**Program Fourteen Part One:**

//summary: this program takes in an index from the user and finds the Fibonacci number for that index

//name: Jenna Wolf

//class: Fundamentals of Programming, CS155 - 01

//instructor: Dr. Art Kazmierczak

//date: 11/17/2023

import java.util.Scanner; //lets inputs be made

public class Main

{

public static void main(String[] args)

{

Scanner input = new Scanner(System.in); //names the input

int index = 0, f0 = 0, f1 = 1, currentfib = 0; //holds the data and sets it to 0

//takes in the index from the user

System.out.print("please enter an index for the Fibonacci number you want to see: ");

index = input.nextInt();

//checks to see if index is one and changes currentfib if it is

if(index == 1)

currentfib = 1;

//for loop that goes until i is bigger than index

for(int i = 1; i < index; i++)

{

currentfib = f0 + f1; //sets currentfib to f0 + f1

f0 = f1; //sets f0 to f1

f1 = currentfib; //sets f1 to currentfib

}

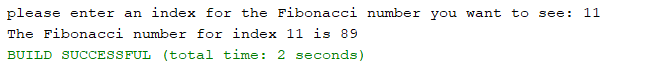
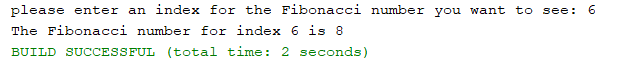
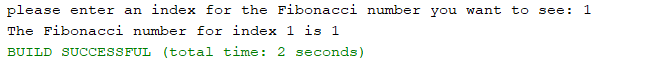
//outputs the Fibonacci number

System.out.print("The Fibonacci number for index " + index + " is " + currentfib);

}

}

Output:



**Program Fourteen Part Two:**

//summary: this program uses recursion to find the greatest common divisor of two numbers

//name: Jenna Wolf

//class: Fundamentals of Programming, CS155 - 01

//instructor: Dr. Art Kazmierczak

//date: 11/17/2023

import java.util.Scanner; //lets inputs be made

public class Main

{

public static void main(String[] args)

{

Scanner input = new Scanner(System.in); //names the input

double x, y; //holds the x and y data

//explains the program and takes in two numbers

System.out.println("This program will find the greatest common divisor of the two numbers you enter.");

System.out.print("Please enter your first number: ");

x = input.nextDouble();

System.out.print("Please enter your second number: ");

y = input.nextDouble();

//callds the gcd function and outputs the results

System.out.println("The greatest common divisor is " + gcd(x, y));

}

public static double gcd(double m, double n)

{

//if m / n has a remainder of 0, n is returned. if not, gcd is called again

//with the current remainder

if(m % n == 0)

return n;

else

return gcd(n, m % n);

}

}

Output:

