

Taller GitHub, Objetos

Integrantes:

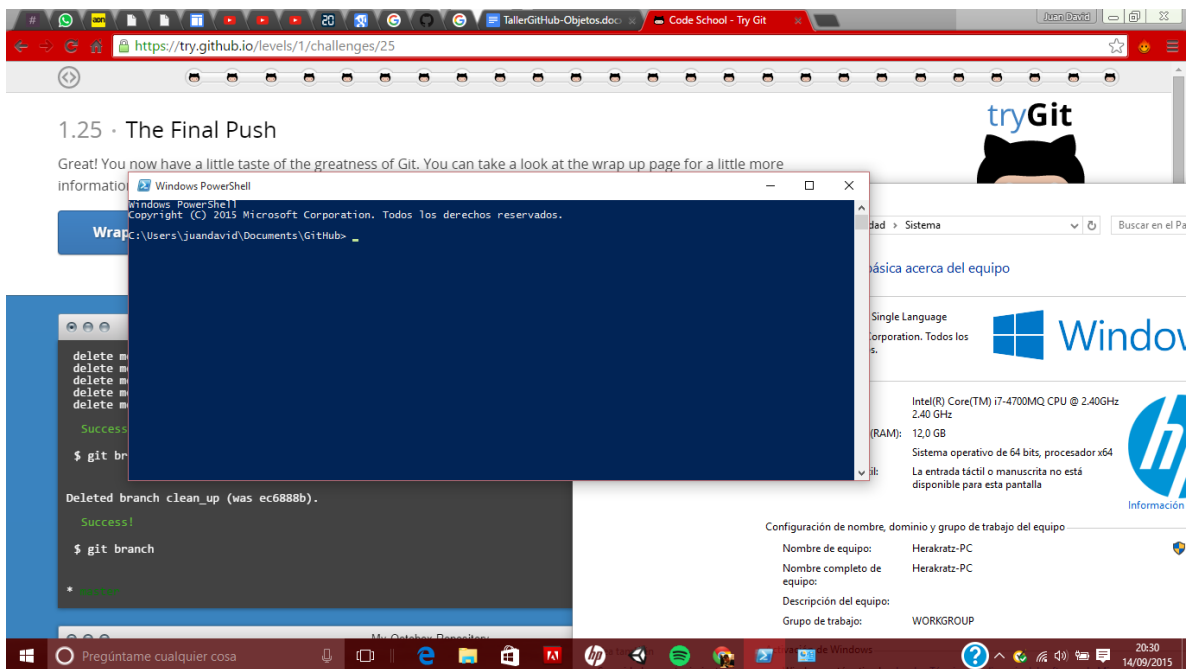
John Jairo Montoya - jjmontoyag@unal.edu.co

