Sean Blanton

Introduction to Java

Project 04

Oct 14 2015

**Jar file is created and is executable (25pts)**

-Evidenced in screenshot

**Programming Guidelines are followed (25pts)**

-Required comments present in class comment block

-Indentations consistent

-Block bracing consistent

**Report with screenshots is created (25pts)**

-This document

**FlashCard super class meets specs (25pts)**

-FlashCard.java

-*should be an abstract class:*

**public abstract class** FlashCard

-*should contain at least two abstract methods: getQuestion, and getAnswer. These two methods should return Strings.*

***public abstract*** *String GetQuestion();*

***public abstract*** *String GetAnswer();*

**MathCard sub class meets specs (25pts)**

-MathCard.java

-*should inherit from the FlashCard class*

**public class** MathCard **extends** FlashCard

-*Constructors for this class should take three arguments: valueA, mathOperation, and valueB*

**public** MathCard(**int** arg1, **char** operation, **int** arg2)

-*should compute the correct answer (valueA mathOperation valueB)*

**private double** doMath()

-*This can be done when the object is instantiated or when the getAnswer method is called*

**public** String GetAnswer()

{

**double** answer = doMath();

-*should deal with ​ impossible ​ math (division by zero).*

**case *CHAR\_DIV\_OP***:

*//Protect against divide-by-zero*

**if**(0 != **m\_rhArg**)

{

result = (**double**) **m\_lhArg** / (**double**) **m\_rhArg**;

}

**else**

{

**m\_errorString** = **"User attempted to divide by 0!"**;

}

**break**;

**VocabCard sub class meets specs (25pts)**

-VocabCard.java

-*should inherit from the FlashCard class*

***public class*** *VocabCard* ***extends*** *FlashCard*

-*Constructors for this class should take two String arguments.*

**public** VocabCard(String word, String englishWord)

**CardStack class is created and meets specs (25pts)**

-CardStack.java

-*should have an ArrayList of ​ Flashcards*

**private** ArrayList<FlashCard> **m\_cardStack**;

-*appropriate methods for creating and/or adding new flash cards to the ArrayList*

**public boolean** AddCard(FlashCard newCard)

**public boolean** AddCard(**int** arg1, **char** operation, **int** arg2)

**public boolean** AddCard(String word, String englishWord)

-*should have a “shuffle” method for randomizing the cards in the ArrayList*

**public void** Shuffle()

-*”shuffle” method should be one line of code*

Collections.*shuffle*(**m\_cardStack**);

-*needs a nextCard method that will return the next card from the ArrayList*

**public** FlashCard NextCard()

**Driver class is created and meets specs (25pts)**

-Driver.java

-*should instantiate a CardStack object and add 4 VocabCard and 4 MathCards to it.*

m\_cardStack = **new** CardStack();

addMathCards();

addVocabCards();

-*should call the CardStacks shuffle method and then display the first card’s question*

m\_cardStack.Shuffle();

exit = *printNextQuestion*();

-*When the “a” key is pressed the card’s answer should be shown*

**case *CHAR\_GIVEANSWER***:

*//Print the Answer for current FlashCard*

System.***out***.println(String.*format*(**"A%d: %s"**, *m\_cardID*, *m\_currentCard*.GetAnswer()));

**break**;

-*When the “n” key is pressed the next card’s question should be shown.*

**case *CHAR\_NEXTCARD***:

*//Print the Question for next FlashCard*

quitTriggered = *printNextQuestion*();

**break**;

-*When the last card’s answer has been shown the program should end*

**if**(**null** != *m\_currentCard*)

{

System.***out***.println(String.*format*(**"Q%d: %s"**, *m\_cardID*, *m\_currentCard*.GetQuestion()));

}

**else**

{

System.***out***.println(**"No more Flash Cards!"**);

outOfCards = **true**;

