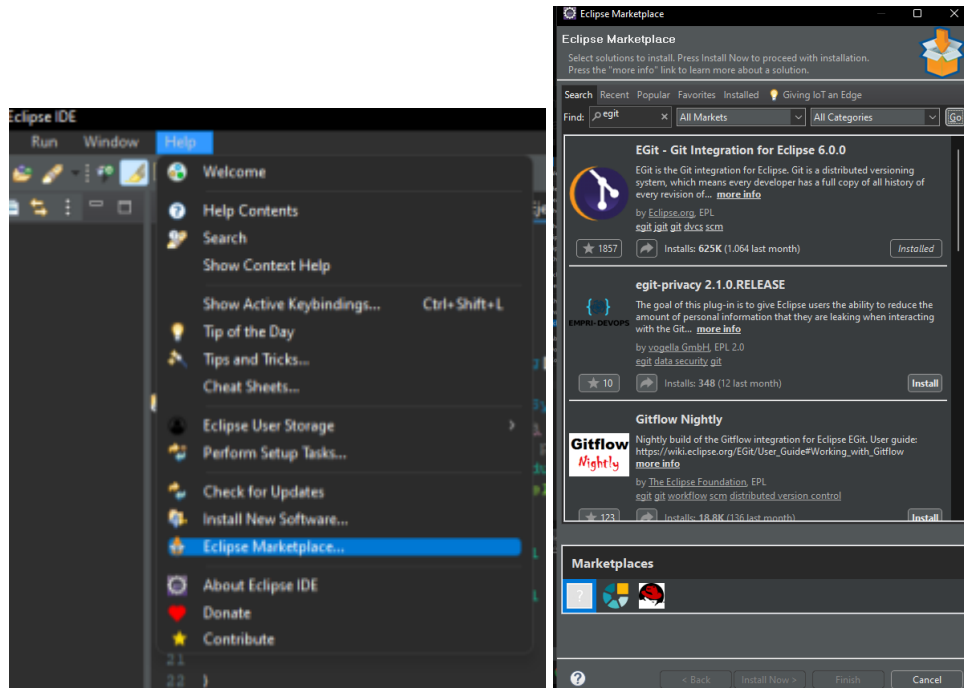


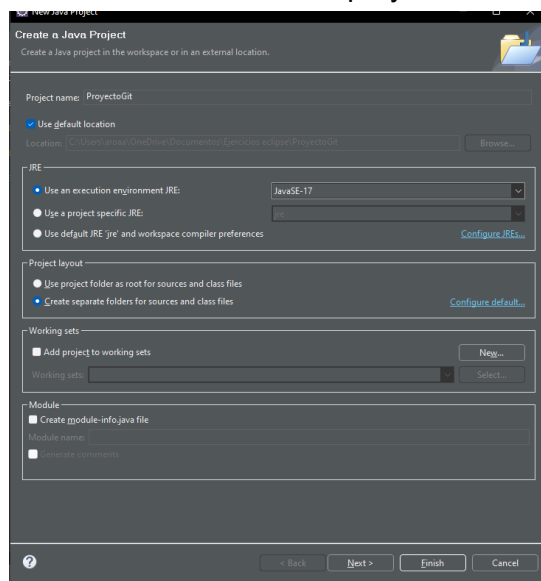
- Crea un nuevo repositorio local llamado Act4GIT\_Tunombre
- Crea un método en java para insertar en un array 20 números aleatorios entre 1 y 10
  - Añade ese programa a tu repositorio local.
- Crea en GitHub un repositorio vacío denominado Act4GH\_Tunombre.
  - Haz que el repositorio Act4GIT\_Tunombre tenga como destino remoto al repositorio en GitHub.
  - Consigue que en GitHub aparezca el contenido de tu repositorio local.
  - En el repositorio GitHub (Act4GH\_Tunombre) añade un readme.md con el contenido necesario para que el proyecto esté bien documentado en GitHub. Y un archivo .gitignore para no subir los .class
  - Etiqueta el repositorio como V1.0
- Crea una rama llamada rama1 desde la rama principal
  - Añade un método el Java para imprimir por pantalla el contenido del array.
  - Guarda esos cambios en el repositorio local.
- Crea otra rama llamada rama2 desde la rama1
  - Añade un método el Java para contar el número de veces que aparece cada número en el array.
  - Guarda esos cambios en el repositorio local.
- Realiza una fusión (merge) entre la rama1 y la rama2 y etiqueta el estado del proyecto como V2.0
- Posiciónate en la rama main y realiza un cambio sobre el método insertar array en el que vamos a añadir 30 números en lugar de 20.
- Fusiona el contenido de todas las ramas en la principal, en la que tienen que aparecer los tres métodos que hemos creado en su última versión
  - Etiqueta el estado de la rama como V3.0
- Sube todos los cambios a tu repositorio de GitHub y actualiza el contenido del Readme para reflejar la información sobre los últimos métodos añadidos.
- Busca en GitHub el repositorio Act2GH\_Tunombre de tu compañero de al lado y realiza un fork sobre el mismo.
  - Modifica uno de los métodos que aparezcan y envíale una Pull request para que añada tus cambios a su repositorio

