Introduction to Java
Unit 2.   Programs

You should start a new BlueJ project for each new unit. And since this is a new unit, start a new project.

To do programs 1 to 4, you need to know how to write if and if/else statements. You should also know about the && logical operator for problem 4.

1. Write a program that asks the user to indicate how many kangaroos they want. If they want less than 10 kangaroos, then charge them $9.25 each. If they order 10 or more kangaroos then each kangaroo only costs $7.
> For example, if they want 2 kangaroos then the program should print:
>> 2 kangaroos cost a total of $18.5   (don't worry about the missing zero)
> If they want 15 kangaroos then the program should print:
>> 15 kangaroos cost a total of $105

2. A person enters the number of hours worked in a week and their hourly wage. The program calculates their paycheck according to the following formula:
> They get paid their hourly wage for the first 40 hours. They get double pay for any hours over 40. The number of hours is an integer; hourly wage is a double.
> For example, if they worked 20 hours at $9.5 per hour then program prints $190
> If they worked 45 hours at $8 then they get $400 (40 hours * 8 + 5 hours * 16)

3. A person enters 4 numbers representing the coordinates of two points and the program prints out the slope. If it is a vertical line the program prints out "The slope is undefined."
> For example, if the user enters 4 and 6 as the x and y coordinates for the first point and 3.5 and 8 as the x and y coordinates for the second point, then the program should print:
> The slope is -4.0
If the user enters 7.5 and 12 as the x and y coordinates for the first point and 7.5 and 15 as the x and y coordinates for the second point, then the program should print:
> The slope is undefined

4. A person enters the price of a product (a double) and then enters their id number (an int). If the id is between 200 (inclusive) and 400 (inclusive), then deduct 10% from the price. Here are two examples.
> If the user enters 18.25 for the price and 377 for their id, display "Price: 16.425" (Don't worry about rounding the answer off to two decimal places.)
> If the user enters 53.99 for the price and 888 for their id, display "Price: 53.99"

To do programs 5 to 9, you need to know the methods of the Math class discussed on pages 4 and 5 of the notes.

5.   Write a program where the user enters a number and the program displays the square root of the number. For example, if the user enters a 7, then the output should be something like this:

>   The square root of 7 is 2.6457513110645907

6.   Write a program where the user enters the length of the hypotenuse and one leg of a right triangle. The program calculates the length of the missing leg. If the user enters a number for the leg that is greater than or equal to the hypotenuse, then program simply prints "Invalid inputs."

For example, if the user says the hypotenuse is 10.0 inches long and a leg is 8.0 inches long, the program should say the remaining leg is 6.0 inches long. If the user says the hypotenuse is 12 inches long and a leg is 12 inches long, then the program prints "Invalid inputs."

7. Write a program where the user enters two numbers, *a* and *b*, and prints out the value of a^b. Then it should ask for another two numbers and repeat the process. For example:

>   Enter two numbers:
>> *User enters 6 and 5.5*
>   Displays:
>> 6^5.5 = 19047.232239881992
>   Enter two numbers:
>> *User enters 100 and -2*
>   Displays:
>> 100^-2 = 1.0E-4
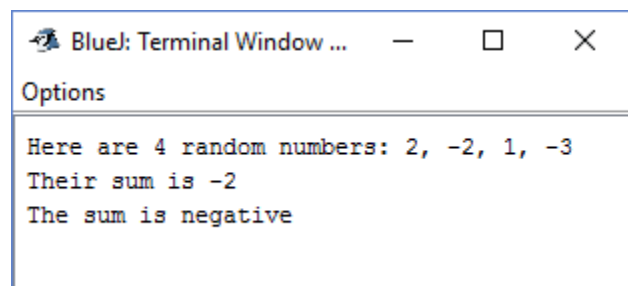
8. Write a program that uses Math.random to generate four random integers between -3 and 3 (inclusive at both ends) and stores them in four variables. Display the four numbers and their sum. If the sum is less than zero then display "Negative" else display "Not Negative." Below is one possible result.

BlueJ: Terminal Window ...

Options

```
Here are 4 random numbers: 2, -2, 1, -3
Their sum is -2
The sum is negative
```

9.   Write a program that generates two random integers *a* and *b*. Each random integer is between 1 and 10. Then the program asks the user what the sum of those two numbers are. The user enters a number and the program indicates if the answer is right or wrong.

