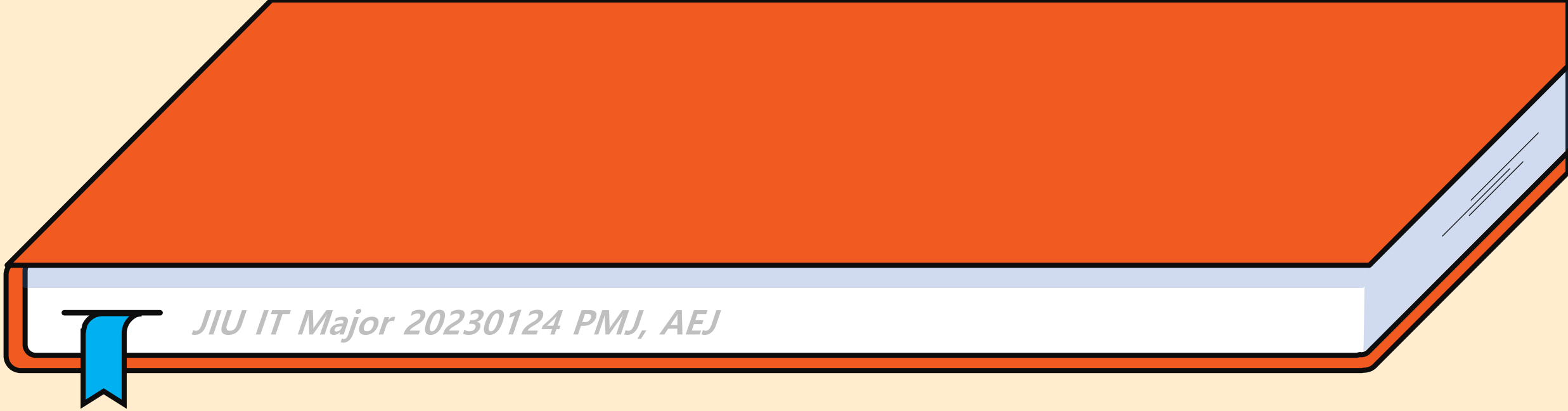


JIU IT Major 20230124 PMJ, AEJ

Object-Oriented Programming

Practice Week2

What is wrong in the following code?

```
if (score >= 60) System.out.println("D");
else if (score >= 70) System.out.println("C");
else if (score >= 80) System.out.println("B");
else if (score >= 90) System.out.println("A");
else System.out.println("F");
```

Question 2. Boolean

x	y	x || y	x && y
true	true	true	true
true	false	true	false
false	true	true	false
false	false	false	false

True = 1, False = 0

x	y	x || y	x && y
true	true	true	true
true	false	true	false
false	true	true	false
false	false	false	false

1 1

1 0

0 1

0 0

3.10.1 Assuming that x is 1,
show the result of the following Boolean expressions:

(true) && (3 > 4)

3.10.1 Assuming that x is 1,
show the result of the following Boolean expressions:

1 && 0

(true) && (3 > 4)

True = 1, False = 0

x	y	x || y	x && y
true	true	true	true
true	false	true	false
false	true	true	false
false	false	false	false

1 1

1 0

0 1

0 0

3.10.1 Assuming that x is 1,
show the result of the following Boolean expressions:

```
!(x > 0) && (x > 0)
```

3.10.1 Assuming that x is 1,
show the result of the following Boolean expressions:

0 && 1

!(x > 0) && (x > 0)

3.10.1 Assuming that x is 1,
show the result of the following Boolean expressions:

$$(x > 0) \,||\, (x < 0)$$

3.10.1 Assuming that x is 1,
show the result of the following Boolean expressions:

1 || 0

(x > 0) || (x < 0)

True = 1, False = 0

x	y	x || y	x && y
true 1	true 1	true	true 1
true 1	false 0	true	false
false 0	true 1	true	false
false 0	false 0	false 0	false

3.10.1 Assuming that x is 1,
show the result of the following Boolean expressions:

```
(x != 0) || (x == 0)
```

3.10.1 Assuming that x is 1,
show the result of the following Boolean expressions:

```
1          ||          0
(x != 0) || (x == 0)
```

3.10.1 Assuming that x is 1,
show the result of the following Boolean expressions:

```
(x >= 0) || (x < 0)
```

3.10.1 Assuming that x is 1,
show the result of the following Boolean expressions:

1 || 0

(x >= 0) || (x < 0)

