1. Implement the user-defined ***fibonacci( )*** method shown below into a class called ***moreFibonacci*** which …

   - Asks the user for which term in the Fibonacci sequence they would like to view.
   - Displays the user's requested term (be sure to communicate which term is being displayed).
   - Displays the sum of all of the terms in the Fibonacci sequence up to (and including) the user's chosen term. In order to display this sum, write your own user-defined method called ***sumFib( )*** that uses a standard for-loop to keep track of and return the sum (you will need to call the ***fibonacci( )*** method within the for-loop in order to generate the Fibonacci numbers.
   - Implements a while-loop within the ***main( )*** method that causes the program to repeat over-and-over until the user enters 0. Entering zero ends the program.

```
public static int fibonacci(int a)
   {
      if(a==0)
         return 0;
      else if(a==1)
         return 1;
      else
         return fibonacci(a-1) + fibonacci(a-2);
   }
```

2. Remember that a method header does not always have to read: *public static int methodName( )*
   Remember that the return type 'int' could also be other types (i.e. double, String, void). The method below is an example of a return type 'void' since it is not returning any values to the main method. The ***getThis( )*** method is simply recursively printing data … not returning values. Read ***unit04WS02*** class below. Be sure to understand what it is doing. Attempt to come up with what the output is and then enter the code into Netbeans. Were you right? If so, great work. If not, be sure you can see why.

```
package unit04;

public class unit04WS04 {
   public static void getThis(int x)
   {
      if(x==0)
         System.out.print("A");
      else if(x%3==0)
      {
         for(int i=0 ; i<x ; i++)
         {
            System.out.print("X");
         }
         System.out.println("");
         getThis(x-1);

      }
      else
         getThis(x-1);

   }
   public static void main(String[] args) {
      getThis(10);
   }
}
```