## 1 Nos instalamos la expansión de EGit en eclipse:

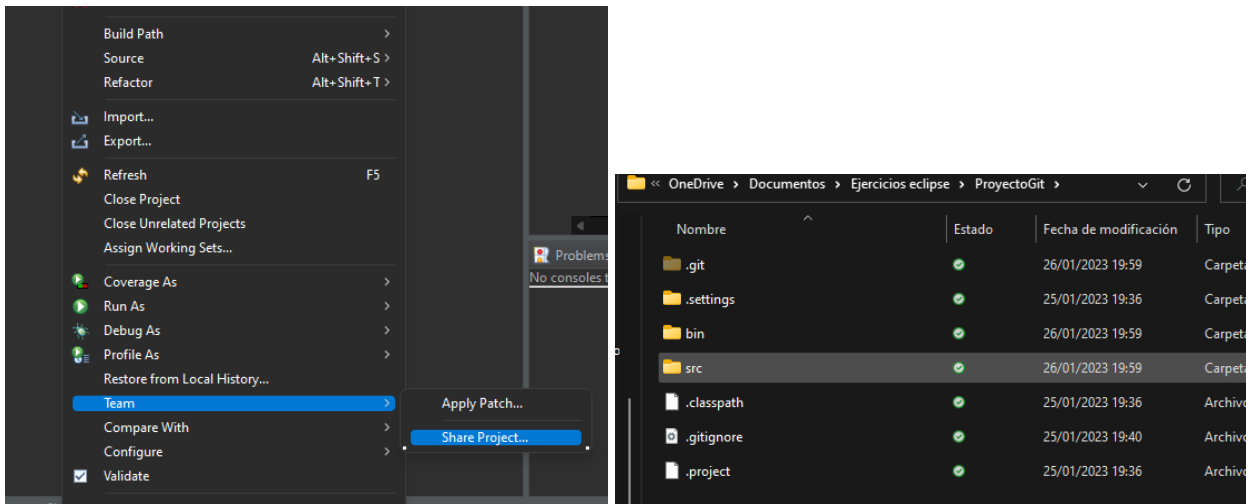


Una vez instalado procedemos con los siguientes pasos.

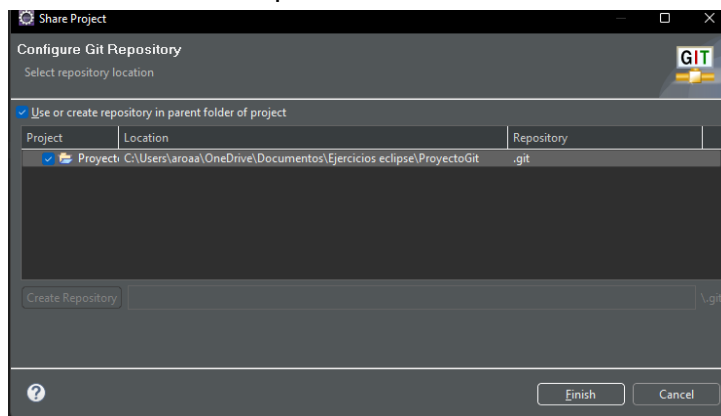
## 2. Nos creamos un nuevo proyecto



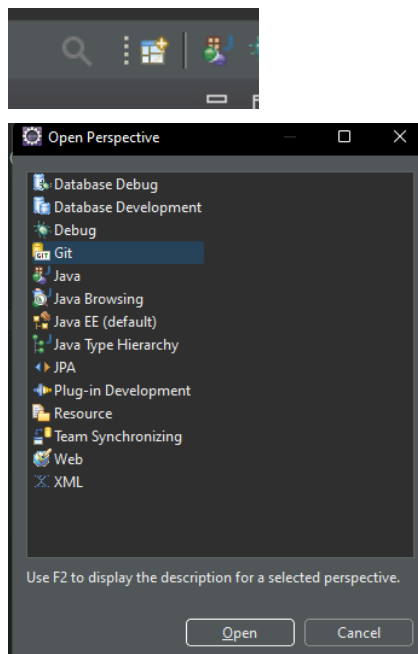
3. Hacemos click derecho en el ratón, seleccionamos team y share project para seleccionar la ruta de la nueva carpeta en la que estará el nuevo repositorio.



