

Put all java programs in one folder named HW4 and zip. Submit only the zipped folder.

Problem 1-Fix code Submit a working program called: Fluffy.java

Fix the following code so that the program runs. The comments (in red) tell you what the code **should** be doing.

```
import java.util.Scanner;

public class Fluffy
{

    public static void main(String[] args)
    {

        String s="The big fluffy dog likes kibbles and bits";

        //turn the String s into an array of words
        char [] splitS=s.split(" ");

        //check if the last word in splitS is bits (boolean value will be true if the last word is bits)
        boolean check=splitS.length()-1.equals("bits");

        //check length of splitS array
        int len=splitS.length();

        //print out variable len
        System.out.println("Length: %c"+len+);

        //output the third word in the splitS array (should be fluffy)
        System.out.println(s[3]);

        //print out the array splitS
        for (int i=0;i<splitS.length+1; i++)
        {
            System.out.println(splitS[i]);

        }

    }
}
```

Problem 2-Write methods Submit a program called: NumeroDeux.java

Complete the following program by writing in the methods. Make sure that the code matches the sample run and make sure not to hardcode any part of the sample run since we will be using other examples as well. DO NOT MODIFY THE MAIN METHOD-only write in the methods (where it says **write code here**). You will turn in the **whole program**.

```
import java.util.Scanner;
```

```
public class NumeroDeux
```

```
{
    public static int getUserNum(String message)
    {
        //write code here
    }
}
```

```
public static String getUserString(int n)
{
    //write code here
}
```

```
public static boolean checkNumLength(int a, int check1, int check2)
{
    //write code here
}
```

```
public static void main(String[] args) //DO NOT MODIFY
{
```

```
    boolean choice=true;
    int userIn=0;
    while(choice)
    {
        userIn=getUserNum("Enter a number between 4-6:");
        choice=checkNumLength(userIn, 4, 6);
    }
}
```

```
String input=getUserString(userIn);
```

```

choice=true;
int userIn2=0;

while(choice)
{
    userIn2=getUserNum("Enter another number (for substring) between 1-3:");
    choice=checkNumLength(userIn2, 1, 3);
}

System.out.println(input.substring(0,userIn2));

}
}

```

SAMPLE RUN:

```

Enter a number between 4-6:3
Enter a number between 4-6:8
Enter a number between 4-6:4
Enter a word at of length 4: dogg
Enter another number (for substring) between 1-3:3
dog

```

Problem 3-Write a program Submit a program called: Chicken.java

The following program compares the total number of eggs (laid by chickens) that two farmers have. The user of the program types inputs in the form *Chicken dash number of eggs* (for example *Chicken1-4* where Chicken1 is the chicken and 4 is the number eggs separated by a dash.) Define the following methods used in the main.

```

import java.util.Scanner;

public class ClassStuff2
{
    /*This method returns a String array filled with information for one of the farmers. Every element in the array should be in the form shown in the sample run (for example, Chicken1-5). You can assume the user always enters this information separated by a dash.*/
    public static String [] getInput(String message)
    {
        //write code here
    }
}

```

/*This method returns the total number of eggs laid by chickens for a specific farmer. It takes the array of information entered by the user (holding information like Chicken1-2). It returns the total number of eggs.*/

```
public static int totalEggs(String [] s)
{
    //write code here
}

public static void main(String[] args)
{

    String [] farmerOne=getInput("-Info for first farmer:");
    String [] farmerTwo=getInput("-Info for second farmer:");

    int numOne=totalEggs(farmerOne);
    int numTwo=totalEggs(farmerTwo);

    if(numOne<numTwo) /*farmer 2 has more*/
    {
        System.out.println("Farmer 2 has more eggs.");
    }

    else if(numTwo<numOne) /*farmer 1 has more*/
    {
        System.out.println("Farmer 1 has more eggs.");
    }

    else /*same number*/
    {
        System.out.println("They have the same number of eggs.");
    }
}
```

Sample Run:

-Info for first farmer:

Day 1: Chicken1-10

Day 2: Chicken2-10

Day 3: Chicken3-10

-Info for second farmer:

Day 1: Chicken1-4

Day 2: Chicken2-5

Day 3: Chicken3-5

Farmer 1 has more eggs.

Problem 4-Write a program Submit a program called: EvenOdd.java

Lärs wants to create a program that asks a user to first enter an **even** number (continuing to prompt the user for an even number if they do not actually enter an even number).

Afterwards, he wants the user to enter an **odd** number (once again, continuously prompting the user if they do not at first enter an odd number). Once the user has done this, he wants the program to exit saying: *****Thanks! Bye!*****

He wants to use the following methods (no credit if methods are not used):

```
public static int getUserInput(String message)
public static boolean checkIfEven(int n)
```

(Feel free to create any additional helper methods).

Sample Run:

--Enter an even number:

7

That is not even. Enter an even number.

9

That is not even. Enter an even number.

4

--Ok thanks! Now enter an odd number:

4

That is not odd. Enter an odd number.

8

That is not odd. Enter an odd number.

9

****Thanks! Bye!****

Problem 5-Write a program Submit a program called: Movie.java

Johan wants to create a program that allows him to type in 3 movies. All movies entered should start with the letter *m* (case shouldn't matter). If the user enters a movie that does not start with the letter *m*, the program should let the user know that it does not start with the

letter *m* and reprompt for a movie starting with *m*. Once all 3 movies are entered, all movies with titles that have an even length less than 5 should be printed to screen (see sample run).

1. Define the following methods:

/*This method takes the number of movies (given in this problem as 3) and returns a String array filled with the movie information*/

```
public static String [] enterMovies(int numOfMovies)
```

/*This method takes the array of movie info and the max length a movie should be in order to print it out (given in this problem as 5)*/

```
public static void movieLength(String [] s, int n)
```

2. Using the methods you created, create the program. Feel free to make any helper methods.

Sample Run:

```
***Enter a movie that starts with m: movie1
```

```
***Enter a movie that starts with m: Not a Movie
```

```
-That movie doesn't start with m.
```

```
***Enter a movie that starts with m: Movie2
```

```
***Enter a movie that starts with m: mov3
```

Movies titles with an even length less than 5:

```
mov3
```