

1. Write a program that prints Hello World! to the console.
Save the program as Hello.java



The screenshot shows an IDE with three tabs: responses.py, Hello.java, and styles.css. The Hello.java file is open, showing the following code:

```
1 public class Hello {  
2     public static void main(String[] args) {  
3         System.out.println("Hello");  
4     }  
5 }  
6
```

The terminal output shows the command to compile and run the program, and the resulting output:

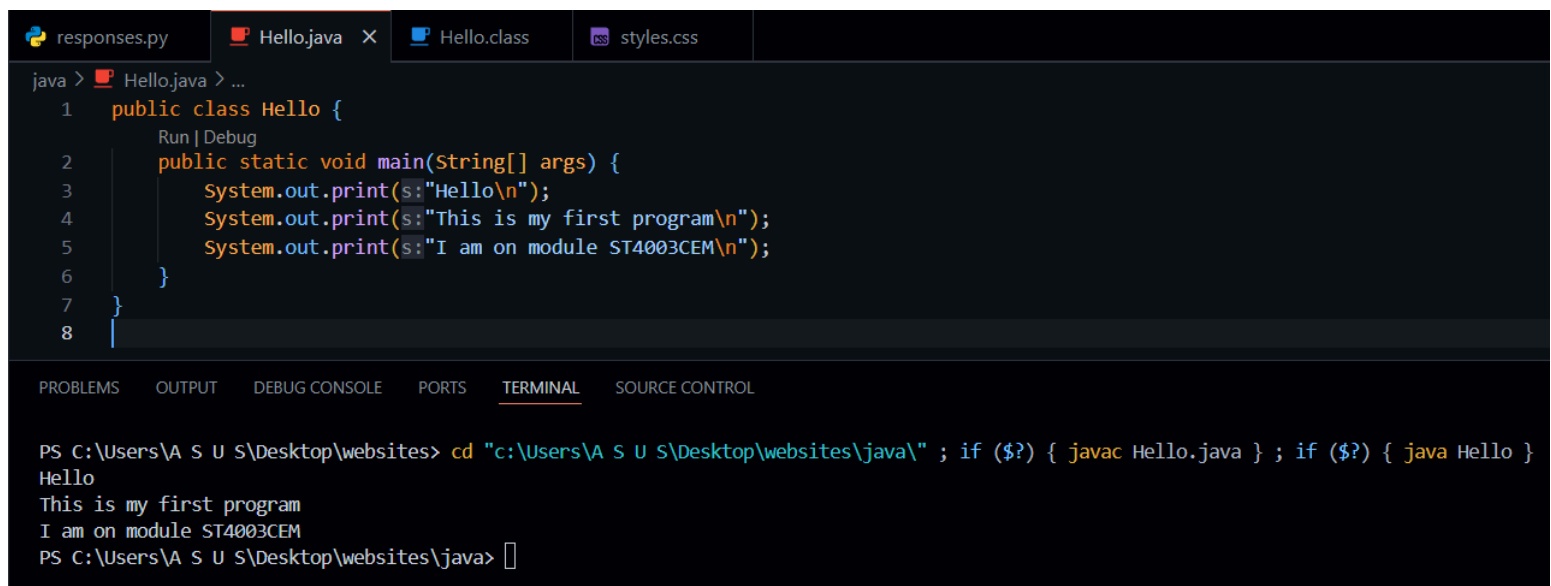
```
PS C:\Users\A S U S\Desktop\websites> cd "c:\Users\A S U S\Desktop\websites\java\" ; if ($?) { javac Hello.java } ; if ($?) { java Hello }  
Hello  
PS C:\Users\A S U S\Desktop\websites\java>
```

2. Extend/Modify the above program to print two additional lines:

This is my first program

I am on module ST4003CEM

System.out.print (not println) and produces the same output.



The screenshot shows the same IDE with an additional tab, Hello.class. The Hello.java file is modified to print three lines:

