Issues in Information Systems
Volume 18, Issue 3, pp. 29-39, 2017

FLIPPED LEARNING ACTIVITES FOR AN INTRODUCTION TO JAVA COURSE

Hsiang-Jui Kung, Georgia Southern University, hjkung@georgiasouthern.edu
LeeAnn Kung, Rowan University, kung@rowan.edu

ABSTRACT

Flipped learning approach becomes popular in higher education. Many studies have focused on discovering benefits of the flipped learning but fell short on demonstrating the design of flipped learning. Computer programming classes are perfect match for the flipped learning approach. This paper presents a series of detailed flipped learning activities of the nested if statement for an introduction to Java class to first demonstrate the structure of a flipped classroom and to lay the foundation of future research.

Keywords: active learning, flipped classroom, student-centered learning, teaching/learning strategies.

INTRODUCTION

Student engagement is one of the key elements in the student-centered learning environments (Shea, Hayes, Smith, Vickers, Bidjerano, & Pickett, 2012). Strayer (2012) reported that educators often have difficulties managing their finite classroom time to achieve an effective balance between lectures and active learning activities. Flipped classroom approach addresses these challenges by "inverting the classroom", that is, interchanging "events that have traditionally taken place inside the classroom now take place outside the classroom and vice versa" (Foertsch, J., Moses, G., Strikwerda, J., & Litzkow, M., 2002). In a traditional class, the instructor lectures and students take notes as their learning activities. Students then complete homework post-class. In the flipped environment, in-class lectures are "flipped" with collaborative hands-on activities. A typical flipped classroom approach usually provides online video lectures and homework assignments prior to class sessions. Students are in charge of learning the material on their own pace and to prepare for the hands-on problem solving activities that will be carried out during class time (Bergmann, Overmyer, & Wilie, 2012; Davies, Dean, & Ball, 2013; Foertsch, Moses, Strikwerda, & Litzkow, 2002; Fulton, 2012; Hughes, 2012; Zappe, Leicht, Messner, Litzinger, & Lee, 2009). Hands-on activities are designed to replace lectures in class and for students to become more active and interactive. Educators can commit more in-class time to monitor students' performance and to provide adaptive and instant feedback to students (Fulton, 2012; Herreid & Schiller, 2013; Hughes, 2012).

Prior studies show different evaluations of flipped class approach. Some studies present benefits of positive impact on learning (e.g., Herreid and Schiller, 2013; Zappe et al., 2009) while some did not find significant difference between traditional and flipped approaches (Davies et al., 2013; Strayer, 2012). Many studies focused on the investigation of the strength and weakness of the flip classroom learning but provided little details of the specific flip class room activities. While contributions of these studies are not to be ignored, a demonstration of actual application of the flipped classroom design principles is needed to guide educators who are novice in this approach and to provide backdrops of research of flipped class approach.

Bergmann and Sams (2012) suggested a list of flip design considerations: support from administrators, support from IT department, time, and thoughtful educators. However, their guidance was limited to technological elements. Chen and colleagues (2014) proposed a FLIPPED model for the flipped learning design: flexible environments, learner-centered approach, intentional content, professional educators, progressive networking activities, engaging learning experiences, and diversified and seamless learning platforms. The progressive networking activities refer to "learning by doing" and/or "learning by networking" activities. The engaging learning experience focuses on tracking the effectiveness of students' outside class learning as well as their in-class learning. The diversified and seamless learning platforms are designed and operated to meet the criteria of the course domain knowledge for individualized, differentiated, personalized learning in a flexible, ubiquitous and seamless manner.

In this paper, we present a series of nested if statement activities in an introduction to Java class to demonstrate the design of a flipped class which includes the before, during, and after class activities/assignments. These are described in the following section. We then present future research plan.

FLIPPED LEARNING ACTIVITIES

Programming classes are perfect candidates for the flipped learning approach because those classes require a lot of hands-on activities in order to practice programming principles and to master a specific programming language. We designed a series of flipped learning activities targeting an introduction to Java class. We choose the nested if statement to demonstrate the use of flipped learning approach in programming class. We turned to prior instructional studies to select the pre, during, and post class activities for this class.

