revisit pg 119, complete checkpoint 3.3 - 3.7
revisit pg 121, complete checkpoint 3.8 - 3.11
revisit pg 127, complete checkpoint 3.12 - 3.13
~~~~~~

The if Statement

```
import java.util.Scanner;

//this program demonstrates the if statement and the scanner utility

public class AverageScore
{
   public static void main(String [] args)
   {

   double score1,    //score #1
          score2,    //score #2
          score3,    //score #3
          average;   //average score

          //create a scanner object to read input
          Scanner keyboard = new Scanner(System.in);

          System.out.println("This program averages 3 test scores.");

          //Get the first score
          System.out.print("Enter score #1: ");
          score1 = keyboard.nextDouble();

          //get second score
          System.out.print("Enter score #2: ");
          score2 = keyboard.nextDouble();

          //get the third score
          System.out.print("Enter score #3: ");
          score3 = keyboard.nextDouble();

          //calculate and display average score
          average = (score1 + score2 + score3) / 3.0;
          System.out.println("The average is " + average);

          //if the average is higher than 95, congratulate the user.
          if (average > 95)
            System.out.println("That's a great score!");
```
~~~~~~

```
   }
}
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~
```

if else Statements

```
import java.util.Scanner;

//this program demonstrates the if-else statement

public class Division
{
   public static void main(String [] args)
   {

   double number1, number2;   //division operands
   double quotient;           //result of division

   //create scanner object for keyboard input
   Scanner keyboard = new Scanner(System.in);

   //get the first number
   System.out.print("Enter a number: ");
      number1 = keyboard.nextDouble();

   //get the second number
   System.out.print("Enter another number: ");
      number2 = keyboard.nextDouble();

   if (number2 == 0)
      {
         System.out.println("Division by zero is not possible.");
         System.out.println("Please run the program again and ");
         System.out.println("enter a number other than zero");
      }

      else
      {
         quotient =  number1 / number2;

         System.out.print("The quotient of " + number1);
         System.out.print(" divided by " + number2);
         System.out.println(" is " + quotient);
      }
```

```
   }
}
```

~~~~~~
~~~~~~~~~~~~~~~

Nested if else Statements

```
import java.util.Scanner;

public class NestedDecisionStructure
{
   public static void main(String[] args)
   {

   Scanner keyboard = new Scanner(System.in);

   int testScore;

   System.out.print("What is your test score? ");
      testScore = keyboard.nextInt();

   if (testScore < 60)
   {
      System.out.println("Your grade is F.");
   }
   else
   {
      if (testScore < 70)
      {
         System.out.println("Your grade is D.");
      }
      else
      {
         if (testScore < 80)
         {
            System.out.println("Your grade is C.");
         }
         else
         {
            if (testScore < 90)
            {
               System.out.println("Your grade is B.");
            }
            else
            {
               if (testScore < 100)
```

```
               {
                  System.out.println("Your grade is A.");
               }
            }
         }
      }


   }
}
```

```
import java.util.Scanner;

//this program demonstrates a nested if statement

public class LoanQualifier
{
   public static void main(String[] args)
   {

   double salary;
   double yearsOnJob;

   //create scanner object for keyboard input
   Scanner keyboard = new Scanner(System.in);

   //get users salary
   System.out.print("Enter your annual salary: ");
   salary = keyboard.nextDouble();

   //get number of years at current job
   System.out.print("Enter the number of years " +
                    "at your current job: ");
   yearsOnJob = keyboard.nextDouble();

   //determine if qualified
   if (salary >= 50000)
   {
         if (yearsOnJob >= 2)
         {
            System.out.println("You qualify for the loan.");
         }

         else
```

```
        {
           System.out.println("You must have been on your current job " +
                              "for at least two years to qualify.");
        }
   }

   else
   {
      System.out.println("You must earn at least $50,000 per year to " +
                         "qualify.");
   }

   }
}
```

~~~~~~

if else if Statements

```
import javax.swing.JOptionPane;

/*
   this program asks the user to enter a numeric test score and displays
   a letter grade for the score. the program uses an if-else-if statement
   to determine the letter grade.
*/
public class TestResults
{
   public static void main(String[] args)
   {

   int testScore;    //numeric test score
   String input;     //to hold user's input

   //get the numeric test score
   input = JOptionPane.showInputDialog("Enter your numeric test score " +
                                       "and I will tell you the grade: ");
   testScore = Integer.parseInt(input);

   //display the grade
   if (testScore < 60)
      JOptionPane.showMessageDialog(null, "Your grade is F.");
   else if (testScore < 70)
      JOptionPane.showMessageDialog(null, "Your grade is D.");
    else if (testScore < 80)
```
~~~~~~

```
      JOptionPane.showMessageDialog(null, "Your grade is C.");
    else if (testScore < 90)
      JOptionPane.showMessageDialog(null, "Your grade is B.");
    else if (testScore < 100)
      JOptionPane.showMessageDialog(null, "Your grade is A.");
    //trailing else used to catch errors
    else
      JOptionPane.showMessageDialog(null, "Invalid score.");

   System.exit(0);

   }
}
```

~~~~~~
~~~~~~~~~~~~~~~~~~~~~

Logical Operators

&& - AND - Connects two boolean expressions into one. Both expressions must be true for the overall expression to be true.

|| - OR - Connects two boolean expressions into one. One or both expressions must be true for the overall expression to be true. it is only necessary for one to be true, and it does not matter which one.

