

## **PROBLEM STATEMENT:**

Given a time in 12-hour AM/PM format, convert it to military (24-hour) time.

Note: - 12:00:00AM on a 12-hour clock is 00:00:00 on a 24-hour clock.

- 12:00:00PM on a 12-hour clock is 12:00:00 on a 24-hour clock.

### **Example**

- `s = '12:01:00PM'`

Return '12:01:00'.

- `s = '12:01:00AM'`

Return '00:01:00'.

### **Function Description**

Complete the *timeConversion* function with the following parameter(s):

- *string s*: a time in 12 hour format

### **Returns**

- *string*: the time in 24 hour format

### **Input Format**

A single string *s* that represents a time in 12-hour clock format (i.e.: hh:mm:ssAM or hh:mm:ssPM).

### **Constraints**

- All input times are valid

### **Sample Input 0**

```
07:05:45PM
```

### **Sample Output 0**

```
19:05:45
```

## **PROGRAM:**

```
import java.io.*;
import java.math.*;
```

```
import java.security.*;
import java.text.*;
import java.util.*;
import java.util.concurrent.*;
import java.util.function.*;
import java.util.regex.*;
import java.util.stream.*;
import static java.util.stream.Collectors.joining;
import static java.util.stream.Collectors.toList;

class Result {

    /*
     * Complete the 'timeConversion' function below.
     *
     * The function is expected to return a STRING.
     * The function accepts STRING s as parameter.
     */

    public static String timeConversion(String s) {
        // Write your code here
        String period=s.substring(8, 10);
        String hr=s.substring(0, 2);
        int nhr=Integer.valueOf(hr);
        String ms=s.substring(3, 8);
        String hrs;
        if (period.equals("PM")) {
            if (nhr != 12) {
                nhr = nhr + 12;
            }
            hrs = String.valueOf(nhr);
        } else { // AM
            if (nhr == 12) {
                hrs = "00";
            } else {
                hrs = hr; // keep same hour
            }
        }

        return hrs + ":" + ms;
    }
}
```

```
public class Solution {
    public static void main(String[] args) throws IOException {
        BufferedReader bufferedReader = new BufferedReader(new
InputStreamReader(System.in));
        BufferedWriter bufferedWriter = new BufferedWriter(new
FileWriter(System.getenv("OUTPUT_PATH")));

        String s = bufferedReader.readLine();

        String result = Result.timeConversion(s);




        bufferedWriter.write(result);
        bufferedWriter.newLine();

        bufferedReader.close();
        bufferedWriter.close();
    }
}
```

**OUTPUT:**

# Congratulations

You solved this challenge. Would you like to challenge your friends?



Test case 0

Test case 1

Test case 2

Test case 3

Test case 4

Test case 5

Test case 6

Compiler Message

Success

Input (stdin)

1

07:05:45PM

Expected Output

1

19:05:45