}

-*any time the “q” key is pressed the program should end*

**case *CHAR\_QUIT***:

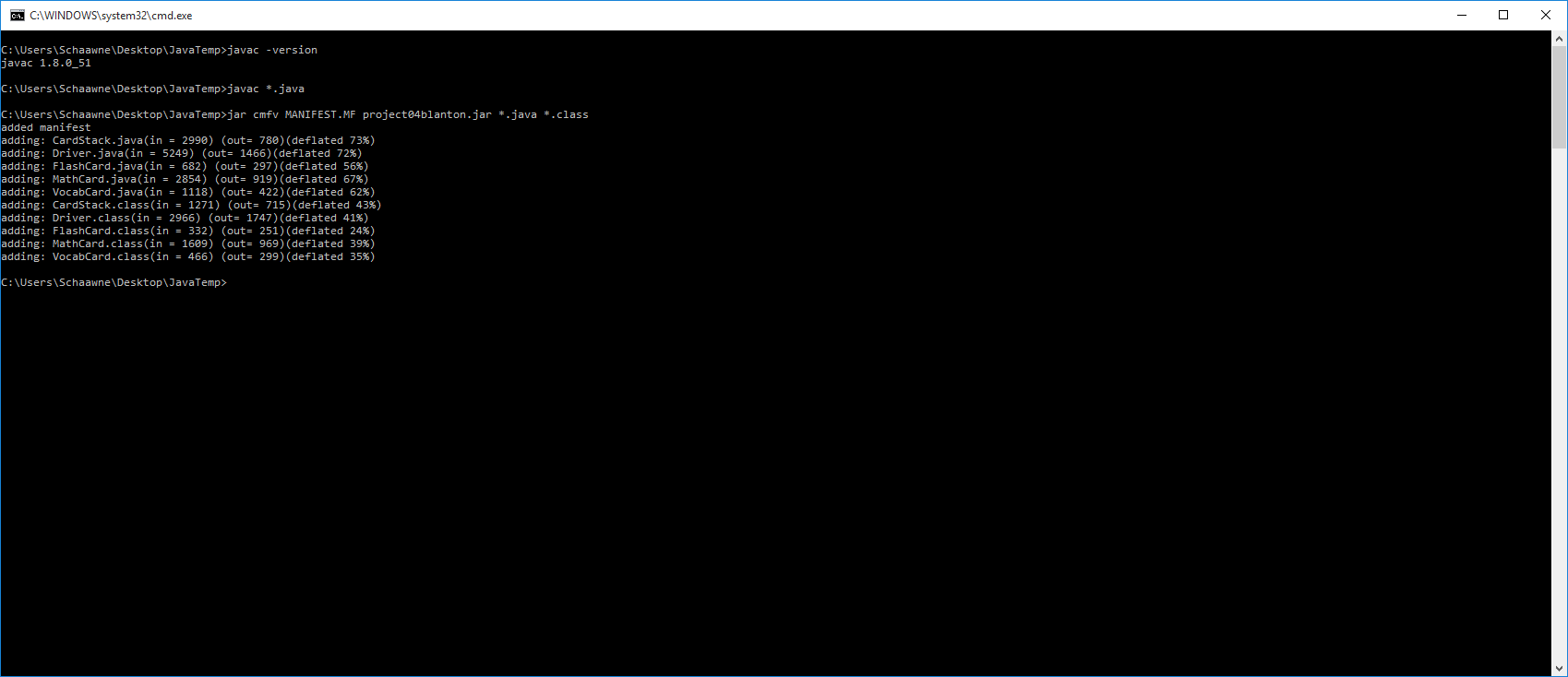
quitTriggered = **true**;

