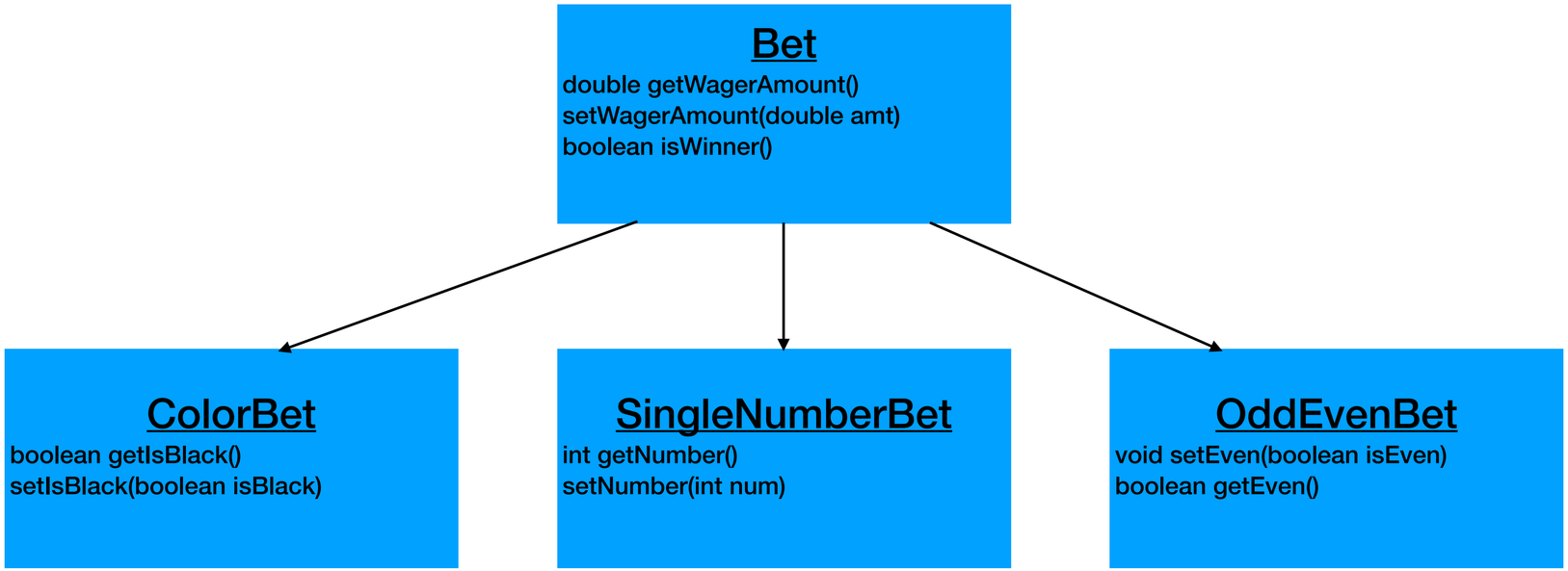
- The application will allow users to maintain a personal profile which will be read from and saved to a file. Data that will stored includes the user's name, betting history, and account balance. When the application is started it will ask the user for their name and load the data from the file with their name, and if no file exists it will create a new one and give them a balance of $1000.00.

- Users will be prompted to create one or more bets. For each bet, they must indicate what the bet type is (e.g. single number, color, even/odd) and then enter the appropriate value depending on the type of bet selected. Once the information regarding the type of bet is entered the user must specify how much they will wager on the bet.

- After the bet(s) have been created the user can specify they are ready for the wheel to be spun. The application will randomly select a winning number to simulate an actual wheel spin. Once the winning number has been determined the bets will be settled with the user adjusting their account balance and betting history as necessary and saving it to their personal profile.

Bet Types

Partial Class Diagram



Possible Challenges

And Feature Considerations

- Creating an interface for bets that cover multiple numbers (e.g. on a roulette table a player may place a chip down in a spot that straddles 4 numbers). Must allow user to bet on the right combinations only to properly simulate a real roulette table.
- Exception handling. How to handle things like corrupt files, bad input, unexpected program termination, etc.
- Possible feature consideration - a visual representation of the wheel spin such as some kind of ASCII art and a delay to the output when a wheel is spun to give it a more realistic feel and make the application more fun to play.
- Not sure how to handle it when the user’s balance reaches 0. Possibly delete profile and force them to create a new one?

The End