#### 4. Nos creamos el repositorio



#### 5. Activamos la perspectiva de Egit(esquina superior derecha), y seleccionamos git.



#### 6. Creamos el programa

```

public class MiPrimerProgramaEGit {
    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        int CANTIDAD = 20;

        int INICIOINTERVALO = 1;
        int FINALINTERVALO = 10;

        int [] elementos = new int [CANTIDAD];

        for (int i = 0; i < CANTIDAD; i++) {

            elementos [i] = (int) (Math.random() * (FINALINTERVALO - INICIOINTERVALO + 1)) + INICIOINTERVALO;

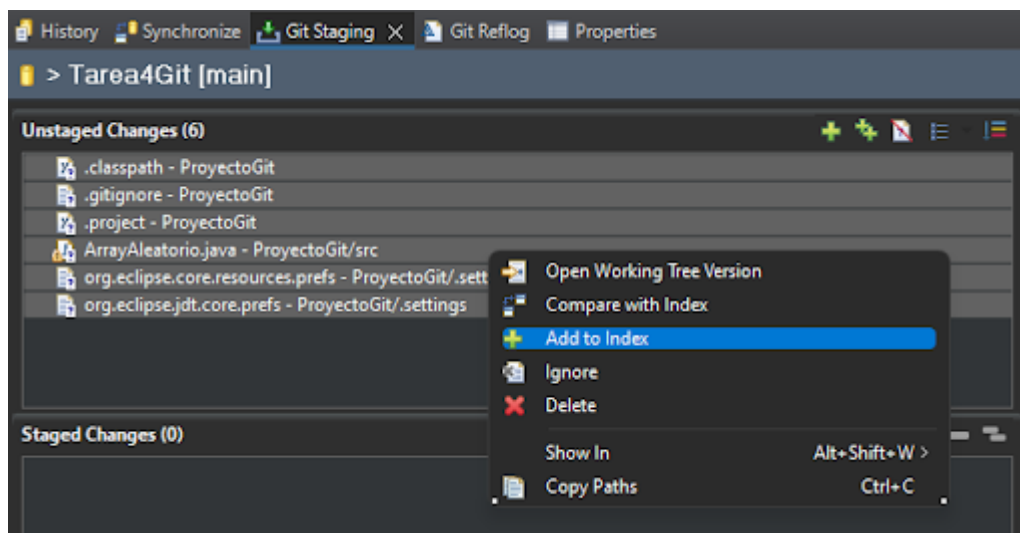
        }

        for (int i = 0; i < CANTIDAD; i++) {
            System.out.println(elementos[i]);
        }

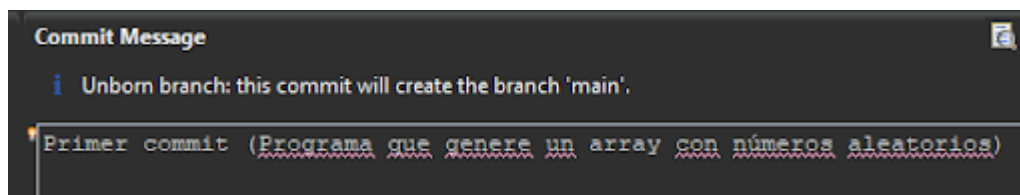
    }
}

```

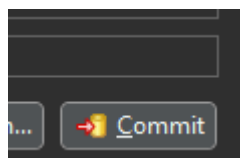
7. Nos vamos a la pestaña de git Staging y añadimos los ficheros y un mensaje de commit.