Juan David Balcázar Bedoya - jdbalcazarb@unal.edu.co

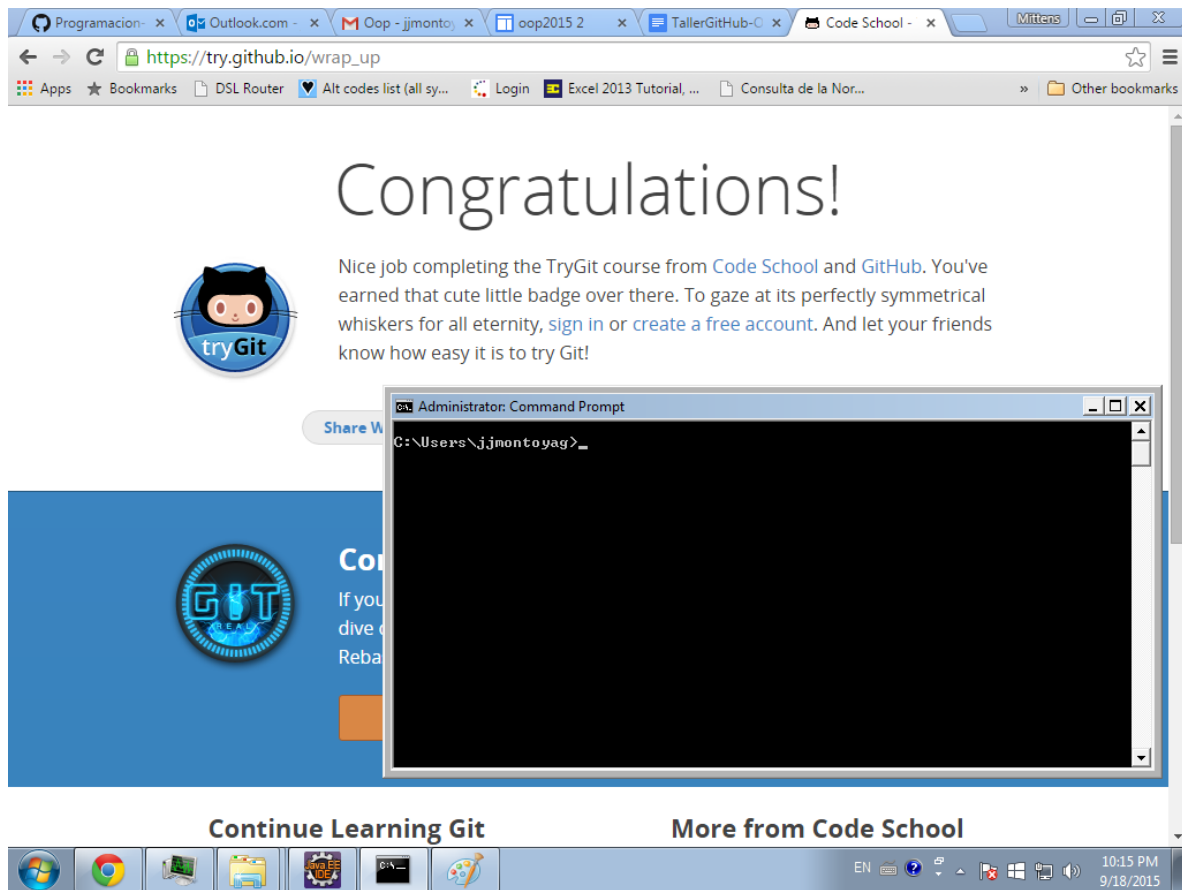
2) probando Git:

- Agregue por cada integrante una imagen donde se muestre el nombre/pc y el desarrollo completo de la guía.

Try GitHub – Juan David Balcázar Bedoya



Try GitHub – John Jairo Montoya



3) Registro

a) Juan David Balcázar

Usuario: Herakratz

Correo: john-dvid@hotmail.com

John Jairo Montoya

Usuario: jjmontoyag

Correo: jjmontoyag@unal.edu.co

b) Repositorio:

<https://github.com/jjmontoyag/Programacion-Orientada-a-Objetos-2015-2>

4)

a) Integers

```
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;

public class Integers {

    private int n1 = 0;
    private int n2 = 0;
    private BufferedReader br;

    //Lee los numeros ingresados por el usuario
    private void readNumbers() throws IOException{

        br = new BufferedReader(new InputStreamReader(System.in));

        System.out.println("Ingrese numero 1: ");
        n1 = Integer.parseInt(br.readLine());
        System.out.println("Ingrese numero 2: ");
        n2 = Integer.parseInt(br.readLine());
    }

    //Dice si es par o no
    private void isEven(int n){

        if(n % 2 == 0){

            System.out.println("El numero " + n + " es PAR");
        }else{

            System.out.println("El numero " + n + " es IMPAR");
        }
    }

    //Dice cual es el maximo
    private void max(int a, int b){

        if(a > b){

            System.out.println(a + " es mayor que " + b);
        }else if(a == b){

            System.out.println("Los numeros dados son iguales");
        }else {

            System.out.println(b + " es mayor que " + a);
        }
    }

    //Halla si A es multiplo o no de B
    private void isAMultipleOfB(int a, int b){
```

```

        if(a % b == 0 ){

            System.out.println(a + " ES multiplo de " + b);
        }else {

            System.out.println(a + " NO es multiplo de " + b);

        }
    }
    //Corre todas los metodos
    public void run() throws IOException{

        readNumbers();
        isEven(n1);
        isEven(n2);
        max(n1, n2);
        isAMultipleOfB(n1, n2);

    }

}