DeGrazia and colleagues (2012) reported that students who were supplied with optional lecture vedios came to class much better prepared than when those who were only been given textbook readings assignments because college students don't generally complete reading assignments (Sappington et al., 2002). Recommended by students, a required pre-class quiz on the lecture material to enforce the completion of before class activities is considered a "best practice" means and is conducted by many instructors. Zappe et al. (2009) concluded that students preferred interactive class time more than in-person lectures. Applying these suggestions and educational principles, the before class activities are designed mostly related to the Bloom's (Krathwohl, 2002) lower-level learning such as remembering and comprehending. Both delivery and content of the before class activities are structured to help students acquire content knowledge and prepare them for the application of focus content in the face-to-face class. Instructors use offloaded content relevant to the topic of the class session for the before class activities.

Before Class

The before class activities contains various online modules including power point slides, a tutorial video, and a required quiz. Students can complete these modules in any sequence of their choices. Power point slides are used to explain the concepts. A flow chart (Figure 1) demonstrates the decision logic of the nested if statement (Figure 2). Students can either view the power point slides or read the text book to learn the basics and then take the required online quiz (Figure 4). This quiz is a "low-stake" quiz which constitutes only a small percentage of the final course grade.

We used Adobe Connect software to create the tutorial video. Video links are provided so students can access them throughout the semester. Students follow the instruction and demonstration of the tutorial to create a JAVA class, in this case, a payroll program. Figure 3 shows the model codes and steps shown in the tutorial video. There are build-in individual components to prevent plagiarism. In this case, students must replace "FML" shown in the example with their own initials. Students are required to follow the video, write a Java class similar to Figure 3, and submit the .java file by the deadline. All the before-class activities are due one day before the scheduled class time.