3.10.1 Assuming that x is 1,
show the result of the following Boolean expressions:

```
(x != 1) == !(x == 1)
```

3.10.1 Assuming that x is 1,
show the result of the following Boolean expressions:

0 == 0

(x != 1) == !(x == 1)

Question 3. Switch

```
System.out.println("switch(1)");
switch(1) {
    case 1:
        System.out.println("one");
        break;
    case 2:
        System.out.println("two");
        break;
    case 3:
        System.out.println("three");
        break;
    default:
        System.out.println("default");
        break;
}
```

Question 3. Switch

```
System.out.println("switch(1)");
switch(1) {
    case 1:
        System.out.println("one");
        break;
    case 2:
        System.out.println("two");
        break;
    case 3:
        System.out.println("three");
        break;
    default:
        System.out.println("default");
        break;
}
```

```
switch(1)
one
```

Question 1. If-elseIf

```
public class Ifelseif {
    Park-Minjoo
    public static void main(String[] args){
        Scanner scan = new Scanner(System.in);
        double score = scan.nextDouble();

        if (score >= 90) System.out.println("A");
        else if (score >= 80) System.out.println("B");
        else if (score >= 70) System.out.println("C");
        else if (score >= 60) System.out.println("D");
        else System.out.println("F");
    }
}
```

Question 2. Switch

```
switch (date) {
    case 0:  numberStr = "Sunday";    break;
    case 1:  numberStr = "Monday";    break;
    case 2:  numberStr = "Tuesday";   break;
    case 3:  numberStr = "Wednesday"; break;
    case 4:  numberStr = "Thursday";  break;
    case 5:  numberStr = "Friday";    break;
    case 6:  numberStr = "Saturday";  break;
    default: numberStr = "Not a valid day"; break;
}
```

Question 3. Day Of the Week

```
Enter today's day: 1 ↵Enter
Enter the number of days elapsed since today: 3 ↵Enter
Today is Monday and the future day is Thursday
```

1 = Monday
1 (date) + 3(elapsed) = 4
4 % 7 = 4

4 = Thursday

Question 3. Day Of the Week

```
Enter today's day: 1 ↵ Enter
Enter the number of days elapsed since today: 3 ↵ Enter
Today is Monday and the future day is Thursday
```

```java
int futureDate = (date + elapsed) % 7;
String dayoftheWeek = "";

switch(date){
    case 0: dayoftheWeek = "Sunday"; break;
    case 1: dayoftheWeek = "Monday"; break;
    case 2: dayoftheWeek = "Tuesday"; break;
    case 3: dayoftheWeek = "Wednesday"; break;
    case 4: dayoftheWeek = "Thursday"; break;
    case 5: dayoftheWeek = "Friday"; break;
    case 6: dayoftheWeek = "Saturday"; break;
}
```

1 = Monday
1 (date) + 3(elapsed) = 4
4 % 7 = 4

4 = Thursday

Question 3. Day of the Week

```
Enter today's day: 1 ↵ Enter
Enter the number of days elapsed since today: 3 ↵ Enter
Today is Monday and the future day is Thursday
```

```
if (futureDate == 0){
    System.out.printf("Todays is %s and the future day is Sunday", dayoftheWeek);
}else if(futureDate == 1){
    System.out.printf("Todays is %s and the future day is Monday", dayoftheWeek);
}else if(futureDate == 2){
    System.out.printf("Todays is %s and the future day is Tuesday", dayoftheWeek);
}else if(futureDate == 3){
    System.out.printf("Todays is %s and the future day is Wednesday", dayoftheWeek);
}else if(futureDate == 4){
    System.out.printf("Todays is %s and the future day is Thursday", dayoftheWeek);
}else if(futureDate == 5){
    System.out.printf("Todays is %s and the future day is Friday", dayoftheWeek);
}else if(futureDate == 6) {
    System.out.printf("Todays is %s and the future day is Saturday", dayoftheWeek);
}
```

Question 4. Palindrome

```
Enter a three-digit integer: 121 ↵ Enter
121 is a palindrome
```

```
// Test for palindrome
int digit1 = [redacted];
int remaining = [redacted];
int digit3 = [redacted];
```

-> 1 (hundreds place)

-> 21

-> 1 (ones place)

Question 4. Palindrome

```
Enter a three-digit integer: 121 ↵ Enter
121 is a palindrome
```

```
// Test for palindrome
int digit1 = (int)(number / 100);
int remaining = number % 100;
int digit3 = (int)(remaining % 10);
```