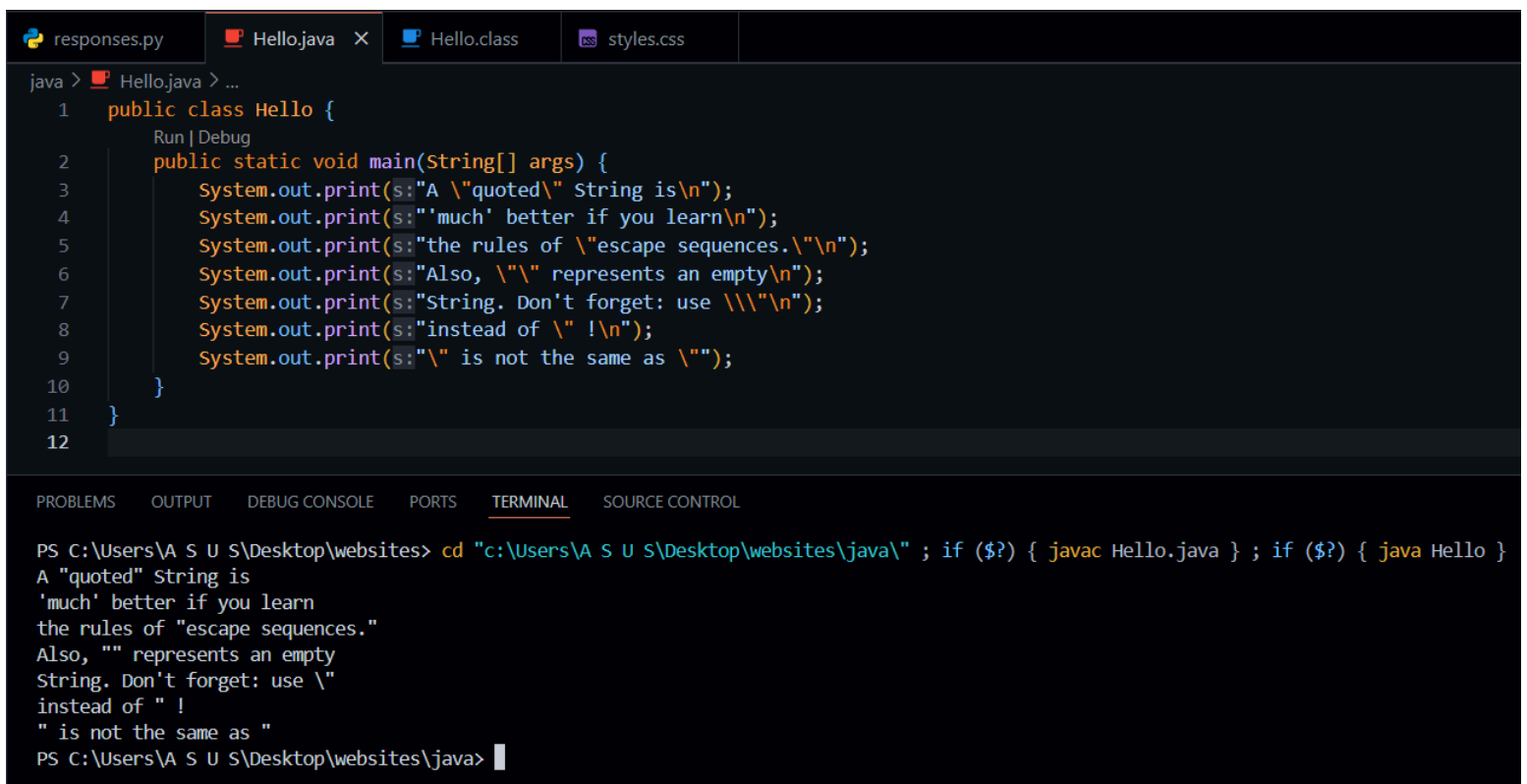
```
1 public class Hello {  
2     public static void main(String[] args) {  
3         System.out.print(s:"Hello\n");  
4         System.out.print(s:"This is my first program\n");  
5         System.out.print(s:"I am on module ST4003CEM\n");  
6     }  
7 }  
8
```

The terminal output shows the command to compile and run the program, and the resulting output:

```
PS C:\Users\A S U S\Desktop\websites> cd "c:\Users\A S U S\Desktop\websites\java\" ; if ($?) { javac Hello.java } ; if ($?) { java Hello }  
Hello  
This is my first program  
I am on module ST4003CEM  
PS C:\Users\A S U S\Desktop\websites\java>
```

3. Write a program that prints the following output:

A "quoted" String is
'much' better if you learn
the rules of "escape sequences."
Also, "" represents an empty
String. Don't forget: use \
instead of " !
" is not the same as "



