Christopher Lim (S11) - 11935103

Test Scripts Documentation (for Phase 1 MP Submission)

How to run

1. Make sure you are in the `src` directory.
2. javac main/filepath (ex. javac main/tests/Test1.java)
3. java main/filepath (ex. java main/tests/Test1)

Driver.java

Requirement #2

This Driver Class is for simulating a user defined number of players to draw the cards from a Deck consisting of Action Cards(Orange Deck) where the first Player to reach a total asset of $40000 (balance - debt >= 40000) wins.

Sequences:

1. Generate CareerDeck and SalaryDeck
2. Generate Players and assign the top CareerCard and SalaryCard of both CareerDeck and SalaryDeck
3. Generate OrangeDeck (composed of ActionCards)
4. Let Players take turns to draw a Card until a Player reaches the Goal (40000)

Note: Each turn waits for a Enter key to proceed to the next turn.

This shows #2 requirement for the MP Phase 1 submission

tests > Test1.java

Test Script #1

Generates CareerDeck and SalaryDeck then creates 3 Players which will have the top card of each CareerDeck and SalaryDeck as CareerCard and SalaryCard.

Sequence:

1. Generate CareerDeck and SalaryDeck
2. Output the CareerDeck and SalaryDeck
3. Create 3 Players and assign the top most CareerCard and SalaryCard from both the CareerDeck and SalaryDeck
4. Output the CareerDeck and SalaryDeck

This proves that generating CareerDeck and SalaryDeck works and picking the top most card of each deck works since it is successfully assign to the Player

tests > Test2.java

Test Script #2

Generates the OrangeDeck, BlueDeck, CareerDeck, and SalaryDeck then draws until it shows the scenario where all the decks are empty.

Sequence:

1. Generates the OrangeDeck, BlueDeck, CareerDeck, and SalaryDeck
2. Displays each decks
3. Empty each deck by drawing a Card until the Deck is empty
4. Draw a card 1 last time to show the deck is empty

This proves that drawing a Card from the Deck until the Deck is empty works.

tests > Test3.java

Test Script #3

This test script is showing that ActionCard.activate() does the correct action to the Player just as expected

Sequence:

1. Generate CareerDeck and SalaryDeck
2. Generate Players and assign the top CareerCard and SalaryCard of both CareerDeck and SalaryDeck
3. Generate OrangeDeck (composed of ActionCards)
4. Draw the top card from the Deck
5. Activate the Card while comparing the values in the Player

This proves that the ActionCard.activate() does the expected action to the player