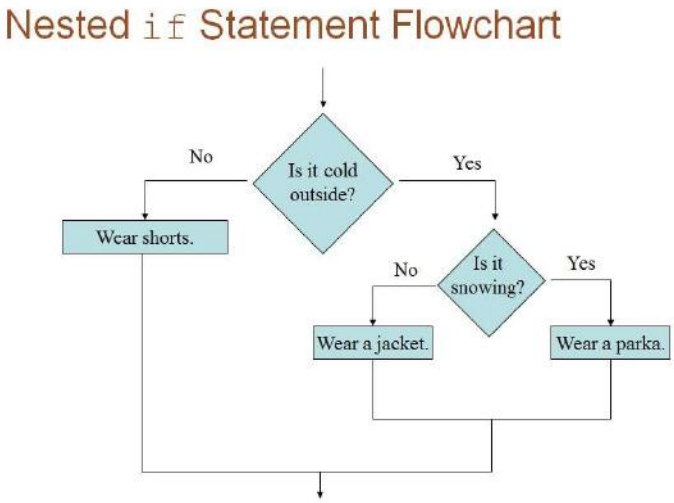


Figure 1. Flowchart of Nested if Statements

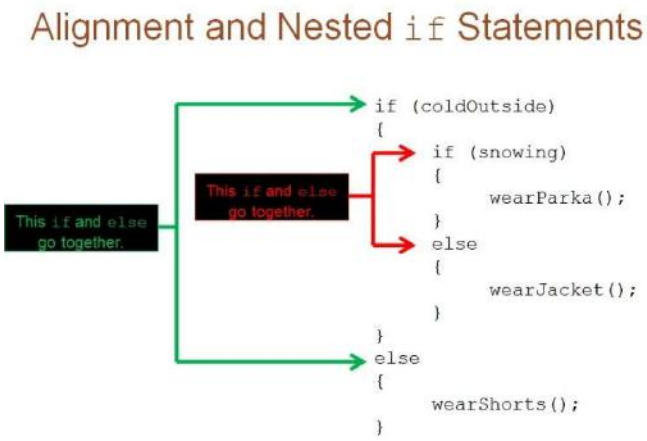


Figure 2. Nested if JAVA Statements of the Flowchart

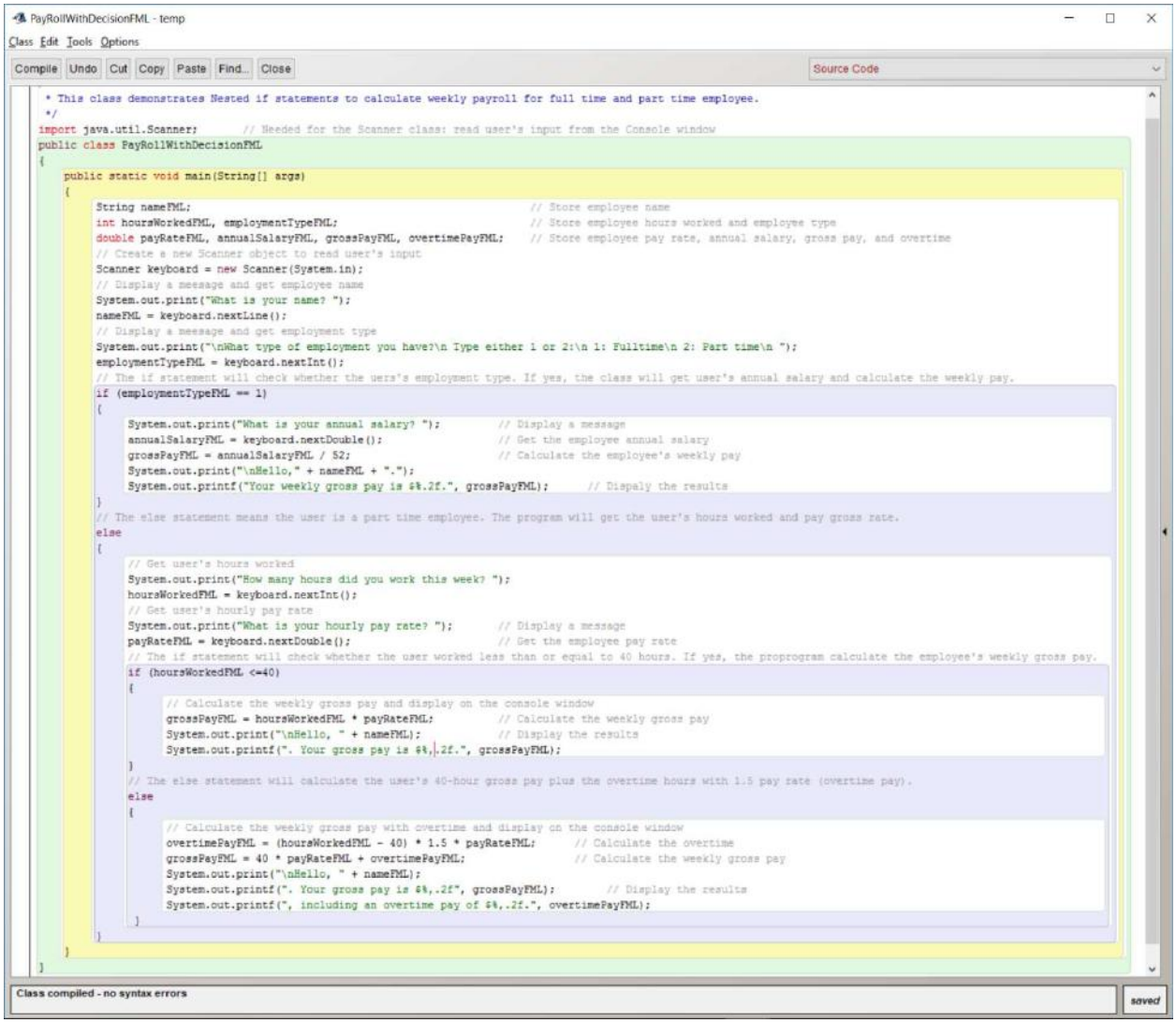


Figure 3. Nested if Statements: Lab Tutorial

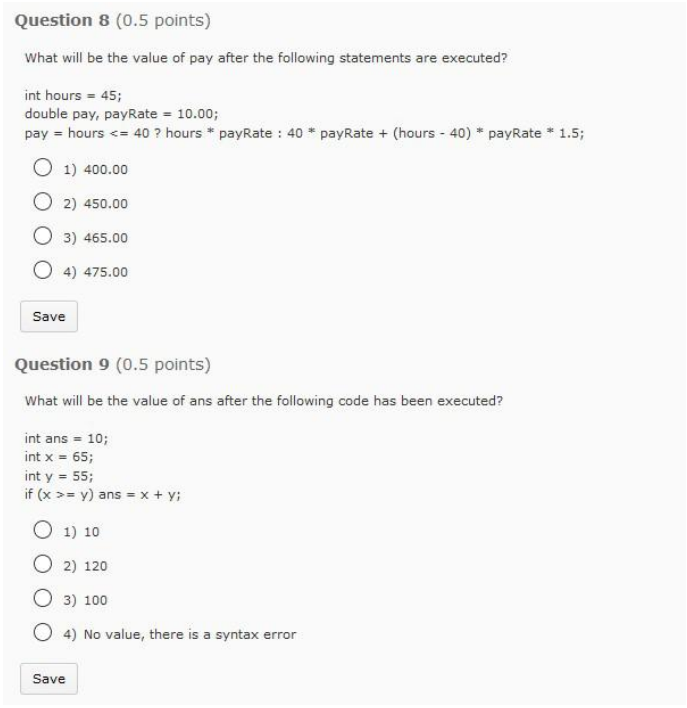


Figure 4. Sample Quiz Questions

During Class

With the topic objectives in mind, the face-to-face class is designed for the application of the knowledge acquired by students from the online before- class activities using the build-in active learning strategies. Active learning strategies enable students to achieve higher levels of learning of Bloom's taxonomy (eg, applying, analyzing, and synthesizing). There are many active learning strategies for educators to use in the face-to-face class. When designing the face-to-face component, it is important to choose only a few active learning strategies to use throughout the course rather than a different one for each class time. This will allow students to become familiar with the active learning strategy and avoid the risk of students focusing on the process of the strategy rather than the learning related to the content.

The best way to learn programming is by emulating sample programs. In the nested if statement topic, a coding activity was used (Figure 5). The instructor demonstrates and explains step by step and students follow. Students are encouraged to change the codes as long as their final produced program meets all the requirements. While student busy typing the Java program, the instructor helps students by trouble shooting simple common errors such as typos or other issues. After all students completed the Figure 5 Java class without any problem, the instructor then starts the discussion component to solicit feedback or reflections such as lessons learned. The instructor can also explain the concepts within this program and modify the program with different statements and quiz the students of the possible outcome after changes.

