import java.util.*;

class DivisibleBy5 {

  public static void main(String[] args) {
    Scanner in = new Scanner(System.in);
    System.out.print("What number do you want me to check? ");
    int numToTest = in.nextInt();
    if(isDivisibleBy5(numToTest)) {
      System.out.println(numToTest + " is divisible by 5.");
    } else {
      System.out.println(numToTest + " is not divisible by 5.");
    }
  }

  // Below is the class method (or static method) for checking whether
  // or not a number is divisible by 5. In real code, you likely wouldn’t
  // write this as a method, because it’s so simple. It is used here
  // as an example of method writing.
  //
  // A few interesting details:
  //  - The method is static. This means that you do not need an object
  //    (of type DivisibleBy5) to call the method. Note also that
  //    the main method is static. Because main doesn’t have access to
  //    an object, it is necessary also to make isDivisibleBy5 static
  //    so that it can be called from main.
  //  - The method is private. This means that it can be called only from
  //    within the same class. We have no need to call this method from
  //    elsewhere, so private is the safest choice. Less visibility leads
  //    to more robust code.
  //  - The method returns a boolean. This means that when it is done
  //    computing, it will produce a true/false result. It *returns* this
  //    result to the caller (the main method).
  //  - The method takes one parameter, an int. A parameter is a piece
  //    of information that the method needs to be able to do its work.
  //    The caller of a method must specify the actual value of a
  //    parameter.
  //  - The method is documented. This refers to the comment before the
  //    method, written in the Javadoc style. Javadoc is a tool
  //    (integrated into Eclipse) that produces the easy-to-read
  //    documentation about Java methods and other features.
  //    Javadoc comments start with "/**" and contain @... tags to
  //    denote particular documentation elements.

  /** Tests whether a number is divisible by 5.
   * @param n The number to test for divisibility.
   * @return True if the number is divisible by 5; false otherwise
   */
  private static boolean isDivisibleBy5(int n) {
    return (n % 5 == 0);
  }
}