The screenshot shows an IDE with a dark theme. The top bar has tabs for 'responses.py', 'Hello.java', 'Hello.class', and 'styles.css'. The 'Hello.java' tab is active, showing the following code:

```
1 public class Hello {  
2     public static void main(String[] args) {  
3         System.out.print(s:"A \"quoted\" String is\n");  
4         System.out.print(s:"'much' better if you learn\n");  
5         System.out.print(s:"the rules of \"escape sequences.\\n\"");  
6         System.out.print(s:"Also, \"\" represents an empty\n");  
7         System.out.print(s:"String. Don't forget: use \\\"");  
8         System.out.print(s:"instead of \" !\n");  
9         System.out.print(s:"\" is not the same as \"");  
10    }  
11 }  
12
```

Below the code editor is a panel with tabs for 'PROBLEMS', 'OUTPUT', 'DEBUG CONSOLE', 'PORTS', 'TERMINAL', and 'SOURCE CONTROL'. The 'TERMINAL' tab is active, showing the command prompt output:

```
PS C:\Users\A S U S\Desktop\websites> cd "c:\Users\A S U S\Desktop\websites\java\" ; if ($?) { javac Hello.java } ; if ($?) { java Hello }  
A "quoted" String is  
'much' better if you learn  
the rules of "escape sequences."  
Also, "" represents an empty  
String. Don't forget: use \  
instead of " !  
" is not the same as "  
PS C:\Users\A S U S\Desktop\websites\java>
```

4. WAP that prints the following pattern:

```
*  
**  
***  
****  
*****
```



The screenshot shows an IDE with a dark theme. The top tab bar contains five tabs: 'responses.py', 'Hello.java' (active), 'HelloWorld.java', 'Hello.class', and 'styles.css'. The main editor area displays the following Java code:

```
1 public class Hello {  
    Run | Debug  
2     public static void main(String[] args) {  
3         System.out.print(s:"*\n");  
4         System.out.print(s:**\n");  
5         System.out.print(s:***\n");  
6         System.out.print(s:****\n");  
7         System.out.print(s:*****\n");  
8     }  
9 }  
10
```

Below the editor is a panel with tabs for 'PROBLEMS', 'OUTPUT', 'DEBUG CONSOLE', 'PORTS', 'TERMINAL' (active), and 'SOURCE CONTROL'. The terminal window shows the command prompt output:

```
PS C:\Users\A S U S\Desktop\websites> cd "c:\Users\A S U S\Desktop\websites\java\" ; if ($?) { javac Hello.java } ; if ($?) { java Hello }  
*  
**  
***  
****  
*****  
PS C:\Users\A S U S\Desktop\websites\java>
```

5. WAP prints the following pattern:



The screenshot shows an IDE with a tab for 'Hello.java'. The code in the editor is as follows:

```
1 public class Hello {  
2     Run | Debug  
3     public static void main(String[] args) {  
4         System.out.print(s:"*****\n");  
5         System.out.print(s:"*****\n");  
6         System.out.print(s:"*****\n");  
7         System.out.print(s:"*****\n");  
8         System.out.print(s:"*****\n");  
9     }  
10 }
```

Below the editor, the 'TERMINAL' tab is active, showing the command prompt output:

```
PS C:\Users\A S U S\Desktop\websites> cd "c:\Users\A S U S\Desktop\websites\java\" ; if ($?) { javac Hello.java } ; if ($?) { java Hello }  
*****  
*****  
*****  
*****  
*****  
PS C:\Users\A S U S\Desktop\websites\java>
```

