Core Java - Strings Assignment

Instructions:

Below you will find a series of assignments that you will complete in this exercise. You will need to create a new project folder for this task and name it **Java_Assignment_3_Strings**. You can use the default settings when creating the new project.

For each of the below assignments you will create a new class as specified below and write your code in a **main()** method of that class. You can use the same (default) package when creating each class.

Assignments

1. Write a class named **Assignment1** that declares two strings, “Hello” and (your name) and concatenates them together. Print the result.

2. Write a class named **Assignment2** that uses a 2 if-statement to compare the strings “abc” and “ABC”. The first if-statement should use the double equals (==) operator and the other that uses the .equals() method of one of the strings to compare to the other. Print a message in both if-statements to indicate that the strings are equal

3. Write a class named **Assignment3** that declares 2 strings “abc” and “abc”. The first should use the string literal syntax and the other should use the **new** keyword. Write a comparison operator that prints a message if they are equal or otherwise. (Use an if/else-statement)

4. Write a class named **Assignment4** that declares a string "abcdefghijklmnopqrstuvwxyz". Use the indexOf() method to print the index of "s" and "f".

5. Write a class named **Assignment5** that uses the substring() method to print the last 5 characters of "Hello My Name is Java".

6. Write a class named **Assignment6** that finds the index of the 2nd space character in "Hello My Name is Java"; Print this number. (Hint: you may have to combine two String methods)

7. Write a class named **Assignment7** that declares the string "http://example.com?access_token=EFEc8328h29jndjd02h12$3829&ds98d". Print only the token from the string (the part AFTER the “=”. (Hint: Use the substring() method)

Bonus

8. Write a class named **Assignment13** that checks if two strings are anagrams. Recall that an *anagram* is a word that can be formed by rearranging the letters of another. For example, angel is an anagram of angle and glean. (Hint: You will have to use the string methods that you are familiar with along with one or more loops)

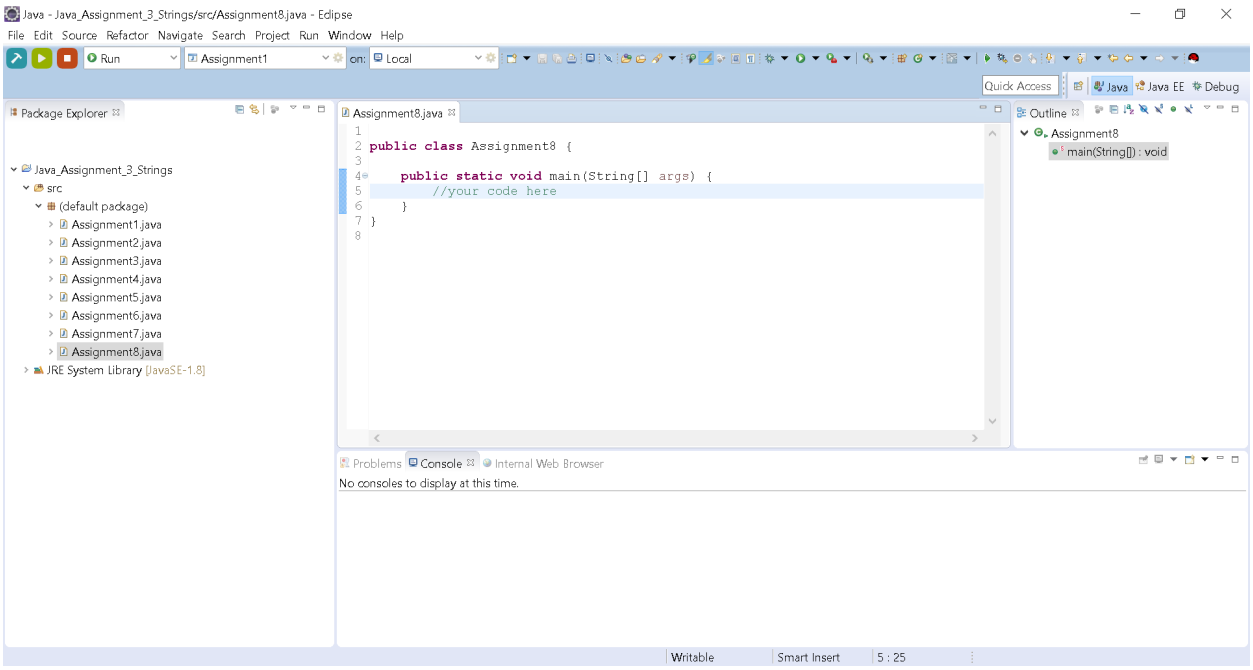
Goal:

The purpose of this assignment is to provide you practice with writing classes that utilize Strings and string-related methods. You will utilize the **main()** method of your classes to execute a particular goal of the assignment.

By the end of this assignment, you will have 1 new project folder with 7+ classes that meet the requirements mentioned above.

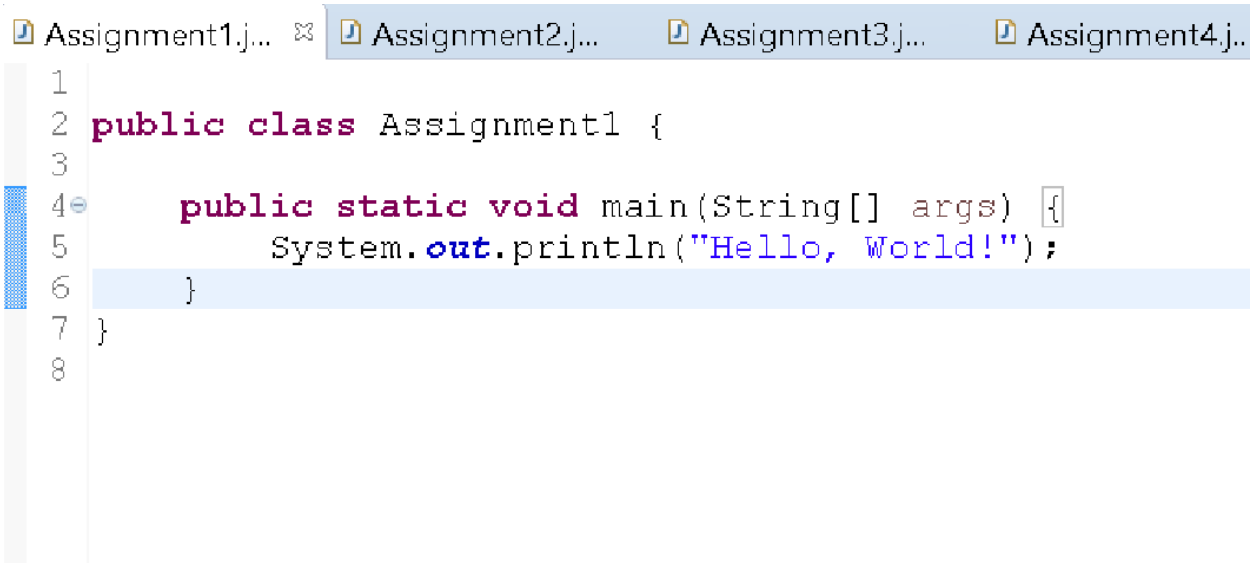
Helpful Notes

Project Structure:



- Your project structure should resemble the above. Note that the name of the project is **Java_Assignment_3_Strings** and all class files are named according to the associated assignment number.

Assignment Classes:



- Notice the indentation of the statement on line 5 to show that it is part of the main() method. Likewise, your code should follow this pattern.