   For example, if the two random numbers are 5 and 8

The program prints	5 + 8 = ?
The user enters	*13*
The program prints	Correct! 5 + 8 equals 13

  On the other hand, suppose the two random numbers are 7 and 1

The program prints	7 + 1 = ?
The user enters	*25*
The program prints	Wrong 7 + 1 does not equal 25

The final set of programs cover if / else if statements as well as everything else that was covered in unit two.

10.   Write a program that lets the user enter a numerical grade and displays the corresponding letter grade. For example, if you enter an 83, the program says "You got a B."
Be sure to use an *if* followed by a few *else-if's* and NOT a series of just *if* statements.

11. Write a program that generates three random integers between 5 and 9 (inclusive at both ends) and displays the three numbers. Then, if the three numbers match display the phrase "Three of a kind." If there is only a pair of matching numbers, then display "Pair." Finally, if all three are different then display "All different." You should use && and ||.

12.   The local salamander store is having a sale. The first one costs $3, the second costs $2, and every salamander after that only costs $1.50.
   Write a program where the user enters the number of salamanders and the program displays the total cost. Assume the user only enters positive integer values.

   For example, three salamanders cost $6.50. Four salamanders cost $8. Five salamanders cost $9.50 and so on.

13.   Write a program to calculate the cost of mailing a package. The user enters the package's weight in pounds and its volume in cubic inches. The cost is determined according to these guidelines:

- If it weighs less than 1 pound, it costs $2.
- If it weighs 1 pound or more, and less than 50 pounds, then it costs $2.50 per pound.
- If it weighs more 50 pounds or more, then it costs $3 per pound.

- If its size is more than 1,000 cubic inches, there is an extra fee of 5 cents for every cubic inch over 1,000.
- If its size is between 100 and 1,000 cubic inches (inclusive), then there is a flat $5 additional fee.
- If its size is less than 100 cubic inches, there is no additional fee.
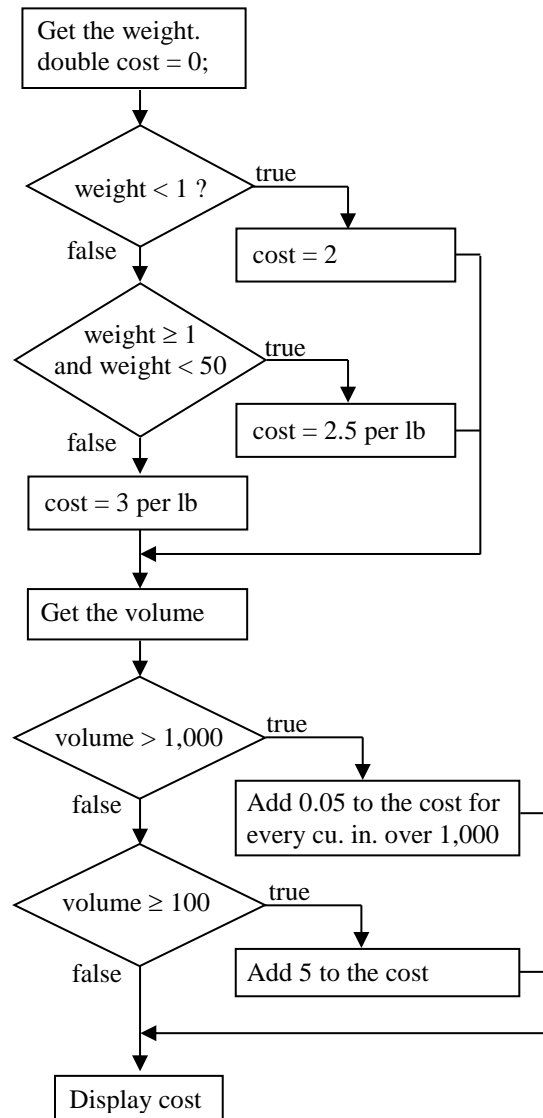
For example, if the user says the package weighs 43 pounds and its volume is 1,200 cubic inches, then the total cost is $117.50 (43 times 2.5 + 200 times 0.05). It is ok that your program will actually display the cost as 117.5

