Pages 48-49 and 58 - 61 *Java Programming A Comprehensive Introduction*

http://docs.oracle.com/javase/tutorial/uiswing/components/dialog.html

Define

Symbol (%) in java programming basic arithmetic- a software component or part of a program that contains one or more routines. One or more independently developed modules make up a program. An enterprise-level software application may contain several different modules, and each module serves unique and separate business operations.

20 % 3 = 2

Operators

++: add one

--: subtract one

Boolean type- two values (usually denoted true and false), intended to represent the truth values of logic and Boolean algebra.

Logical Operators

& - bitwise AND

│ - bitwise OR

^ - bitwise XOR

││   - logical OR;

&& - logical AND

Difference between & and &&?

& evaluates both sides of the operations. And && evaluates the left side of the operations, if it's true, it continues and evaluates the right side.

Logic Table

P	Q	P &Q	P │ Q	P ^ Q	!P
FALSE	FALSE	False	False	True	True
TRUE	FALSE	False	True	False	False
FALSE	TRUE	False	True	False	True
TRUE	TRUE	True	True	False	False

Programming Assignments- Attach Snipping Photos of source code and output for each separate programming task.

Task 1-

Write a program that converts Seconds into minutes.

Use the **new** Scanner (System.in); method and allow the user to input the number of seconds she wants to convert.

Use the modulus arithmetic symbol for part of the calculation/conversion.

Output= Show number of seconds input, conversion to minutes, and remaining seconds.

```java
 7    package javaapplication1;
 8
 9 ⊟  import java.util.Scanner;
10
11    public class JavaApplication1 {
12
13 ⊟      /**
14          * @param args the command line arguments
15          */
16 ⊟      public static void main(String[] args) {
17              // TODO code application logic here
18
19              Scanner input = new Scanner(System.in);
20
21              boolean b1, b2;
22
23              int seconds;
24              int minutes;
25              int hours;
26              int years;
27
28              //user enter seconds
29              System.out.print("Enter you Seconds: ");
30              seconds = input.nextInt();
31
32
33              minutes = (seconds / 60) % 60;
34
35              seconds = seconds % 60;
36
37              System.out.println(minutes + " minutes " + seconds + " seconds");
38
39          }
40
41    }
42
```

JavaApplication1 ❯   ◑ main ❯

Output - JavaApplication1 (run) ✕

```
run:
Enter you Seconds: 350
5 minutes 50 seconds
BUILD SUCCESSFUL (total time: 14 seconds)
```

Task 2- We are going to enhance our Seconds to minutes conversion program from Task 1.

Use the **new** Scanner (System.in); method and allow the user to input the number of seconds she wants to convert.

We now want the program to calculate hours, minutes, seconds up to one day.

We will use **(if)** conditional statements and code blocks to create the program. (No need for if/else or anything else we have not covered in class to this point)

Output= Show number of seconds input and a conversion to Hours, minutes and remaining seconds. Or have a print line for any input in seconds > than 1 day.

```java
6
7    package javaapplication1;
8
9    import java.util.Scanner;
10
11   public class JavaApplication1 {
12
13       /**
14        * @param args the command line arguments
15        */
16       public static void main(String[] args) {
17           // TODO code application logic here
18
19           Scanner input = new Scanner(System.in);
20
21           int seconds, minutes, hours, years;
22
23           System.out.print("Enter your seconds: ");
24           seconds = input.nextInt();
25
26           if (seconds < 86400) {
27               hours = seconds / 3600;
28               minutes = (seconds / 60) % 60;
29               seconds = seconds % 60;
30               System.out.println(hours + " hours " + minutes + " minutes " + seconds + " seconds ");
31           }
32
33           if (seconds >= 86400) {
34               System.out.println("The seconds are exceed a day's seconds");
35           }
36       }
37   }
38
```

Output - JavaApplication1 (run)

```
run:
Enter your seconds: 86411
The seconds are exceed a day's seconds
BUILD SUCCESSFUL (total time: 4 seconds)
```

```
6
7     package javaapplication1;
8
9  ⊟ import java.util.Scanner;
10
11    public class JavaApplication1 {
12
13 ⊟      /**
14         * @param args the command line arguments
15         */
16 ⊟      public static void main(String[] args) {
17             // TODO code application logic here
18
19             Scanner input = new Scanner(System.in);
20
21             int seconds, minutes, hours, years;
22
23             System.out.print("Enter your seconds: ");
24             seconds = input.nextInt();
25
26             if (seconds < 86400) {
27                 hours = seconds / 3600;
28                 minutes = (seconds / 60) % 60;
29                 seconds = seconds % 60;
30                 System.out.println(hours + " hours " + minutes + " minutes " + seconds + " seconds ");
31             }
32
33             if (seconds >= 86400) {
34                 System.out.println("The seconds are exceed a day's seconds");
35             }
36         }
37     }
38    |
```

Output - JavaApplication1 (run) ✕

```
run:
Enter your seconds: 86411
The seconds are exceed a day's seconds
BUILD SUCCESSFUL (total time: 4 seconds)
```

Task 3-

boolean leapyear = (year % 4 == 0 && year % 100 != 0)

(year % 400 = 0);

Use the statements above in your program.

Connect the two statements above with a short-circuit operator and create a program where the user can enter an integer year using the the **new** Scanner (System.in); method and be returned a true if the year is a leap year or false if the year is not a leap year.

Sample Outputs

"Enter Year" // User prompt

2000

"2000 is a leap year? true" // Output

"Enter Year" // User prompt

190

"190 is a leap year? false" // Output

```java
 6
 7     package javaapplication1;
 8
 9   □ import java.util.Scanner;
10
11     public class JavaApplication1 {
12
13   □      /**
14           * @param args the command line arguments
15           */
16   □      public static void main(String[] args) {
17              // TODO code application logic here
18
19              Scanner input = new Scanner(System.in);
20
21              int year;
22
23              boolean leapyear;
24
25              System.out.print("Enter Year: ");
26              year = input.nextInt();
27
28              leapyear = ((year % 4 == 0) && (year % 100 != 0)) || (year % 400 == 0);
29
30              if(leapyear) {
31                  System.out.println(year + " is a leap year? true");
32              } else {
                     System.out.println(year + " is a leap year? false");
34              }
35
36          }
37     }
38
```

JavaApplication1 ⟩ main ⟩ if (leapyear) else ⟩

Output - JavaApplication1 (run) ※

```
run:
Enter Year: 2000
2000 is a leap year? true
BUILD SUCCESSFUL (total time: 4 seconds)
```

```java
7     package javaapplication1;
8
9   □ import java.util.Scanner;
10
11    public class JavaApplication1 {
12
13  □     /**
14          * @param args the command line arguments
15          */
16  □     public static void main(String[] args) {
17              // TODO code application logic here
18
19              Scanner input = new Scanner(System.in);
20
21              int year;
22
23              boolean leapyear;
24
25              System.out.print("Enter Year: ");
26              year = input.nextInt();
27
28              leapyear = ((year % 4 == 0) && (year % 100 != 0)) || (year % 400 == 0);
29
30              if(leapyear) {
31                  System.out.println(year + " is a leap year? true");
32              } else {
33                  System.out.println(year + " is a leap year? false");
34              }
35
36          }
37      }
38
```

JavaApplication1 》 ⑪ main 》

Output - JavaApplication1 (run) ✖

```
run:
Enter Year: 190
190 is a leap year? false
BUILD SUCCESSFUL (total time: 12 seconds)
```