

Logging In

There are 3 functions commented out in the starter code.

```
//doWhileSample  
//whileSample  
//forLoopSample
```

Each function takes a String as a parameter.

Task 1

Write down the password which needs to be entered into the system

Task 2 - doWhile() sample

Analyse the `doWhileSample` function and answer the following questions:

What is the purpose of the variable <code>count</code> ?	
Describe this code in English: <code>!myPassword.equals(pWord)</code>	
How can the user exit the loop? Think carefully about how to answer this...	
Enter the correct password. Does the loop code run?	

Task 3 - whileSample()

Analyse the `whileSample` function and answer the following questions:

Enter the correct password. Does the loop code run?	
---	--

Task 4 - forloopSample()

What is the purpose of the keyword **break**?

ADAPTATION 1

The `doWhileSample()` function is being adapted.

It will now process **an array** of password samples.

If the user enters any of the correct samples, the user can log in.

The user is given 5 chances.

Here is a sample program:

```
Java
static Scanner kbin = new Scanner(System.in);

public static void main(String[] args){

    //populate a String array with String objects
    String[] passwords = getPasswords();

    //display the passwords - for code testing only!
    displayPasswords(passwords);

    //pass String array reference to doWhileSample
    doWhileSample(passwords);
}

public static String[] getPasswords(){
    //declare a String array and initialise its length
    String[] samples = new String[3];

    for(int i = 0; i<3; i++){
        System.out.println("Enter password # "+(i+1));
        samples[i] = kbin.nextLine();
    }
}

public static void displayPasswords(String[] sampleData){

    //loop through the array, display each String at index i
    for(int i = 0; i<sampleData.length; i++){
        System.out.print(sampleData[i]+"\\t\\t")
    }
}
```

Code the adaptation of the `doWhileSample()` and test it with the above program.

ADAPTATION 2 - ANSI Codes

Use your solution to adaptation 1.

The program uses [ANSI codes](#)

When the user only has 2 chances left, a warning:

Warning! 2 Chances Left

appears.

Mess around with ANSI codes to make the warning stand out.

Go

KEY POINTS

We can use **any** loop structure to solve a given problem.

Commonly however:

For loops are used when we know how many iterations will be required.

While loops are used when we don't know how many iterations will be required.

Do While loops are used when we don't know how many iterations will be required, but there will be at least one!