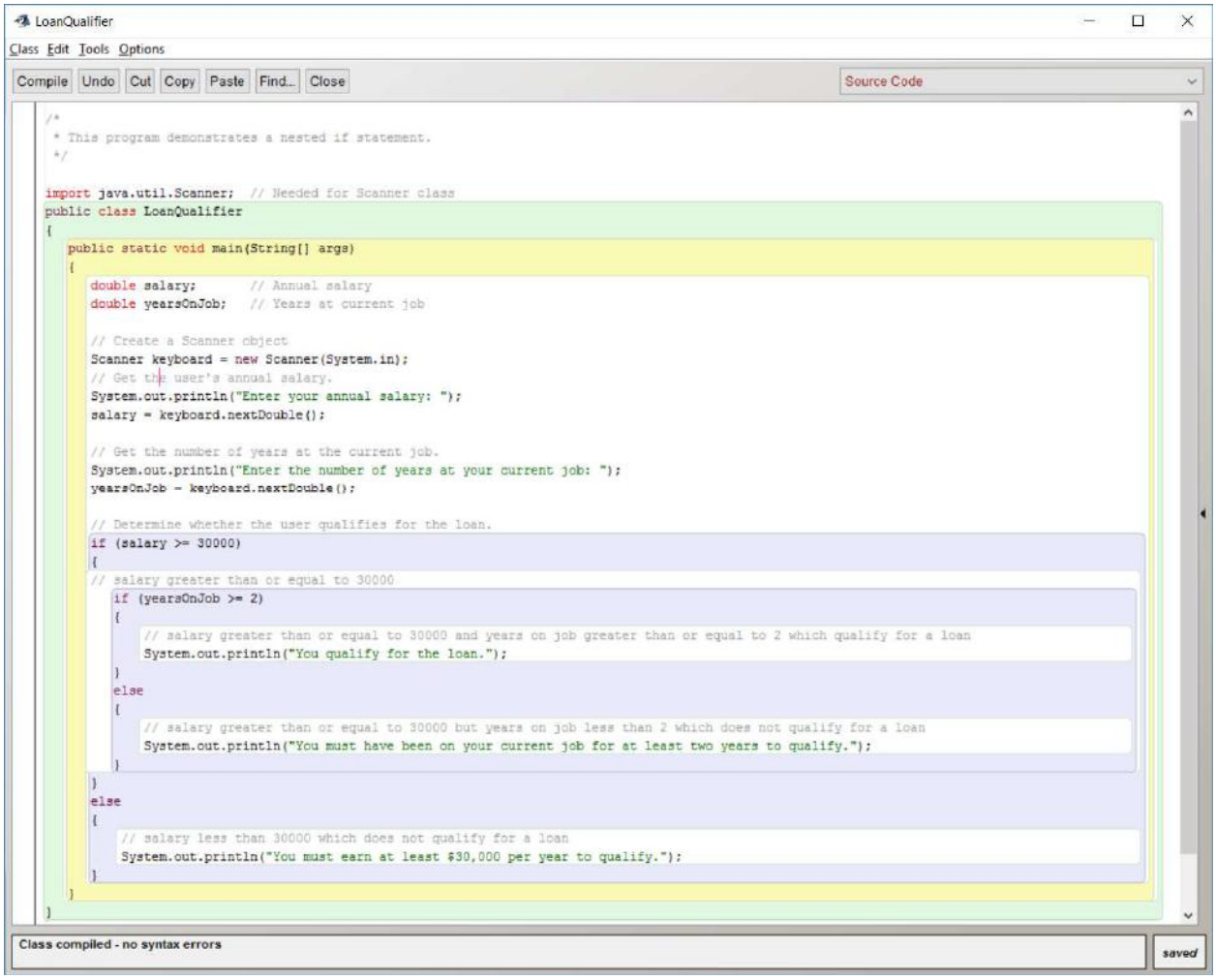
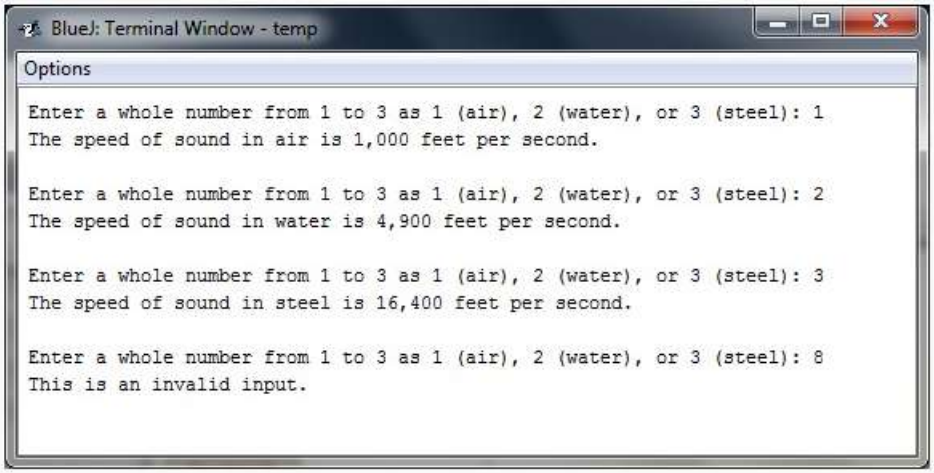


Figure 5. In-class Coding

After all students completed the Java program and all questions were answered, a pop quiz (Figure 6) can be used to assess students' learning, specifically in checking whether students can apply the concepts/statements learned from the in-class coding as well as providing another opportunity for the instructor's feedback.