-> 1 (hundreds place)

-> 21

-> 1 (ones place)

Question 4. Palindrome

```
// Test for palindrome
int digit1 = (int)(number / 100);
int remaining = number % 100;
int digit3 = (int)(remaining % 10);
```

-> 1 (hundreds place)

-> 21

-> 1 (ones place)

```
// Display result
System.out.println(
        number + (                          ? " is a " : " is not a ") + "palindrome");
```

Question 4. Palindrome

```
// Test for palindrome
int digit1 = (int)(number / 100);
int remaining = number % 100;
int digit3 = (int)(remaining % 10);
```

1 (hundreds place)
==
1 (ones place)

```
// Display result
System.out.println(
        number + (                          ? " is a " : " is not a ") + "palindrome");
```

Question 4. Palindrome

```
// Test for palindrome
int digit1 = (int)(number / 100);
int remaining = number % 100;
int digit3 = (int)(remaining % 10);
```

1 (hundreds place)
==
1 (ones place)

```
// Display result
System.out.println(
        number + ((digit1 == digit3) ? " is a " : " is not a ") + "palindrome");
```

Question 5. 12-hour clock

```
// Obtain the total seconds since midnight, Jan 1, 1970
long totalSeconds = [redacted];

// Compute the current second in the minute in the hour
long currentSecond = totalSeconds % 60;

// Obtain the total minutes
long totalMinutes = [redacted];

// Compute the current minute in the hour
long currentMinute = [redacted];

// Obtain the total hours
long totalHours = [redacted];

// Compute the current hour
long currentHour = totalHours % 24;
currentHour = currentHour + offset;
```

Question 5. 12-hour clock

```
// Obtain the total seconds since midnight, Jan 1, 1970
long totalSeconds = totalMilliseconds / 1000;

// Compute the current second in the minute in the hour
long currentSecond = totalSeconds % 60;

// Obtain the total minutes
long totalMinutes =                    ;

// Compute the current minute in the hour
long currentMinute =                    ;

// Obtain the total hours
long totalHours =                    ;

// Compute the current hour
long currentHour = totalHours % 24;
currentHour = currentHour + offset;
```

1 sec = 1000 millis

Question 5. 12-hour clock

```
// Obtain the total seconds since midnight, Jan 1, 1970
long totalSeconds = totalMilliseconds / 1000;

// Compute the current second in the minute in the hour
long currentSecond = totalSeconds % 60;

// Obtain the total minutes
long totalMinutes =                    ;

// Compute the current minute in the hour
long currentMinute =                    ;

// Obtain the total hours
long totalHours =                    ;

// Compute the current hour
long currentHour = totalHours % 24;
currentHour = currentHour + offset;
```

Current = total % 60

Question 5. 12-hour clock

```
// Obtain the total seconds since midnight, Jan 1, 1970
long totalSeconds = totalMilliseconds / 1000;

// Compute the current second in the minute in the hour
long currentSecond = totalSeconds % 60;

// Obtain the total minutes
long totalMinutes = [redacted];

// Compute the current minute in the hour
long currentMinute = [redacted];

// Obtain the total hours
long totalHours = [redacted];

// Compute the current hour
long currentHour = totalHours % 24;
currentHour = currentHour + offset;
```

1 min = 60 s

Question 5. 12-hour clock

```
// Obtain the total seconds since midnight, Jan 1, 1970
long totalSeconds = totalMilliseconds / 1000;

// Compute the current second in the minute in the hour
long currentSecond = totalSeconds % 60;

// Obtain the total minutes
long totalMinutes = totalSeconds / 60;

// Compute the current minute in the hour
long currentMinute = [redacted];

// Obtain the total hours
long totalHours = [redacted];

// Compute the current hour
long currentHour = totalHours % 24;
currentHour = currentHour + offset;
```

1 min = 60 s

Question 5. 12-hour clock

```
// Obtain the total seconds since midnight, Jan 1, 1970
long totalSeconds = totalMilliseconds / 1000;

// Compute the current second in the minute in the hour
long currentSecond = totalSeconds % 60;

// Obtain the total minutes
long totalMinutes = totalSeconds / 60;

// Compute the current minute in the hour
long currentMinute =                    ;

// Obtain the total hours
long totalHours =                       ;

// Compute the current hour
long currentHour = totalHours % 24;
currentHour = currentHour + offset;
```