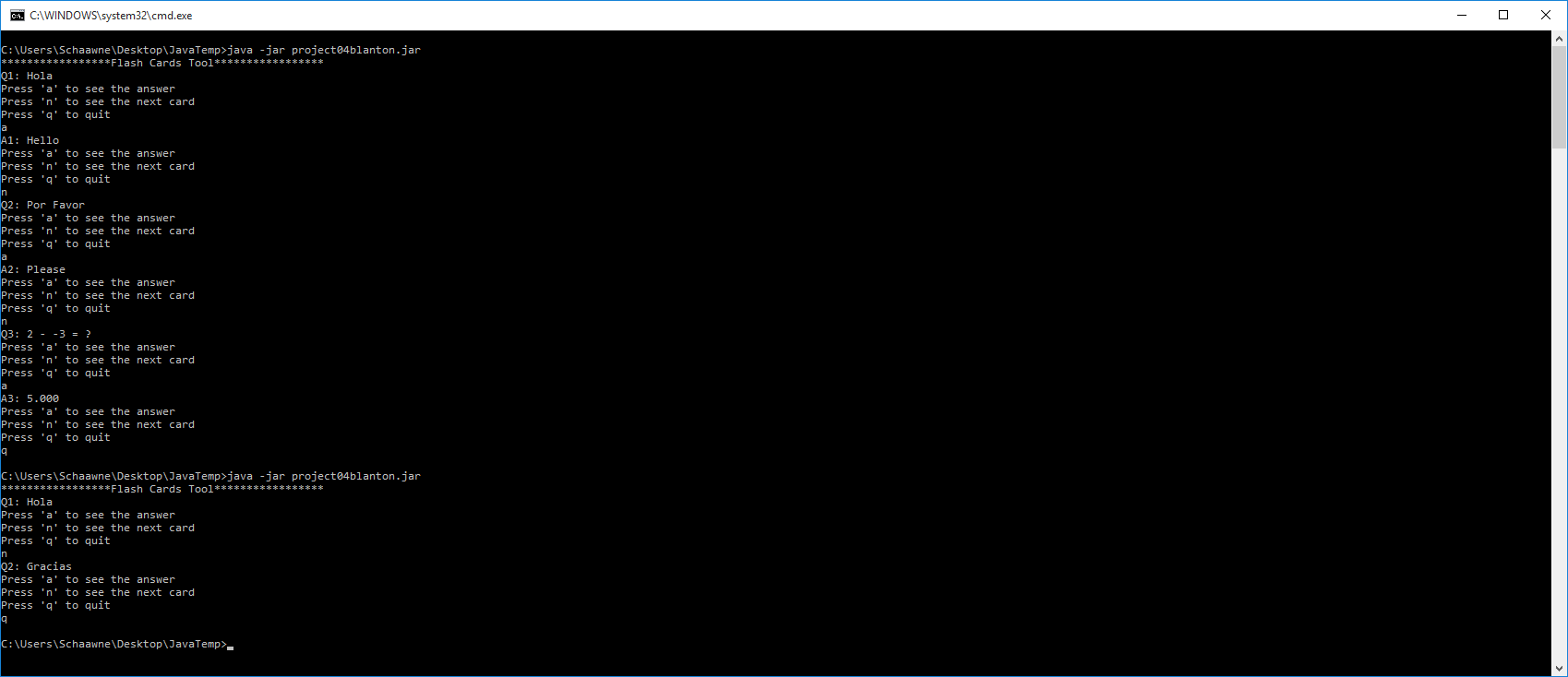
**break**;

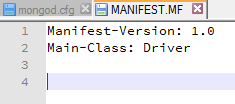
-*Use a Scanner for input*

*m\_inputScanner =* ***new*** *Scanner(System.****in****);*

**char** userInput = *m\_inputScanner*.next(**"."**).charAt(0);







# 

# 

# 

# Summary

Created a basic Flash Card and Card Stack set of classes in java. Used intelliJ IDEA community IDE. Used consts for user input and private functions with arrays for easy modification/expansion. Manually compiled and packed source for release to provide concise screenshots of functionality. Attempting to use javadoc comments as practice but no requirement to submit generated documentation.