Question 1 (5 points)



A colleague of yours had a partially finished Java class. Your task is to use nested if statement to check the user input. The results after running your finished class will be similar to the attachment. The decision logic is described as the followings:

1) the user input is 1, then display the speed of sound (1,100 feet per second) in the Terminal window,

2) the user input is 2, then display the speed of sound (4,900 feet per second) in the Terminal window,

3) the user input is 3, then display the speed of sound (16,400 feet per second) in the Terminal window, and

4) the user input is other whole number, then display invalid input in the Terminal window.

```
import java.util.Scanner;
/**
   This program uses nested statement to
   check the user's input and determine the Speed of Sound
   in different medium.
*/
public class PopQuiz7
{
   public static void main(String[] args)
   {
      int medium;         // To hold the user input number

      // Create a Scanner object for keyboard input.
      Scanner keyboard = new Scanner(System.in);

      // Get the user's medium of choice.
      System.out.print("Enter a whole number from 1 to 3 as 1 (air), 2 (water), or 3 (steel): ");
      medium = keyboard.nextInt();

      // insert your code below this line.


   }
}
```

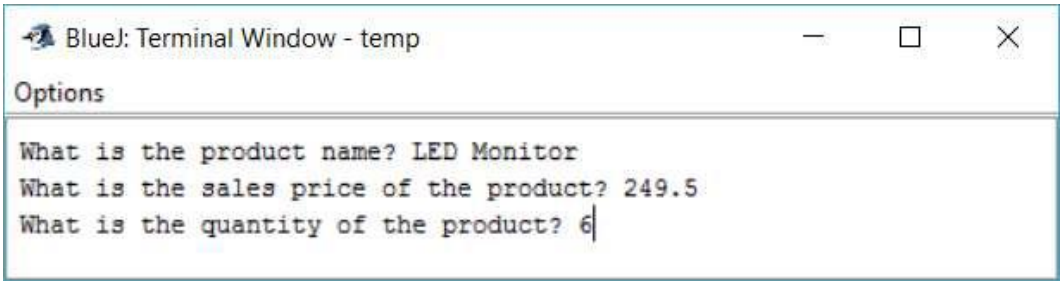
Figure 6. Pop Quiz

After Class

Assessments are an integral part of the after-class component of the flipped classroom and should be aligned with the objectives of the offloaded content and the in-class activities. A lab assignment related to the nested if statement will be a good repetitive assignment to reinforce students' learning of this topic. The following description is the nested if statement assignment. Students need to analyze the problem, synthesize the concepts learned, and apply specific Java statements to complete this assignment. Figure 7 presents the solution of the after-class lab assignment.

Lab Assignment

1. Use BlueJ or other text editor to create a new Java class called **SalesDiscountFML.java**. In this assignment, you will use Scanner to capture user inputs and assign to variables. **Note: In terms of FML, F is your First Name initial, M is your Middle Name initial, and L is your Last Name initial.**
2. In the description part (the comments area), you should include your name, date created/modify this class, and the purpose of this class.
3. Import **java.util.Scanner** class to your Java class.
4. Declare the following variables:
    a. **String** variables: **nameFML**, **inputStringFML**
    b. **int** variable: **quantityFML**
    c. **double** variable: **salesPriceFML**, **totalSalesAmountFML**
5. Use in-line comments to explain your Java statements (see Lab Tutorial).
6. Write Java statements to read the user's input using **Scanner** (the console window) as the followings.
    a. create a **Scanner** object to get the user inputs,
    b. use the **Scanner** object to read the product's name and assign to the **String** variable: **nameFML**.
    c. use the **Scanner** object to read the product's sales price and assign to the **double** variable: **salesPriceFML**.
    d. use the **Scanner** object to read product's sales quantity and assign to the **int** variable: **quantityFML**.
    e. calculate the user's total sales amount and store the result to the **double** variable: **totalSalesAmountFML**.