! - NOT - The ! operator reverses the truth of a boolean expression. If it is applied to an expression that is true, the operator returns false. If it is applied to an expression that is false, the operator returns true.

EX.

```
x > y && a < b  ~~~~~~~~~ Is x greater than y AND less than b?
x == y || x == z ~~~~~~~~ Is x equal to y OR is x equal to z?
! (x > y) ~~~~~~~~~~~~~~~ Is the expression x > y NOT true?
```

Precedence of all operators so far

Order of Precedence
Operators
Description

1
- (unary negation) !
Unary negation, logical NOT

2
* / %
Multiplication, division, modulus

3
+ -
Addition, subtraction

4
< > <= >=
Less than, greater than, less than or equal to, greater than or equal to

5
== !=
Equal to, not equal to

6
&&
Logical AND

7
||
Logical OR

8
= += -= *= /= %=
Assignment and combined assignment
~~~~~~
~~~~~~~~~~~~
Logical AND

```
import java.util.Scanner;

//this program demonstrates the logical && operator

public class LogicalAnd
{
   public static void main(String[] args)
   {

   double salary;       //annual salary
   double yearsOnJob;   //years at current job

   //create scanner class
   Scanner keyboard = new Scanner(System.in);

   //get users annual salary
   System.out.print("Enter your annual salary: ");
   salary = keyboard.nextDouble();

   //get users number of years on the job
   System.out.print("Enter the number of years at the current job: ");
   yearsOnJob = keyboard.nextDouble();

      //determine if user qualifies
      if (salary >= 50000 && yearsOnJob >= 2)
      {
         System.out.println("You qualify for the loan.");
      }
      else
      {
         System.out.println("You do not qualify for the loan.");
      }

   }
}
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~
```

Logical OR

```
import java.util.Scanner;

//this program demonstrates the logical || operator

public class LogicalOr
{
```

```
   public static void main(String[] args)
   {

   double salary;       //annual salary
   double yearsOnJob;   //years on job

   //create scanner class
   Scanner keyboard = new Scanner(System.in);

   //get users annual salary
   System.out.print("Enter your annual salary: ");
   salary = keyboard.nextDouble();

   //get users years on job
   System.out.print("Enter number of years on the job: ");
   yearsOnJob = keyboard.nextDouble();

      //determine qualification
      if (salary >= 50000 || yearsOnJob >= 2)
      {
         System.out.println("You Qualify for the loan.");
      }
      else
      {
         System.out.println("You do not qualify");
      }
   }
}
```

~~~~~~

Comparing Strings

You can compare String objects to string literals.

```
        if (name1.equals("Mark"))
```

To determine whether two strings are not equal apply ! operator

```
        if (!name1.equals("Mark"))
```

```
        StringReference.compareTo(OtherString)
```

StringReference is a variable that references a String object, and OtherString is either another variable that references a String object or a string literal. The method returns an integer value that can be used in the following manner:

If the method's return value is negative, the string referenced by
~~~~~~

StringReference (the   calling object) is less than the OtherString argument.

If the method's return value is 0, the two strings are equal.

If the method's return value is postive, the string referenced by StringReference (the   calling object) is greater than the OtherString argument.

EX.

```
//statement uses compareTo method to compare two strings.
if (name1.compareTo(name2) == 0)
        System.out.println("The names are the same.");

//compares the string referenced by name1 to the string literal "Joe".
if (name1.compareTo("Joe") == 0)
        System.out.println("The names are the same.");
```

~~~~~~

String Compare Using Equal Method

```
//this program correctly comares two string objects using the equals method
public class StringCompare
{
   public static void main(String[] args)
   {

   String name1 = "Mark",
          name2 = "Mark",
          name3 = "Mary";

          //compare "Mark" and "Mark"
          if (name1.equals(name2))
          {
            System.out.println(name1 + " and " + name2 +
                              " are the same.");
          }
          else
          {
            System.out.println(name1 + " and " + name2 +
                              " are NOT the same.");
          }

          //compare "Mark" and "Mary"
          if (name1.equals(name3))
          {
            System.out.println(name1 + " and " + name3 +
                              " are the same.");
          }
```
~~~~~~

```
          else
          {
            System.out.println(name1 + " and " + name3 +
                                " are NOT the same.");
          }

   }
}
```

~~~~~~

String Compare Using compareTo method

```
//This program compares two String objects using the compareTo method
public class StringCompareTo
{
   public static void main(String[] args)
   {

      String name1 = "Mary",
             name2 = "Mark";

      //compare "Mary" and "Mark"

      if (name1.compareTo (name2) < 0)
      {
         System.out.println(name1 + " is less than " + name2);
      }
      else if (name1.compareTo (name2) == 0)
      {
         System.out.println(name1 + " is equal to " + name2);
      }
      else if (name1.compareTo (name2) > 0)
      {
         System.out.println(name1 + " is greater than " + name2);
      }
   }
}
```

~~~~~~

Variable Scope

```
import java.util.Scanner;

/*
This program demonstrates how variables can be declared in various
locations throughout the program
```

```
*/

public class VariableScope
{
   public static void main(String[] args)
   {

   //create scanner class
   Scanner keyboard = new Scanner(System.in);

   //get user's first name
   System.out.print("Enter your first name: ");
   String firstName;
   firstName = keyboard.nextLine();

   //get user's last name
   System.out.print("Enter your last name: ");
   String lastName;
   lastName = keyboard.nextLine();

   //display message
   System.out.println("Hello " + firstName + " " + lastName + ".");

   }
}
```