If the user says the package weights 0.48 pounds and its volume is 37 cubic inches, the total cost is $2 (2 for the weight and nothing for the volume).

```
Get the weight.
double cost = 0;

        ↓
    weight < 1 ?  ──true──→  cost = 2
        │ false
        ↓
    weight ≥ 1
  and weight < 50 ──true──→  cost = 2.5 per lb
        │ false
        ↓
   cost = 3 per lb
        ↓
   Get the volume
        ↓
   volume > 1,000 ──true──→  Add 0.05 to the cost for
        │ false               every cu. in. over 1,000
        ↓
   volume ≥ 100 ──true──→  Add 5 to the cost
        │ false
        ↓
   Display cost
```

14. Write a program that calculates the area of any triangle given the lengths of its three sides. Use Heron's formula

$$area = \sqrt{s(s - a)(s - b)(s - c)}$$

where s is half the triangle's perimeter and a, b, and c are the lengths of the sides. Assume the user enters positive numbers for the sides but they may not always make a triangle. If any of the factors (s-a, s-b, s-c) are negative or zero then it is not a triangle.

Here's some inputs to test your program with.

- If a = 3, b = 5, c = 4 then the program should display: The area is 6
- If a = 13, b = 5, c = 12 then the program should display: The area is 30
- If a = 6.5, b = 3.1, c = 7.2 then the program should display: The area is 10.074998759305137
- If a = 15, b = 6, c = 7 then the program should display: Not a triangle

15.   Write a very simple version of 21. The program consists of four steps:

(1) At the start it generates two random integers between 1 and 10 and displays a message like this:

"You have a 3 and a 8 for a total of 11."

"Enter a 1 if you want another card or any other number for no card."

(2) If the user enters a 1, then generate another random integer between 1 and 10 and add it to the total. The display a message like this:
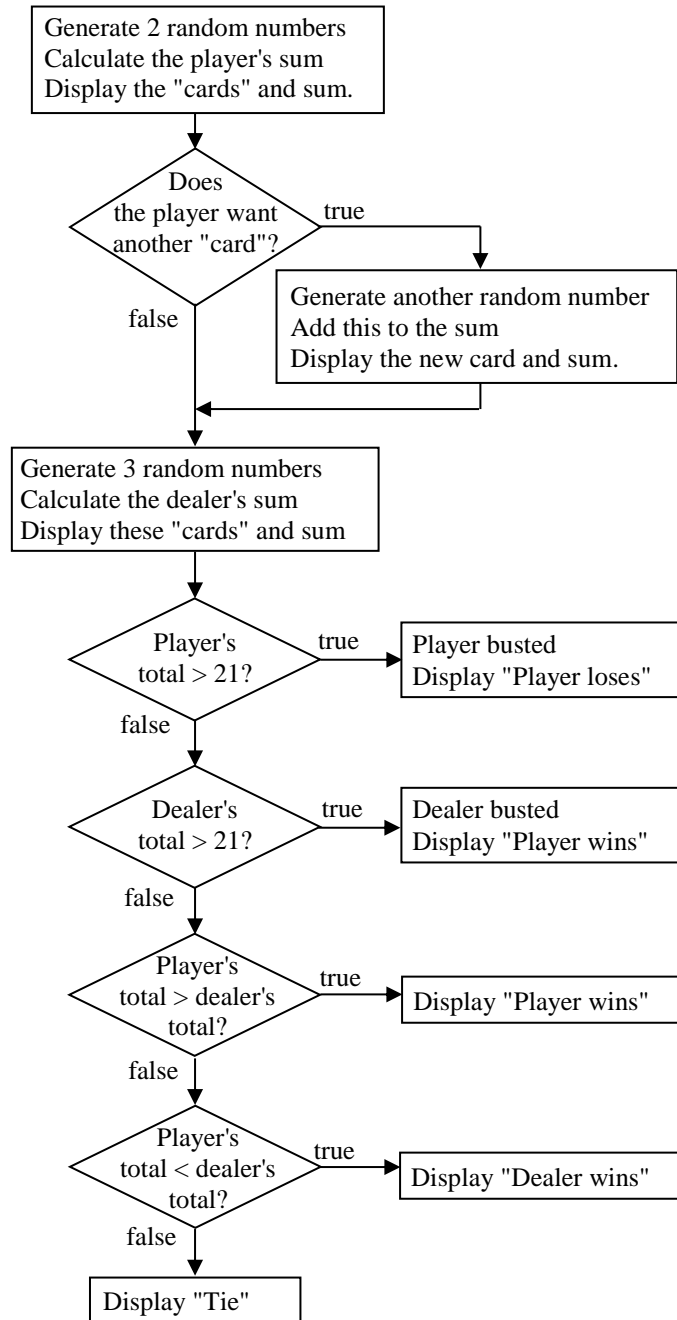
"You got a 5 for a total of 16."
If the user enters any number other than a one, do not give them another card.