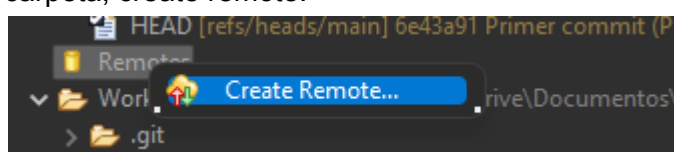
Añadimos todos los ficheros al Index



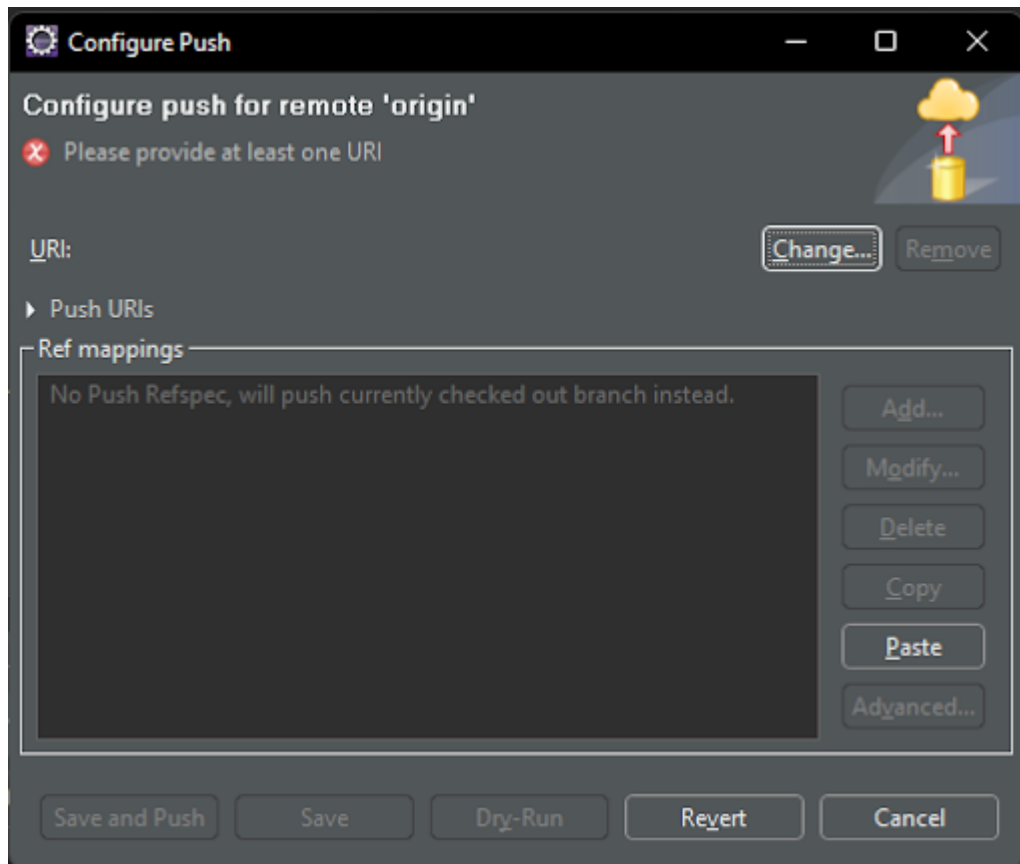
Añadimos también un commit y realizamos commit



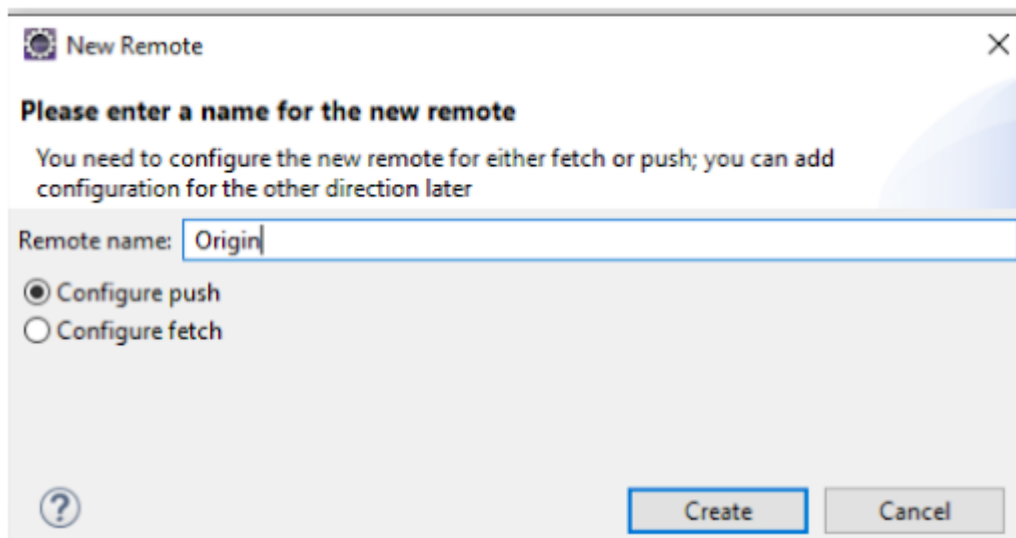
Después de realizar el commit, realizamos el push, para eso, hacemos click derecho en la carpeta, create remote.