7. Write a nested if statement to determine the sales discount based on the total sales amount the following flow chart.

After running the main method of your **SalesDiscountFML** class, your results will be similar to the following screen shots.

```
BlueJ: Terminal Window - temp
Options
What is the product name? LED Monitor
What is the sales price of the product? 249.5
What is the quantity of the product? 6
LED Monitor's sales is qualify for 10 percent discount and the discounted total sales amount is $1,347.30 with the price of $249.50 and quantity of 6.
You saved $149.70.
```

You can run with a 5% discount scenario as the following screen shots:

```
BlueJ: Terminal Window - temp
Options
What is the product name? LED Monitor
What is the sales price of the product? 249.5
What is the quantity of the product? 3
LED Monitor's sales is qualify for 5 percent discount and the discounted total sales amount is $711.07 with the price of $249.50 and quantity of 3.
You saved $37.43.
```

Run with no discount scenario as the following screen shots:

```
BlueJ: Terminal Window - temp
Options
What is the product name? LED Monitor
What is the sales price of the product? 249.5
What is the quantity of the product? 2
LED Monitor's total sales amount is $499.00 with the price of $249.50 and quantity of 2.
```

8. Submit your **SalesDiscountFML.java** file before the deadline.

SalesDiscountFML

Class Edit Tools Options

Compile Undo Cut Copy Paste Find... Close Source Code

```
/**
 * This class demonstrates Nested if statements to calculate the discount of a sales.
 */
import java.util.Scanner;       // Needed for the Scanner class: read user's input from the Console window

public class SalesDiscountFML
{
   public static void main(String[] args)
    {
        String nameFML;                                     // Store the product name
        int quantityFML;                                    // Store the sales quantity
        double salesPriceFML, totalSalesAmountFML;     // Store the discount rate, sales price, and total sales amount

        // Create a new Scanner object to read user's input
        Scanner keyboard = new Scanner(System.in);

        // Display a message and get product name
        System.out.print("What is the product name? ");
        nameFML = keyboard.nextLine();

        // Display a messag and get the sales price
        System.out.print("What is the sales price of the product? ");
        salesPriceFML = keyboard.nextDouble();

        // Display a message and get the sales quantity
        System.out.print("What is the quantity of the product? ");
        quantityFML = keyboard.nextInt();

        // Calculate the total sales amount
        totalSalesAmountFML = salesPriceFML * quantityFML;

        // use the nested if statements to determine the discount rate
        if(totalSalesAmountFML > 1000)
        {
            System.out.printf("%s's sales is qualify for 10 percent discount and the discounted total sales amount is $%,.2f", nameFML, totalSalesAmountFML*(1-0.1));
            System.out.printf(" with the price of $%,.2f and quantity of %d.", salesPriceFML, quantityFML);
            System.out.printf("\nYou saved $%,.2f.", (totalSalesAmountFML * 0.1));
        }
        else if(totalSalesAmountFML >= 500)
            {
                System.out.printf("%s's sales is qualify for 5 percent discount and the discounted total sales amount is $%.2f", nameFML, totalSalesAmountFML*(1-0.05));
                System.out.printf(" with the price of $%.2f and quantity of %d.",  salesPriceFML, quantityFML);
                System.out.printf("\nYou saved $%,.2f.", (totalSalesAmountFML * 0.05));
            }
            else
            {
                System.out.printf("%s's total sales amount is $%,.2f with the price of $%,.2f and quantity of %d.", nameFML, totalSalesAmountFML, salesPriceFML, quantityFML);
            }
    }
}
```

Class compiled - no syntax errors

saved

Figure 7. Nested if Statement: Lab Assignment Solution

FUTURE RESEARCH

This paper proposed a series of detailed flipped learning activities for an introduction to Java class. The preliminary result shows that about 70% student prefer the flipped approach to the traditional. The authors plan to develop other series of activities for all the topics covered in the introduction to Java class. After developing the flipped learning activities, the authors intend to apply the flipped classroom approach to the introduction to Java class in the Fall of 2017 using all the activities designed. We then will assess students' learning and survey students in the Java class for their reflections of this approach. We will continue collecting qualitative and quantitative data to get better insight of the flipped classroom approach.

REFERENCES

Bergmann, J., Overmyer, J., & Wilie, B. (2012). *The flipped class: Myths versus reality*. The Daily Riff (Retrieved May 2017 from http://www.thedailyriff.com/articles/theflipped-class-conversation-689.php).

Bergmann, J., & Sams, A. (2012). Before you flip, consider this. *Phi Delta Kappan*, *94*(2), 25.

Chen, Y., Wang, Y., Kinshuk, & Chen. N-S. (2014). Is FLIP enough? Or should we use the FLIPPED model instead? *Computers & Education*, *79*(October), 16-27.