(3) Generate three random integers between 1 and 10 and display a message like this:

"The dealer has a 6, 3, and 6 for a total of 15."

(4) Display a message that says either "you win," "you lose," or "tie." The rules of 21 apply:

- If the user got more than 21, then he/she loses (regardless of what the dealer gets).
- If the dealer gets more than 21 (and the user has 21 or less), the dealer loses.
- Otherwise, high score wins. A tie is a tie.

```
┌─────────────────────────────────┐
│ Generate 2 random numbers       │
│ Calculate the player's sum      │
│ Display the "cards" and sum.    │
└─────────────────────────────────┘
              │
              ▼
        ╱─────────────╲        true   ┌──────────────────────────────────┐
       ╱ Does          ╲ ──────────▶  │ Generate another random number   │
       ╲ the player want╱             │ Add this to the sum              │
        ╲ another "card"?            │ Display the new card and sum.    │
         ╲─────────────╱              └──────────────────────────────────┘
           │ false
           ▼
┌─────────────────────────────────┐
│ Generate 3 random numbers       │
│ Calculate the dealer's sum      │
│ Display these "cards" and sum   │
└─────────────────────────────────┘
              │
              ▼
        ╱─────────────╲   true    ┌──────────────────────────┐
       ╱ Player's      ╲ ───────▶ │ Player busted            │
       ╲ total > 21?   ╱          │ Display "Player loses"   │
        ╲─────────────╱           └──────────────────────────┘
           │ false
           ▼
        ╱─────────────╲   true    ┌──────────────────────────┐
       ╱ Dealer's      ╲ ───────▶ │ Dealer busted            │
       ╲ total > 21?   ╱          │ Display "Player wins"    │
        ╲─────────────╱           └──────────────────────────┘
           │ false
           ▼
        ╱─────────────╲   true    ┌──────────────────────────┐
       ╱ Player's      ╲ ───────▶ │ Display "Player wins"    │
       ╲ total > dealer's        └──────────────────────────┘
        ╲ total?       ╱
         ╲─────────────╱
           │ false
           ▼
        ╱─────────────╲   true    ┌──────────────────────────┐
       ╱ Player's      ╲ ───────▶ │ Display "Dealer wins"    │
       ╲ total < dealer's        └──────────────────────────┘
        ╲ total?       ╱
         ╲─────────────╱
           │ false
           ▼
┌─────────────────┐
│ Display "Tie"   │
└─────────────────┘
```

16. This is a rather challenging problem but a good one. Write a program that solves equations of the following form by using the quadratic formula: $\quad ax^2 + bx + c = 0$

Assume that a and b are not both zero. The program asks the user to enter values for a, b, and c. The program will display one of the following responses:

$x^2 - 3x - 10 = 0$ solutions -2 and +5
$x^2 + x + 0.25 = 0$ solution -0.5
$x^2 + 3 = 0$ no solution

         No real solution

         The solution is _____. (If there is only one solution.)

         The solutions are _____ and _____. (If there are two solutions.)

Hint 1. You need to use the quadratic formula.

Hint 2. Create a variable that represents the discriminant. If the discriminant is negative, then there is no solution. If the discriminant is zero, there is exactly one solution. And if it's positive then there are two solutions.

Here are some possible results.

Enter a	Enter a	Enter a
1	2	5
Enter b	Enter b	Enter b
-4	7	3
Enter c	Enter c	Enter c
4	-15	60
There is one solution: 2.0	There are two solutions: 1.5, -5.0	There is no solution.