Homework 1 CS3330 Fall 2014

Objectives:

- Introduction to basic OOP (Constructor, Mutators, and Accessors)
- Improve upon basic Java skills
- Use Random in java

Purpose:

Homework 1 is the foundation to the next 3 homeworks, by the end of the semester you will have created a complex game that has RPG similarities. The component of the game your are making in this homework is obtaining user input from the keyboard, allowing for decision to be made by the game player. The program is called Scavenger, you will look for items to add to your personal bag, once you collected the max amount items your bag can hold, the game will end.

Scanvenger.java

Attributes to add:

private static Scanner userInput: The Object used for collecting and parsing user input from the keyboard.
private static Item[] playerItems: An array of type Item used to hold the collected items from the gameItems.
private static Item[] gameItems: An array of type Item that holds are the possible items the game has available.
private static Command[] gameCommands: An array of type Command that holds the game commands.
private static int currentPlayerItemsIdx: An integer used to keep track where the current empty index in the playerItems array.

Methods to Implement:

public static void initGameDataAndArrays(): When called the method initializes all the private attributes in the program. The initialization are as follows:

Instantiate a new Scanner Object for user_input attribute.

Instantiate a new Array of Items with a size of 10 for playerItems attribute and fill the area with new Items that have empty for the name, and zeroes for the remaining parameters.

Instantiate a new GameDataReader Object and call getGameItems from the GameDataReader class and assign the returning array to gameItems array.

Intialize the currentPlayerItemsIdx to zero

Instantiate a new Array of Commands for gameCommands attribute and fill the game command names. Game Commands: help, pickup, drop, view,quit.

private static Item scavengeArea(): When called, the method simulates the player is actually searching the area for an item and grabs the item. Using the Random Class to create a number within the bounds of the gameItems array and a random value between 1 and 4. Using the number generated between 1 and 4 loop that many times, displaying to the user “Scavenging area for an Item...”. Lastly, return an item from the gameItems array at the index randomly generated within the bounds of the gameItems.

public static boolean runCommand(String commandName, Item item):

When called the method uses the commandName parameter to decide which method to call and returns true if the user didn’t enter “quit” for the commandName. If the commandName is “pickup”, call addItemToPlayerItems method with the item passed item parameter, upon success display the message “Item successfully added”, and return true. If the commandName is “drop”, display the message “Item not added”, and return true. If the commandName is “help”, call the method displayGameCommands, and return false. If the commandName is “view”, call the method displayPlayerItems, and return false. If the commandName is “quit”, display the message “Game Terminated” and call System.exit(0).

public static void displayPlayerItems(): Displays the playerItems array using the accessor methods to grab the name and value of each item. If the currentPlayerItemsIdx is equal to zero, display the message “Bag is empty”.

public static void displayGameCommands(): Displays the gameCommands array using the accessor methods to grab the name of each command.

public static boolean validCommand(String[] splitStrings): Verfies that the first element in the parameter splitStrings array is a game command in the game. If the commands is found return true, else false.

public static boolean addItemToPlayerItems(Item item): If the currentPlayerItemsIdx is greater or equal to the size of playerItems return false else add the passed in parameter item to the playerItems at the index of currentPlayerItemsIdx and increase currentPlayerItemsIdx by 1.

public static void PlayerItemsStats(): Finds the total sum of the value of items collected, the minimum **weight**, and the maximum **weight** in the playerItems array. Display the results to the user.

Command.java

Attributes:

private String commandName

Methods to Implement:

public Command(String commandName): sets the attribute commandName using the mutator method setCommandName.

private void setCommandName(String commandName): Assigns the passed in parameter commandName to the attribute commandName. Note: when using an object's attribute use the **this** keyword to refer to it.

public String getCommandName(): Returns the attribute commandName.

Item.java

Attributes:

private String name
private int value
private int weight

Methods to Implement

public Item(String name, int value, int weight): calls the mutators methods to assign the attributes.

Add the mutator and accessor for each of the following attributes. If you need help look above at Command.java for assistance. Each mutator must begin with “set” and each accessor method must begin with “get”.

GameDataReader.java

Attributes:

NONE

Methods to Implement

public GameDataReader(): Is an empty constructor, does nothing

public Item[] getGameItems(): Returns a filled array of Items created from the contents from the file located at “GameData/GameItems.csv”. Create an array of Items that is size 25, each time you read from the file, split and convert the read in string, create a new Item to store in the created array. To parse a csv, you will need to use the String.split method to break apart the read in string into a string array. To convert a string into an integer, you will use the Integer.parseInt method.

Sample Output (Items will vary due to Randomness):

```
Enter a name: matt
Welcome to the Scavenger Game matt

Scavenging area for an Item...
Scavenging area for an Item...
Item found: Premier Potion
Command: pikup
Invalid Try Again, Command: pickup

Item successfully added Premier Potion

Scavenging area for an Item...
Scavenging area for an Item...
Scavenging area for an Item...
Item found: Ultimate Revolver
Command: drop

Item not added

Scavenging area for an Item...
Scavenging area for an Item...
Item found: Standard Revolver
Command: view
Premier Potion 40

Command: help
help
pickup
drop
view
quit

Command: pickup

Item successfully added Standard Revolver

Scavenging area for an Item...
Scavenging area for an Item...
Scavenging area for an Item...
Scavenging area for an Item...
Item found: Standard Herbs
Command: pickup

Item successfully added Standard Herbs

Scavenging area for an Item...
Scavenging area for an Item...
Scavenging area for an Item...
Scavenging area for an Item...
Item found: Standard Flamethrower
Command: pickup

Item successfully added Standard Flamethrower

Scavenging area for an Item...
Scavenging area for an Item...
Scavenging area for an Item...
Scavenging area for an Item...
Item found: Ultimate Flamethrower
Command: pickup

Item successfully added Ultimate Flamethrower

Scavenging area for an Item...
Scavenging area for an Item...
Scavenging area for an Item...
Item found: Standard Bat
Command: pickup

Item successfully added Standard Bat

Scavenging area for an Item...
Scavenging area for an Item...
Scavenging area for an Item...
Scavenging area for an Item...
Item found: Premier Axe
Command: pickup
```

```
Item successfully added Premier Axe

Scavenging area for an Item...
Scavenging area for an Item...
Item found: Standard Machete
Command: drop

Item not added

Scavenging area for an Item...
Scavenging area for an Item...
Item found: Standard Flamethrower
Command: pickup

Item successfully added Standard Flamethrower

Scavenging area for an Item...
Scavenging area for an Item...
Scavenging area for an Item...
Item found: Premier Bat
Command: view
Premier Potion 40
Standard Revolver 20
Standard Herbs 20
Standard Flamethrower 20
Ultimate Flamethrower 60
Standard Bat 20
Premier Axe 40
Standard Flamethrower 20

Command: pickup

Item successfully added Premier Bat

Scavenging area for an Item...
Scavenging area for an Item...
Scavenging area for an Item...
Scavenging area for an Item...
Item found: Premier Revolver
```

```
Command: drop

Item not added

Scavenging area for an Item...
Scavenging area for an Item...
Scavenging area for an Item...
Item found: Premier Bat
Command: pickup

Item successfully added Premier Bat

Premier Potion 40
Standard Revolver 20
Standard Herbs 20
Standard Flamethrower 20
Ultimate Flamethrower 60
Standard Bat 20
Premier Axe 40
Standard Flamethrower 20
Premier Bat 40
Premier Bat 40

Player Item Stats: Total Weight is 80 Total Value is 320 Min Weight found is 5 Max Weight found 15
```