Base Conversion

One algorithm for converting a base 10 number to base b involves repeated division by the base b. Initially one divides the number by b. The remainder from this division is the units digit (the rightmost digit) in the base b representation of the number (it is the part of the number that contains no powers of b). The quotient is then divided by b on the next iteration. The remainder from this division gives the next base b digit from the right. The quotient from this division is used in the next iteration. The algorithm stops when the quotient is 0. Note that at each iteration the remainder from the division is the next base b digit from the right—that is, this algorithm finds the digits for the base b number in reverse order.

Here is an example for converting 30 to base 4:

```
              quotient       remainder
              --------       ---------
   30/4 =          7              2
   7/4  =          1              3
   1/4  =          0              1
```

The answer is read bottom to top in the remainder column, so 30 (base 10) = 132 (base 4).

Think about how this is recursive in nature: If you want to convert x (30 in our example) to base b (4 in our example), the rightmost digit is the remainder x % b. To get the rest of the digits, you perform the same process on what is left; that is, you convert the quotient x / b to base b. If x / b is 0, there is no rest; x is a single base b digit and that digit is x % b (which also is just x).

The file *BaseConversion.java* contains the shell of a method *convert* to do the base conversion and a main method to test the conversion. The convert method returns a string representing the base b number, hence for example in the base case when the remainder is what is to be returned it must be converted to a String object. This is done by concatenating the remainder with a null string. The outline of the convert method is as follows:

```
public static String convert (int num, int b)
{
   int quotient;  // the quotient when num is divided by base b
   int remainder; // the remainder when num is divided by base b

   quotient = _____________________________;

   remainder = _____________________________;
```

```
        if ( ____________________________________ )  //fill in base case
        {
             return ("" + ___________________________________ );
        }
        else
        {
          // Recursive step: the number is the base b representation of
            // the quotient concatenated with the remainder

       return ( ____________________________________________________ );

        }
     }
```

Fill in the blanks above (for now don't worry about bases greater than 10), then in BaseConversion.java complete the method and main. Main currently asks the user for the number and the base and reads these in. Add a statement to print the string returned by convert (appropriately labeled).

Test your function on the following input:

- Number: 89 Base: 2 ---> should print 1011001
- Number: 347 Base: 5 ---> should print 2342
- Number: 3289 Base: 8 ---> should print 6331

Improving the program: Currently the program doesn't print the correct digits for bases greater than 10. Add code to your convert method so the digits are correct for bases up to and including 16.

```
// ****************************************************************
//   BaseConversion.java
//
//   Recursively converts an integer from base 10 to another base
// ****************************************************************

import java.util.Scanner;
```

```
public class BaseConversion
{
    public static void main (String[] args)
    {
      int base10Num;
      int base;

      Scanner scan = new Scanner(System.in);

      System.out.println ();
      System.out.println ("Base Conversion Program");
      System.out.print ("Enter an integer: ");
      base10Num = scan.nextInt();
      System.out.print ("Enter the base: ");
      base = scan.nextInt();

      // Call convert and print the answer

    }
    // -----------------------------------------------
    //   Converts a base 10 number to another base.
    // -----------------------------------------------
    public static String convert (int num, int b)
    {
      int quotient;  // the quotient when num is divided by base b
      int remainder; // the remainder when num is divided by base b
    }

}
```