Question 5. 12-hour clock

```
// Obtain the total seconds since midnight, Jan 1, 1970
long totalSeconds = totalMilliseconds / 1000;

// Compute the current second in the minute in the hour
long currentSecond = totalSeconds % 60;

// Obtain the total minutes
long totalMinutes = totalSeconds / 60;

// Compute the current minute in the hour
long currentMinute = totalMinutes % 60;

// Obtain the total hours
long totalHours =                    ;

// Compute the current hour
long currentHour = totalHours % 24;
currentHour = currentHour + offset;
```

1 hour = 60 m

Question 5. 12-hour clock

```
// Obtain the total seconds since midnight, Jan 1, 1970
long totalSeconds = totalMilliseconds / 1000;

// Compute the current second in the minute in the hour
long currentSecond = totalSeconds % 60;

// Obtain the total minutes
long totalMinutes = totalSeconds / 60;

// Compute the current minute in the hour
long currentMinute = totalMinutes % 60;

// Obtain the total hours
long totalHours = totalMinutes / 60;

// Compute the current hour
long currentHour = totalHours % 24;
currentHour = currentHour + offset;
```

1 hour = 60 m

Question 5. 12-hour clock

```
// Obtain the total seconds since midnight, Jan 1, 1970
long totalSeconds = totalMilliseconds / 1000;

// Compute the current second in the minute in the hour
long currentSecond = totalSeconds % 60;

// Obtain the total minutes
long totalMinutes = totalSeconds / 60;

// Compute the current minute in the hour
long currentMinute = totalMinutes % 60;

// Obtain the total hours
long totalHours = totalMinutes / 60;

// Compute the current hour
long currentHour = totalHours % 24;
currentHour = currentHour + offset;
```

1 hour = 60 m

Question 5. 12-hour clock

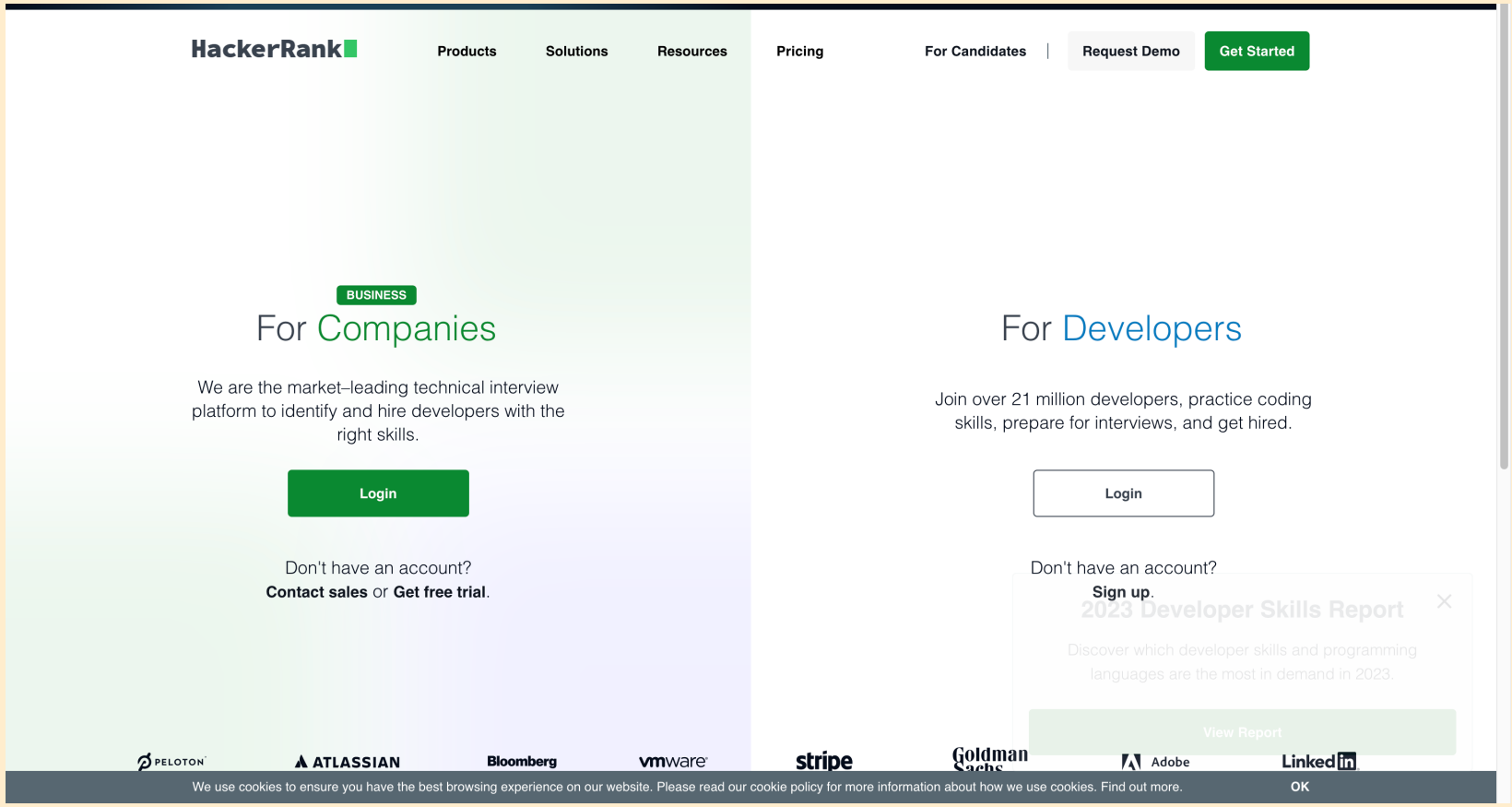
```
// Display results
System.out.println(
        "Current time is " + ((currentHour > 12) ?
                currentHour) + ":" + currentMinute + ":" + currentSecond +
                ((currentHour > 12) ? " PM" : " AM"));
```