Nos aparecerá esta pestaña, realizamos el change a nuestro repositorio de Git creado anteriormente lo guardamos y procedemos al push.

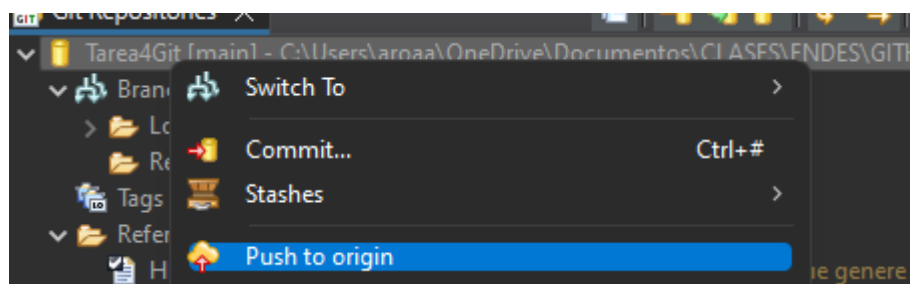
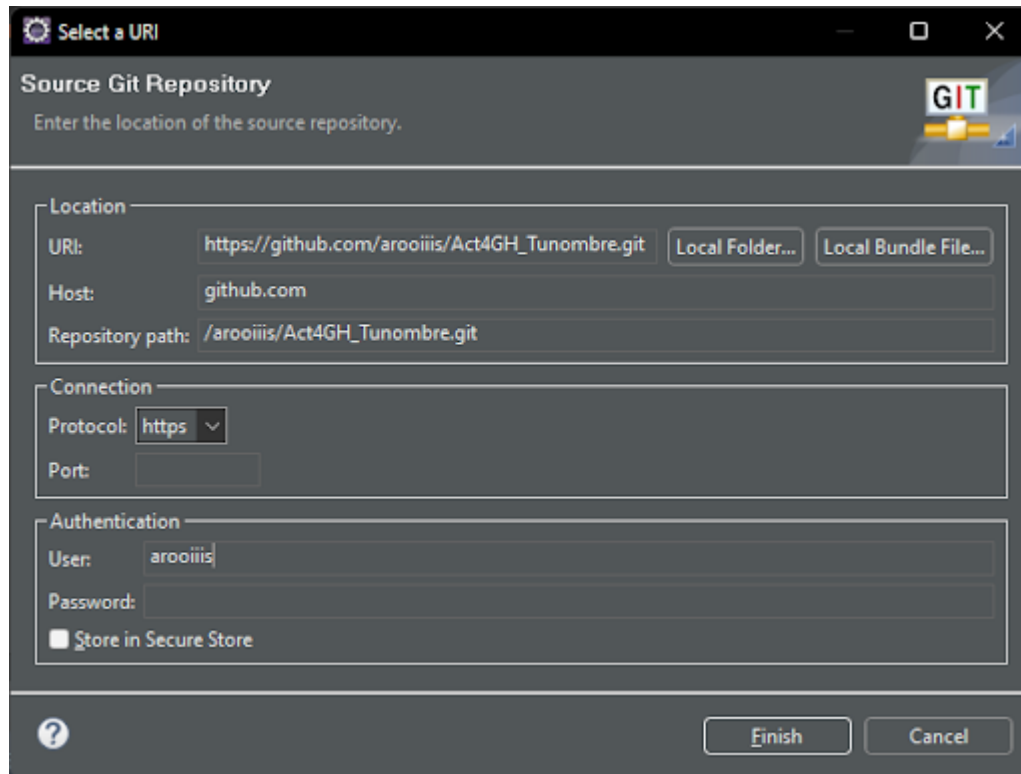


