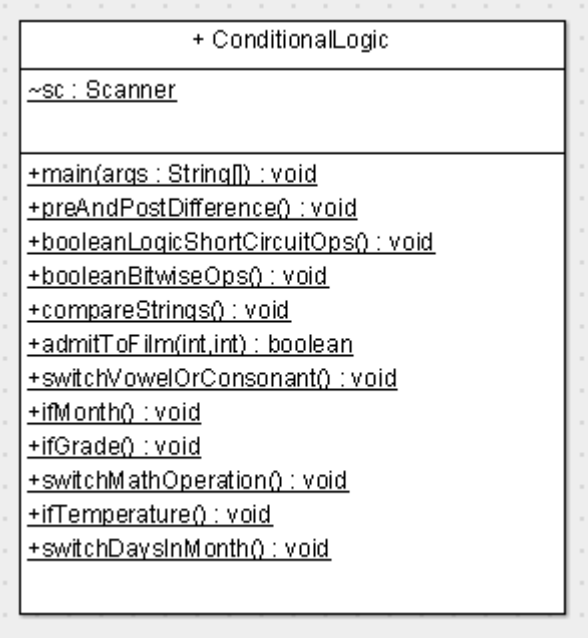

1. preAndPostDifference()
    a. declare two local variables x and y initialised to 5 and 10 respectively
    b. output the value of x preceded by "The value of x is "
    c. output the value of ++x preceded by "The value of ++x is "
        i. System.out.println("The value of ++x is " + ++x);
    d. output the value of x++ preceded by "The value of x++ is "
        i. System.out.println("The value of x++ is " + x++);
    e. output the value of x preceded by "The value of x is "
    f. the values output for x should be 5, 6, 6 and 7
    g. do the exact same for y, except use the prefix/postfix decrement operator i.e. --
    h. the values output for y should be 10, 9, 9 and 8.
2. booleanLogicShortCircuitOps()
    a. declare a boolean variable b and initialise it to false
    b. declare an int variable x and initialise it to 3
    c. code the following :
        if(x<0 && (b=true)) {}
        System.out.println(b);
    d. code the following :
        if(x>0 || (b=true)) {}
        System.out.println(b);
    e. What was the value of "b" in the above println() statements and why?
3. booleanBitwiseOps()
    a. the logic is identical to the booleanLogicShortCircuitOps() except that the operators are NOT the short-circuit ones i.e. use & instead of && and | instead of ||
    b. What was the value of "b" in the println() statements and why?.
4. compareStrings()
    a. using Scanner, get in two strings from the user, namely s1 and s2. Input "Sean" for both.
    b. compare s1 and s2 in two ways:
        i. firstly, using == i.e. System.out.println(s1 == s2);
        ii. secondly, using the equals() method i.e. System.out.println(s1.equals(s2));
        iii. why is the output different?

5. admitToFilm()
    a. in main() ask the user for the certification for a film (an int) and the age of the person seeking admittance (also an int)
    b. pass these two values down to admitToFilm()
    c. in admitToFilm() do the following:
        i. using if statement, check if the customer is old enough to see the film. Return true or false depending on the result.
    d. back in main, output the boolean returned from admitToFilm()

6. switchVowelOrConsonant()
    a. ask the user to enter a character (use *sc.next().charAt(0);* to retrieve the char from the keyboard. Store the char in a variable named *letter.*
    b. using an "if" statement, make sure *letter* is valid i.e. a-z or A-Z. Hint: use single quotes for characters. The statement is something like:
        if (((java code here) && (java code here)) || ((java code here) && (java code here)))
    c. if *letter* is valid:
        i. switch on *letter*
            1. if it's a vowel (uppercase or lowercase) then output the letter entered followed by the text " is a vowel".
            2. otherwise output the letter entered followed by the text " is a consonant".
        ii. change the uppercase letter case labels to use their Unicode/ASCII values i.e. instead of *case 'A':* use *case 65:*
        iii. move the *default* section to the top of the *switch* statement. Does it still work properly?
    d. if *letter* is invalid, output an error message that includes the erroneous *letter*
7. ifMonth()
    a. prompt the user to enter a month in the range 1..12
    b. initialise *month* with the users input
    c. using an *if-else if-else* structure, output the text for the month entered i.e. if the user entered 1 output "January"; if the value is out of range, output the erroneous value followed by the text " is out of range".
    d. instead of using integer literals e.g. 1, 2, 3 etc.. use constants e.g. JAN, FEB, MAR. To do this, declare constants: *final int JAN=1, FEB=2, MAR=3;*
8. ifGrade()
    a. ask the user to enter a *mark* in the range 0..100
    b. using an *if-else if-else* structure do the following:
        i. if the *mark* is out of range, output an error with the erroneous value in the message
        ii. otherwise:
            1. if the *mark* is 70..100 output "A"
            2. if the *mark* is 60..69 output "B"
            3. if the *mark* is 50..59 output "C"
            4. if the *mark* is 40..49 output "D"
            5. otherwise output "Fail"

9. switchMathOperation()
    a. initialise two local variables: *answer* is a *double* (0.0) and *operationOK* is a *boolean* (*true*).
    b. ask the user to input two *double* numbers, *num1* and *num2* (*sc.nextDouble()*).
    c. ask the user to input a character; store in a variable named *operation*.
    d. *switch* on *operation*:
        i. if it's a '+' add *num1* and *num2* together and store the sum in *answer*
        ii. if it's a '-' subtract *num2* from *num1* and store the result in *answer*
        iii. if it's a '*' multiply *num1* and *num2* together and store the product in *answer*
        iv. if it's a '/' divide *num1* by *num2* and store the result in *answer*
        v. any other character: output "Unknown mathematical operator " and append the char in question to the error message. Also, set *operationOK* to *false*
    e. after the *switch*, if *operationOK* is *true*, output the *answer*. Therefore, if we input an invalid mathematical operation, no *answer* will be output.
10. ifTemperature()
    a. declare a local variable *temperature* and initialise it to 0
    b. declare the following constants (*final int*): COLD is 0; MILD is 15; WARM is 20; VERY_WARM is 25 and HOT is 30.
    c. ask the user to enter a temperature (an *int*)
    d. using an *if-else if-else* structure and the constants declared, code the following:
        i. if the *temperature* is <=0 output "cold"
        ii. if the *temperature* is between 1..14 output "a little cold but ok"
        iii. if the *temperature* is between 15..19 output "mild"
        iv. if the *temperature* is between 20..24 output "warm"
        v. if the *temperature* is between 25..29 output "very warm"
        vi. any *temperature* >30 output "hot"

11. switchDaysInMonth()
    a. declare a local variable *numDays* and initialise it to 0
    b. declare constants for the months of the year e.g. *final int JAN=1, FEB=2, MAR=3;*
    c. ask the user to enter a month (1..12) and store in a local variable namely *month*
    d. switch on *month*:
        i. if it's January, March, May, July, August, October or December then set *numDays* to 31 and then *break*
        ii. if its April, June, September or November then set *numDays* to 30 and then *break*
        iii. if its February we need to figure out if the year is a leap year:
            1. ask the user to enter a year and store in a local variable namely *year*
            2. if *year* is a multiple of 400 (use % operator) OR
               if *year* is a multiple of 4 AND **not** a multiple of 100
               then it's a leap year i.e. set *numDays* to 29
               otherwise set *numDays* to 28