```
# Program: Poker Hand Assignment
# Date: Nov. 29, 2022
# Programmer: Tahmid Ehsan
# Description: A simulation of a poker game using
# classes, functions and imports.


# Imports classes and libraries
from Hands import *
import time
import sys

# Uses system to print out different lines withn the same line
def print_same_line(string):
    sys.stdout.write(string)
    sys.stdout.flush()

# Function used to display the stats of players at any given moment
def display_stats():
    print()
    print("Player 1:\t\t\tPlayer 2:")
    for i in range(5):
        print(player1.show_hand()[i],"\t\t\t",
              player2.show_hand()[i])

# Generates information for a new card
def new_card():
    return card_deck.get_card()


# While loop set to rerun the program at user's request
random_variable = True
while random_variable:

    # Initializes deck object and shuffles it
    card_deck = Deck()
    card_deck.shuffle()

    # Defines two objects as players
    player1 = Hands()
    player2 = Hands()
```

```
print("-------------------------------------------------"
      "----------------------")
print()
print("Player 1:\t\t\tPlayer 2:")

# For loop used to display alternating card distribution
for i in range(5):
    new = new_card()
    player1.add_2_hand(new)
    new = new_card()
    player2.add_2_hand(new)
    print_same_line(str(player1.show_specific_card(i)))
    time.sleep(1)
    print("\t\t\t",player2.show_specific_card(i))
    time.sleep(1)

print()
print()

# Changes specific cards in hands using for loops
display = input("Would you like to display the stats of the two "
                "players?: (y/n)\n")
if (display == "y"):
    display_stats()
    print()

card_switches = input("Would you, player 1, like to"
                      " switch cards?: (y/n)\n")
if (card_switches == "y"):
    amount = int(input("How many?: (0,2)\n"))
    switches = []
    for i in range(amount):
        card_number = int(input("Please enter which card"
                                " number you would "
                                "like to switch with: (0,4) \n"))
        switches.append(card_number)
    for i in range(len(switches)):
        new = new_card()
        player1.change_hand(switches[i],new)

display = input("Would you like to display the stats of the"
                " two players?: (y/n)\n")
if (display == "y"):
```

```
    display_stats()
    print()

# Organizes the ranks of the hands of the players in
# order to determine their hand type and game winners.
player1.organize()
player2.organize()
player1.hand_type()
player2.hand_type()
player1.game_points()
player2.game_points()
player1.game_tie_breaker()
player2.game_tie_breaker()


# Makes comparisions in order to to determine game winner
if (player1.game_points() > player2.game_points()):
    print(f'Player 1 wins with their {player1.hand_type()}'
          f' hand type over player 2s '
          f'{player2.hand_type()} hand type!')

elif (player1.game_points() == player2.game_points()):
    if (player1.game_tie_breaker() > player2.game_tie_breaker()):
        print(f'Player 1 wins by a tie breaker, they '
              f'have the better {player1.hand_type()}!')
    elif (player2.game_tie_breaker() == player1.game_tie_breaker()):
        print("Neither player wins, true tie.")
    else:
        print(f'Player 2 wins by a tie breaker, they '
              f'have the better {player1.hand_type()}!')
else:
    print(f'Player 2 wins with their {player2.hand_type()}'
          f' hand type over player 1s '
          f'{player1.hand_type()} hand type!')
print()
print()

# Asks user if they wish to exit the program
exit_var = input("Would you like to restart"
                 " the program?: (y/n) \n")
print()

# If user decides to rerun the program
```

```
# certain lists must be cleared
player1.clear_hands()
player2.clear_hands()

if (exit_var == "n"):
    print("--------------------------------------------------"
      "----------")
    random_variable = False
```