```

b) Spaces

```

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
import java.util.ArrayList;

public class Spaces {

    private int number = 0;
    private ArrayList<Integer> digits;
    private BufferedReader br;

    //Pide el numero
    private void readNumber() throws NumberFormatException, IOException{

        br = new BufferedReader(new InputStreamReader(System.in));
        System.out.println("Ingrese un numero entero: ");
        number = Integer.parseInt(br.readLine());

    }

    private ArrayList<Integer> findDigits(){

        boolean quit = false;
        double digit = number;
        int decimals = 1;
        digits = new ArrayList<Integer>();
        //Divide de diez en diez y guarda el digito que es menor a 10
        //hasta terminar el numero de dijitos.
        while(!quit){

            if(number < 10){

                digits.add(number);
                quit = true;
                break;

            }

        }

    }

}

```

```

        if(digit < 10){

            int digitToAdd = (int)Math.floor(digit);
            digits.add(digitToAdd);
            number = number - (digitToAdd * decimals);
            digit = number;
            decimals = 1;
        }else {

            digit = digit / 10;
            decimals = decimals * 10;
        }
    }

    return digits;
}
//Imprime los numeros guardados en la lista digits
private void printDigits(){

    for(int i = 0; i < digits.size(); i++){

        System.out.print(digits.get(i) + "  ");
    }

}

public void run() throws NumberFormatException, IOException{

    readNumber();
    findDigits();
    printDigits();

}
import java.io.IOException;

public class Taller1 {

    public static void main(String[] args) throws IOException {

        //*****INTEGERS*****//
        Integers integers = new Integers();
        integers.run();

        //****SPACES*****//
        Spaces spaces = new Spaces();
        spaces.run();

    }

}

```

c) Average

```
package average;
```

```

import java.util.Scanner;

/**
 *
 * @author Juan David Balcázar Bedoya Programación Orientada a Objetos 2015-2
 * Taller 1 4.C Average
 */

/*Escriba un programa que pregunte por números enteros (9999 para salir),
calcule e imprima el promedio de los números ingresados.
Un ejemplo de entrada es las secuencias [999 1 15 5 9999] indicando el promedio
de todos los valores que preceden a 9999 debe ser calculado, este debe imprimir:
[The average is: 255]
*/
public class Average {

    private void promedio(Scanner sc) { //Recibe un scanner para poder leer infinitos
numeros
        //Declaracion variables
        int numero;
        int suma = 0;
        int contador = 0;

        for (;;) { // for infinito
            numero = sc.nextInt(); //Lectura de un numero
            if (numero != 9999) { // verifica que el numero ingresado no sea 9999
                suma = suma + numero;
                contador++;
            } else {
                System.out.println("The Average is: " + suma / contador); //Calcula e
imprime el promedio
                return;
            }
        }
    }

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in); // nuevo Scanner sc
        Average average = new Average(); //Nuevo Average

        System.out.println("Digite los numeros para el promedio: ");
        System.out.println("9999 para salir");
        average.promedio(sc); //llama al metodo promedio y le da como parametro sc
    }
}