Davies, R. S., Dean, D. L., & Ball, N. (2013). Flipping the classroom and instructional technology integration in a college-level information systems spreadsheet course. *Educational Technology Research and Development*, *61*(4), 563–580.

DeGrazia, J. L., Falconer, J. L., Nicodemus, G., & Medlin, L. (2012). Incorporating screencasts into chemical engineering courses. In Proceedings of the ASEE Annual Conference & Exposition.

Foertsch, J., Moses, G., Strikwerda, J., & Litzkow, M. (2002). Reversing the lecture/homework paradigm using eteachR web-based streaming video software. *Journal of Engineering Education*, *91*(3):267–274.

Fulton, K. (2012). Upside Down and Inside Out: Flip Your Classroom to Improve Student Learning. *Learning & Leading with Technology*, *39*(8), 12–17.

Fulton, K. (2014). *Time for Learning: Top 10 reasons why flipping the classroom can change education*; Corwin, a SAGE Company: Thousand Oaks, CA.

Hamdan, N., McKnight, P., McKnight, K., & Arfstrom, K. (2013). The flipped learning model: A white paper based on the literature review. Retrieved from the Flipped Learning Network website http://researchnetwork.pearson.com/wp-content/uploads/WhitePaper_FlippedLearning.pdf.

Herreid, C., & Schiller, N. (2013). Case studies and the flipped classroom. *Journal of College Science Teaching*, *42*(5), 62.

Kim, M. K., Kim, S. M., Khera, O., & Getman. J. (2014). The experience of three flipped classrooms in an urban university: An exploration of design principles. *Internet and Higher Education*, *22*(July), 37-50.

Kong, S. C. (2014). Developing information literacy and critical thinking skills through domain knowledge learning in digital classrooms: An experience of practicing flipped classroom strategy, *Computers & Education*, *78*(September), 160-173.

Krathwohl D. R. (2002). A revision of Bloom's taxonomy: an overview. *Theory into Practice*. *41*(4), 212-218.

Jensen, J. L., Kummer, T. A., & Godoy, P. (2015). Improvements from a flipped classroom may simply be the fruits of active learning, *Life Sciences Education*, *14*(5), 1-12.

Milman, N. (2012). The flipped classroom strategy: What is it and how can it be used? *Distance Learning ,9*(3), 85-87.

Murray, D., Koziniec, T., & McGill, T. (2015). Student perceptions of flipped learning. Proceedings of Australasian Computer Education Conference, Sydney, Australia: Australian Computer Society.

O'Flaherty J. & Phillips C., (2015). The use of flipped classrooms in higher education: a scoping review, *Internet and Higher Education*, 25, 85–95.

Sappington, J., Kinsey, K., & Munsayac, K. (2002). Two studies of reading compliance among college students. *Teaching of Psychology*, *29*(4), 272–274.

Shea, P., Hayes, S., Smith, S. U., Vickers, J., Bidjerano, T., & Pickett, A. (2012). Learning presence: Additional research on a new conceptual element within the Community of Inquiry (CoI) framework. *Internet and Higher Education*, *15*(2), 89–95.

Strayer, J. F. (2012). How learning in an inverted classroom influences cooperation, innovation and task orientation, *Learning Environments Research*, *15*(2), 171–193.

Stull, J., Varnum, S., Ducette, J., Schiller, J., & Bernacki, M. (2013). The many faces of formative assessment. *International Journal of Teaching Learning in Higher Education*, *23* (1), 30−39.

Weaver, G. & Sturtevant, H. (2015). Design, implementation, and evaluation of a flipped format general chemistry course. *Journal of Chemical Education*, *92* (9), 1437−1448.

Yeung K. & O'Malley P. J. (2014). Making 'the flip' work: Barriers to and implementation strategies for introducing flipped teaching methods into traditional higher education courses, *New Directions, 10,* 59–63.

Zappe, S., Leicht, R., Messner, J., Litzinger, T., & Lee, H. (2009). *"Flipping" the classroom to explore active learning in a large undergraduate course.* Proceedings of the 2009 American Society for Engineering Education Annual Conference and Exhibition.