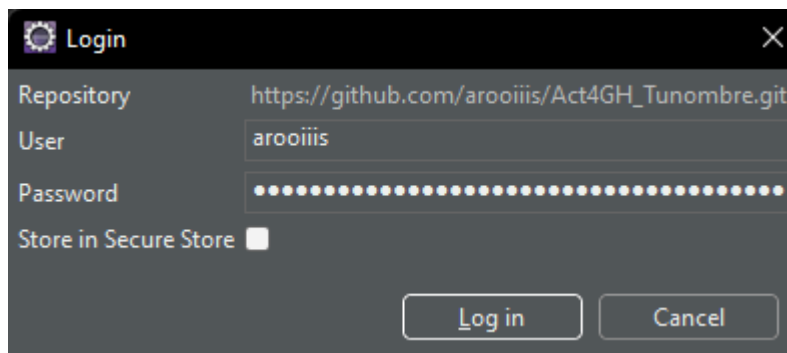
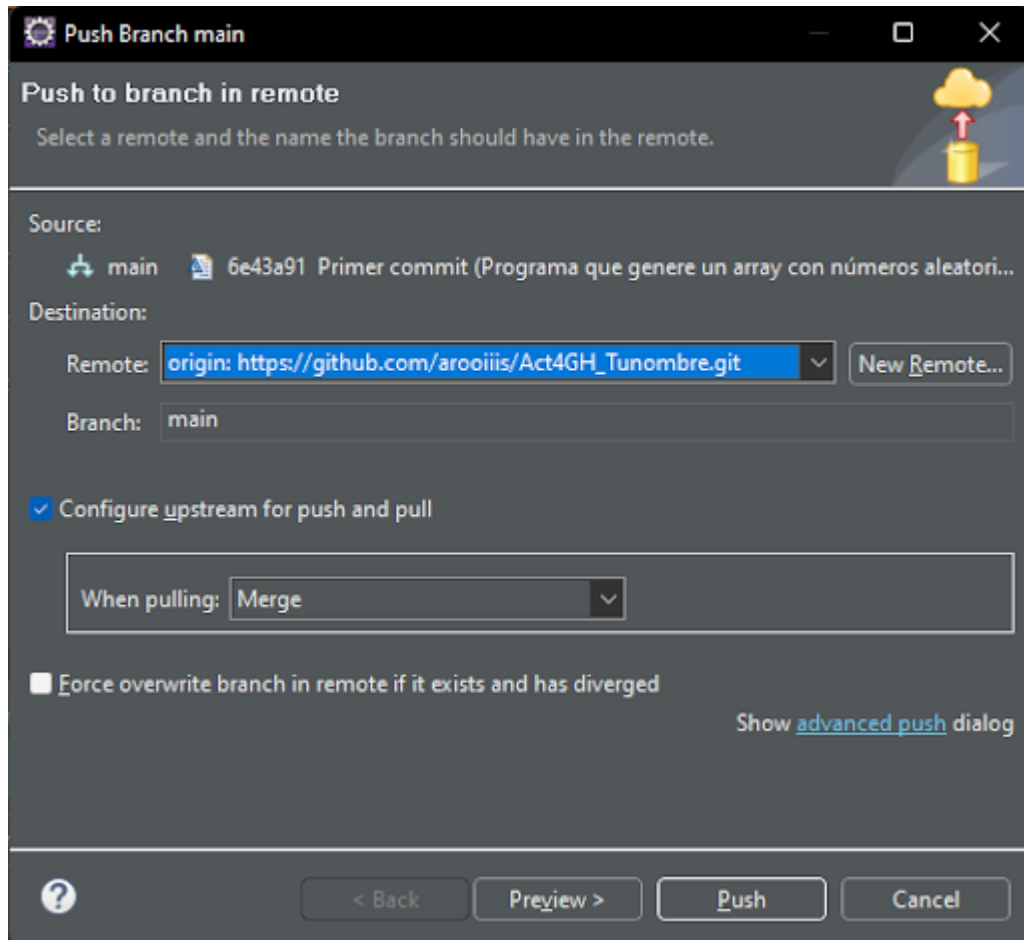
ponemos la rama Origin



(le había puesto el nombre mal, lo cambié luego)

