Status	Finished
Started	Sunday, 22 September 2024, 7:03 AM
Completed	Sunday, 22 September 2024, 7:24 AM
Duration	21 mins 14 secs

Question **1**

Correct

Marked out of 5.00

You have recently seen a motivational sports movie and want to start exercising regularly. Your coach tells you that it is important to get up early in the morning to exercise. She sets up a schedule for you:

On weekdays (Monday - Friday), you have to get up at 5:00. On weekends (Saturday & Sunday), you can wake up at 6:00. However, if you are on vacation, then you can get up at 7:00 on weekdays and 9:00 on weekends.

Write a program to print the time you should get up.

Input Format

Input containing an integer and a boolean value.

The integer tells you the day it is (1-Sunday, 2-Monday, 3-Tuesday, 4-Wednesday, 5-Thursday, 6-Friday, 7-Saturday). The boolean is true if you are on vacation and false if you're not on vacation.

You have to print the time you should get up.

Example Input:

1 false

Output:

6:00

Example Input:

5 false

Output:

5:00

Example Input:

1 true

Output:

9:00

For example:

Input	Result
1 false	6:00
5 false	5:00
1 true	9:00

Answer: (penalty regime: 0 %)

```
import java.util.Scanner;
public class WakeUpTime {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
// Input: Day of the week (1 to 7) and vacation status (
int day = scanner.nextInt();
boolean onVacation = scanner.nextBoolean();
// Determine the wake-up time based on the day and vacat
String wakeUpTime;
if (onVacation) {
// Vacation schedule
if (day == 1 || day == 7) { // Sunday or Saturday
wakeUpTime = "9:00";
} else { // Weekdays
wakeUpTime = "7:00";
}
} else {
// Regular schedule
if (day == 1 || day == 7) { // Sunday or Saturday
wakeUpTime = "6:00";
} else { // Weekdays
wakeUpTime = "5:00";
}
```

```
}
// Output the wake-up time
System.out.println(wakeUpTime);
scanner.close();
}
}
```

	Input	Expected	Got	
✓	1 false	6:00	6:00	✓
✓	5 false	5:00	5:00	✓
✓	1 true	9:00	9:00	✓

Passed all tests! ✓

Question **2**

Correct

Marked out of 5.00

Consider the following sequence:

1st term: 1

2nd term: 1 2 1

3rd term: 1 2 1 3 1 2 1

4th term: 1 2 1 3 1 2 1 4 1 2 1 3 1 2 1

And so on. Write a program that takes as parameter an integer n and prints the nth terms of this sequence.

Example Input:

1

Output:

1

Example Input:

4

Output:

1 2 1 3 1 2 1 4 1 2 1 3 1 2 1

For example:

Input	Result
1	1
2	1 2 1
3	1 2 1 3 1 2 1
4	1 2 1 3 1 2 1 4 1 2 1 3 1 2 1

Answer: (penalty regime: 0 %)

```
import java.util.Scanner;

public class SequenceGenerator {

    public static String generateTerm(int n) {
                String term = "1";
                for (int i = 2; i <= n; i++) {
            term = term + " " + i + " " + term;
        }

        return term;
    }

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
                int n = scanner.nextInt();
               System.out.println(generateTerm(n));
        scanner.close();
    }
}
```

	Input	Expected	Got	
✓	1	1	1	✓
✓	2	1 2 1	1 2 1	✓

	Input	Expected	Got	
✓	3	1 2 1 3 1 2 1	1 2 1 3 1 2 1	✓
✓	4	1 2 1 3 1 2 1 4 1 2 1 3 1 2 1	1 2 1 3 1 2 1 4 1 2 1 3 1 2 1	✓

Passed all tests! ✓

Question **3**

Correct

Marked out of 5.00

Consider a sequence of the form 0, 1, 1, 2, 4, 7, 13, 24, 44, 81, 149...

Write a method program which takes as parameter an integer n and prints the nth term of the above sequence. The nth term will fit in an integer value.

Example Input:

5

Output:

4

Example Input:

8

Output:

24

Example Input:

11

Output:

149

For example:

Input	Result
5	4
8	24
11	149

Answer: (penalty regime: 0 %)

```
import java.util.Scanner;

public class CustomSequence {

        public static int findNthTerm(int n) {
        if (n == 1) return 0;
        if (n == 2 || n == 3) return 1;

        int t1 = 0, t2 = 1, t3 = 1;

        for (int i = 4; i <= n; i++) {
            int nextTerm = t1 + t2 + t3;
            t1 = t2;
            t2 = t3;
            t3 = nextTerm;
        }

        return t3;
    }

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

                int n = scanner.nextInt();

                System.out.println(findNthTerm(n));

        scanner.close();
    }
}

```

	Input	Expected	Got	
✓	5	4	4	✓
✓	8	24	24	✓
✓	11	149	149	✓

Passed all tests! ✓

◄ Lab-02-MCQ

Jump to...

Lab-03-MCQ ►