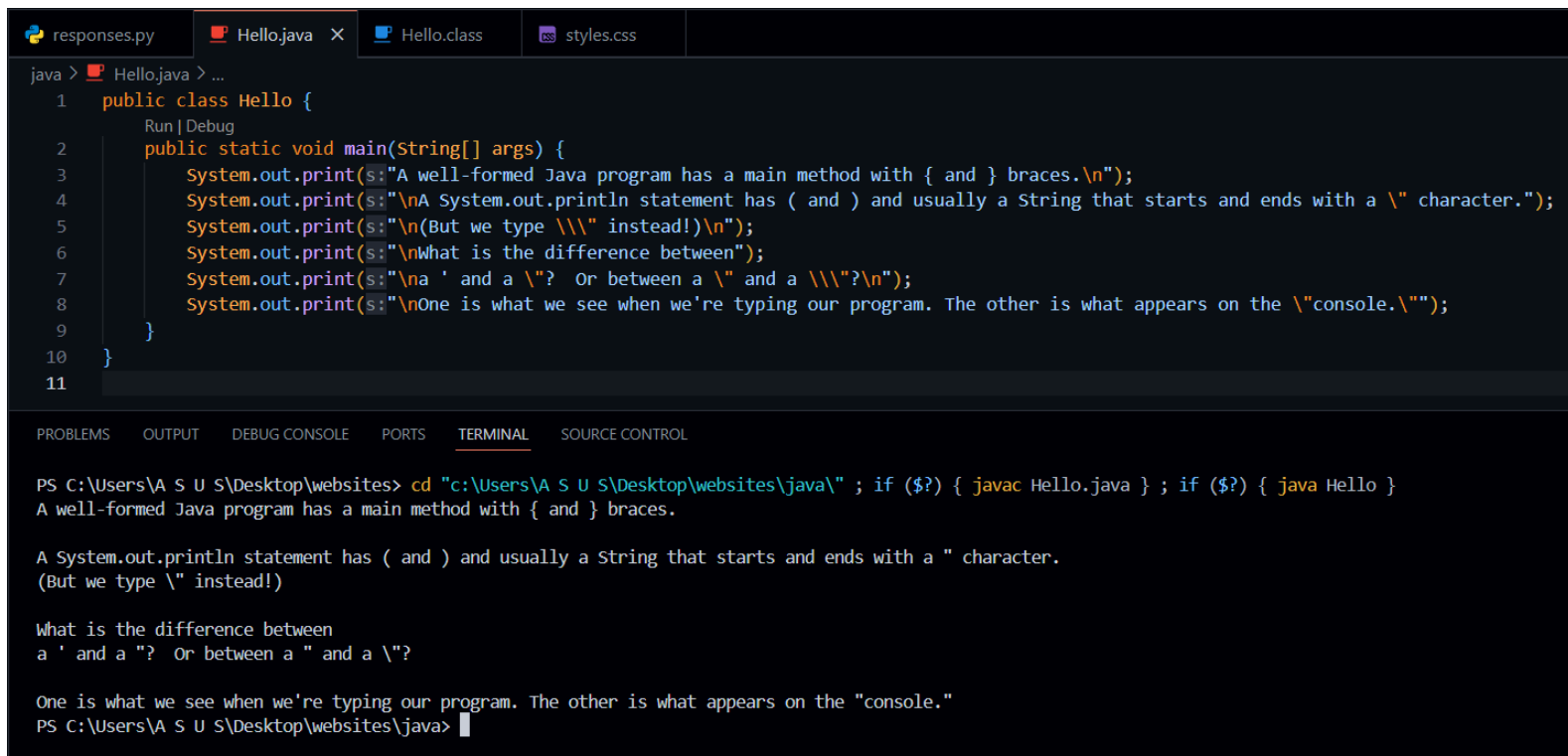
6. WAP that prints the following output:

A well-formed Java program has a main method with { and } braces.

A System.out.println statement has (and) and usually a String that starts and ends with a " character.
(But we type \" instead!)

What is the difference between
a ' and a "? Or between a " and a \"?

One is what we see when we're typing our program. The other is what appears on the "console."



The screenshot shows an IDE with a Java file named `Hello.java` open. The code defines a `Hello` class with a `main` method that prints several lines of text. The IDE's output window shows the exact output of the program, including the escaped backslashes in the strings. The terminal window shows the command to compile and run the program, and the resulting output.

```
java > Hello.java > ...
1 public class Hello {
2     public static void main(String[] args) {
3         System.out.print(s:"A well-formed Java program has a main method with { and } braces.\n");
4         System.out.print(s:"\nA System.out.println statement has ( and ) and usually a String that starts and ends with a \" character.");
5         System.out.print(s:"\n(But we type \" instead!)\n");
6         System.out.print(s:"\nWhat is the difference between");
7         System.out.print(s:"\na ' and a \"? Or between a \" and a \"?");
8         System.out.print(s:"\nOne is what we see when we're typing our program. The other is what appears on the \"console.\"");
9     }
10 }
11
```

PROBLEMS OUTPUT DEBUG CONSOLE PORTS TERMINAL SOURCE CONTROL

```
PS C:\Users\A S U S\Desktop\websites> cd "c:\Users\A S U S\Desktop\websites\java\" ; if ($?) { javac Hello.java } ; if ($?) { java Hello }
A well-formed Java program has a main method with { and } braces.

A System.out.println statement has ( and ) and usually a String that starts and ends with a " character.
(But we type \" instead!)

What is the difference between
a ' and a "? Or between a " and a \"?

One is what we see when we're typing our program. The other is what appears on the "console."
PS C:\Users\A S U S\Desktop\websites\java>
```