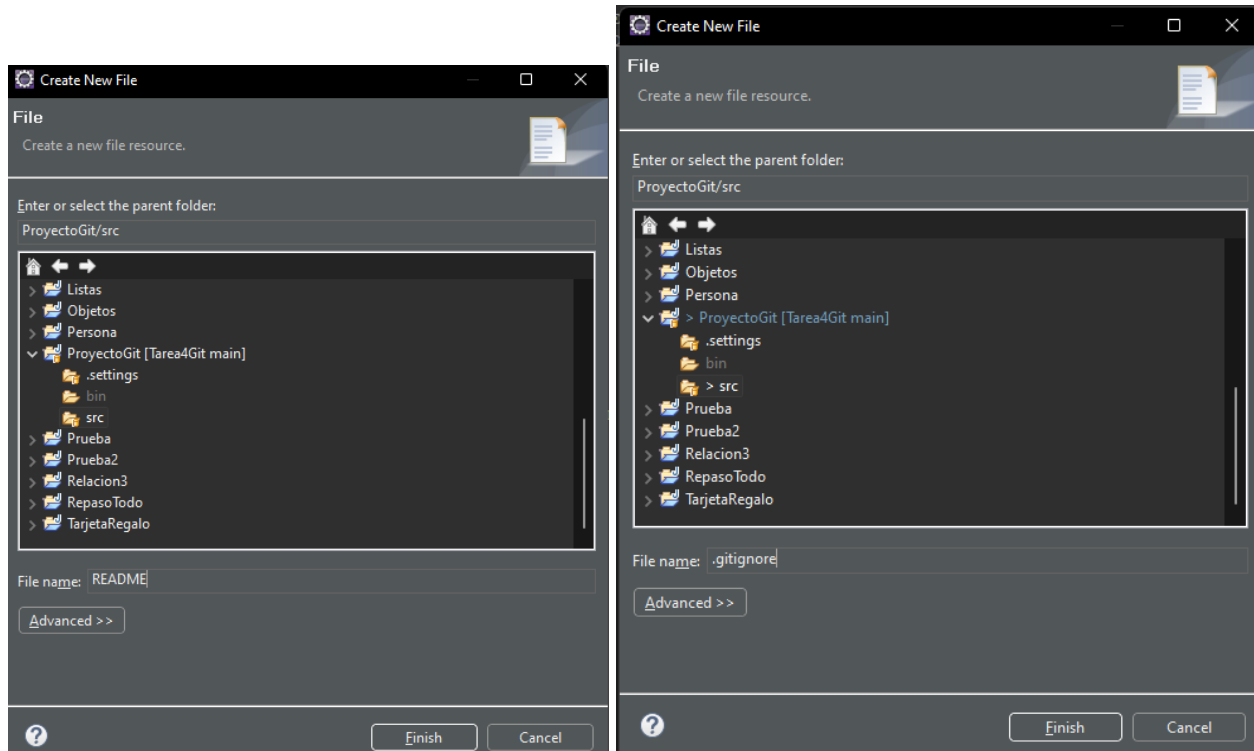
Nos pedirá para poder confirmar, nuestro user y la contraseña(token, yo la tenía creada de anteriormente)



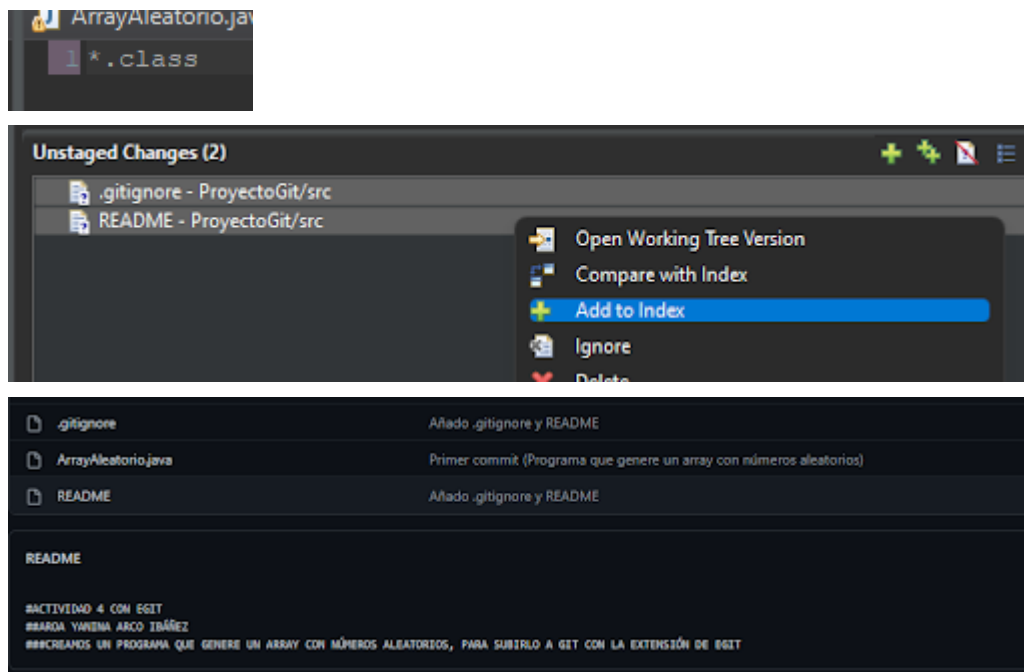


Una vez, introducido nuestro usuario y contraseña, nos deja realizar el pus.