If (currentHour) > 12

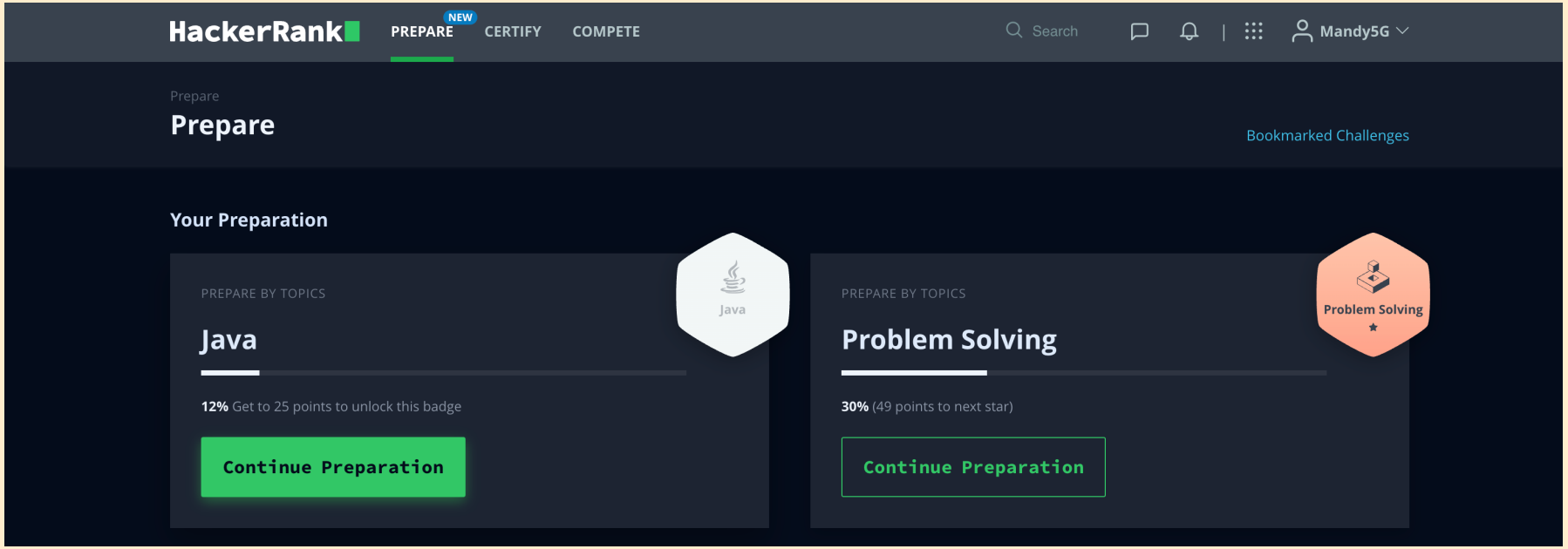
else If (currentHour) <= 12

Question 3. Stdin and Stdout

https://www.hackerrank.com/



Question 3. Stdin and Stdout



https://www.hackerrank.com/

Introduction to Java

The term Java from Java island...

JIU IT Major 20230124 PMJ, AEJ