```

d)ATM

```
package atm;

import java.util.Scanner;

/**
 * @author Juan David Balcázar Bedoya
 * Programación Orientada a Objetos 2015-2
 * Taller 1 4.D ATM
 *
 */
/*
Los clientes de un banco quieren realizar retiros de un cajero,
Los clientes escriben el monto que quieren retirar, Luego el cajero les devuelve
la cantidad solicitada con el menor número de monedas y billetes.
Su tarea es simular el proceso que realiza el banco, dado un monto múltiplo de 50 pesos,
usted debe imprimir la cantidad de billetes y monedas que daría de cada denominación.
Para esta tarea se usaran los siguientes tipos de billetes y monedas:
Billetes: 1000, 2000, 5000, 10000, 20000, 50000
Monedas: 50, 100, 200, 500
*/
public class ATM {

    private void calcularDinero(int monto) {

        //Declaracion variables, billetes y monedas
        int b50000 = 0;
        int b20000 = 0;
        int b10000 = 0;
        int b5000 = 0;
        int b2000 = 0;
        int b1000 = 0;
        int m500 = 0;
        int m200 = 0;
        int m100 = 0;
        int m50 = 0;
        int multiplo = monto % 50; //modulo del monto ingresado

        if (multiplo == 0) { // Si es multiplo de 50 hace lo siguiente
            System.out.println("El total de billetes :");

            if (monto >= 50000) { //compreba que monto es mayor a 50000
                b50000 = monto / 50000; //obtiene la cantidad de billetes
                monto = monto % 50000; //modulo de monto, dice la cantidad faltante por
calcular
                System.out.println("    " + b50000 + " billetes de $50.000"); //imprime
la cantidad de billetes
            }
            if (monto >= 20000) {
```

```

        b20000 = monto / 20000;
        monto = monto % 20000;
        System.out.println("    " + b20000 + " billetes de $20.000");
    }
    if (monto >= 10000) {
        b10000 = monto / 10000;
        monto = monto % 10000;
        System.out.println("    " + b10000 + " billetes de $10.000");
    }
    if (monto >= 5000) {
        b5000 = monto / 5000;
        monto = monto % 5000;
        System.out.println("    " + b5000 + " billetes de $5.000");
    }
    if (monto >= 2000) {
        b2000 = monto / 2000;
        monto = monto % 2000;
        System.out.println("    " + b2000 + " billetes de $2.000");
    }
    if (monto >= 1000) {
        b1000 = monto / 1000;
        monto = monto % 1000;
        System.out.println("    " + b1000 + " billetes de $1.000" + "\n");
    }
    if (monto != 0) { //comprueba que se necesita entregar monedas
        System.out.println("El total de monedas :");
    }
    if (monto >= 500) {
        m500 = monto / 500;
        monto = monto % 500;
        System.out.println("    " + m500 + " monedas de $500");
    }
    if (monto >= 200) {
        m200 = monto / 200;
        monto = monto % 200;
        System.out.println("    " + m200 + " monedas de $200");
    }
    if (monto >= 100) {
        m100 = monto / 100;
        monto = monto % 100;
        System.out.println("    " + m100 + " monedas de $100");
    }
    }
    if (monto >= 50) {
        m50 = monto / 50;
        monto = monto % 50;
        System.out.println("    " + m50 + " monedas de $50");
    }
    } else {
        System.out.println("El monto que solicita no es multiplo de 50");
    }
}

public static void main(String[] args) {

    ATM cajero = new ATM();

```



```

        int monto;
        Scanner sc = new Scanner(System.in); //nuevo Scanner sc
        System.out.println("_____Bienvenido al ATM_____ ");
        System.out.println("Digite el monto a retirar: ");
        monto = sc.nextInt(); //lectura del dato monto

        cajero.calcularDinero(monto); //Se llama al metodo calcularDinero y le pasamos
        como parameto monto
    }

}

```

e) Prime Numbers

```

package primenumbers;

import java.util.Scanner;

/**
 *
 * @author Juan David Balcázar Bedoya Programación Orientada a Objetos 2015-2
 * Taller 1 4.E Prime Numbers
 */
/*Un número primo es un numero natural que solo tiene 2 divisores, 1 y sí mismo.
Dado un número entero (máximo 109) imprima una lista ordenada de los números primos
menores o iguales al número ingresado.
*/
public class PrimeNumbers {

    // Metodo para calular los primos menores al digitado
    public void numeroPrimo(long numero) {
        int contador = 0;
        for (int a = 1; a <= numero; a++) { // Ciclo for para buscar los primos desde 1
            hasta el numero
            for (int i = 1; i <= a; i++) { //Ciclo for para comprobar que sea primo el
                numero a
                if (a % i == 0) { //m
                    contador = contador + 1;
                }
            }
            if (contador == 2) {
                System.out.print(a + " ");
                contador = 0; // hace contador 0 para poder hacer el mismo proceso con el
                siguiente numero
            } else {
                contador = 0;
            }
        }
    }

    public static void main(String[] args) {

        long numero;
        PrimeNumbers primos = new PrimeNumbers();
        Scanner sc = new Scanner(System.in);
    }
}

```

```

        System.out.println("Digite un numero: ");

        numero = sc.nextInt();
        if (numero < Math.pow(10, 9)) // verificar que el numero ingresado no sea mayor a
10^9
        {
            primos.numeroPrimo(numero);
        } else {
            System.out.println("es mayor de 10^9");
        }
    }
}

```

f) Tic-Tac-Toe (opcional)

Incluido en archivo aparte por ser extenso

Los dos primeros ejercicios se encuentran en la carpeta PuntosDelTaller