Después creamos dos archivos, el README y el.gitignore, para eso, nos vamos a la perspectiva de java > click derecho en la carpeta > new > file.



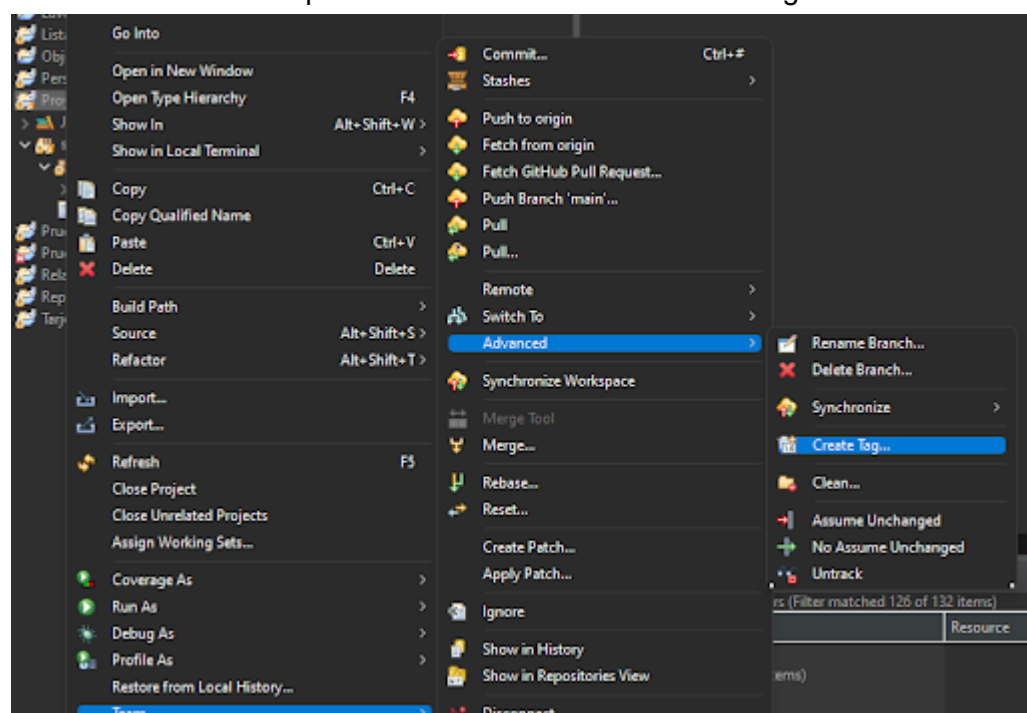
En el archivo de .ignore escribimos .class para ignorar estos archivos y hacemos commit y push.



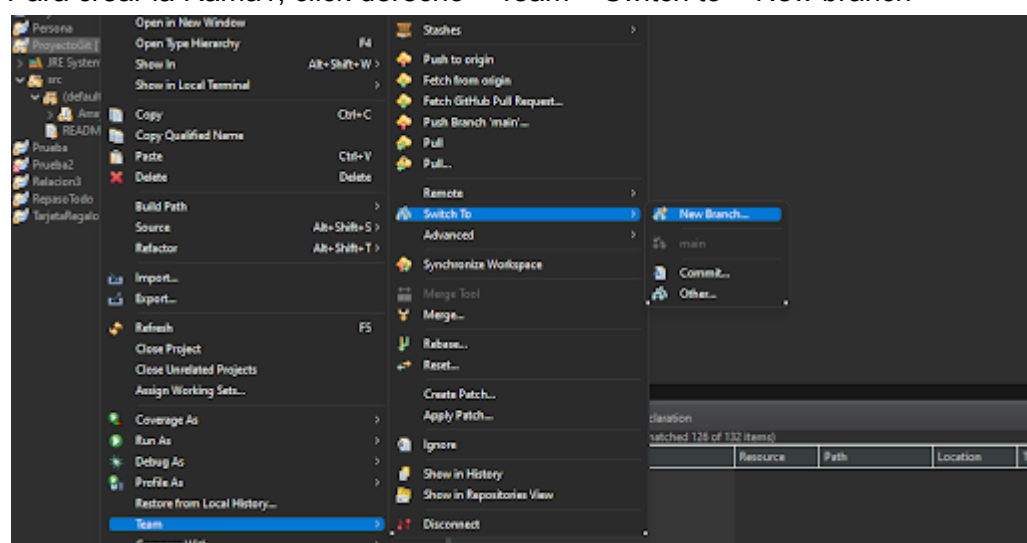


Para crear la etiqueta V.1

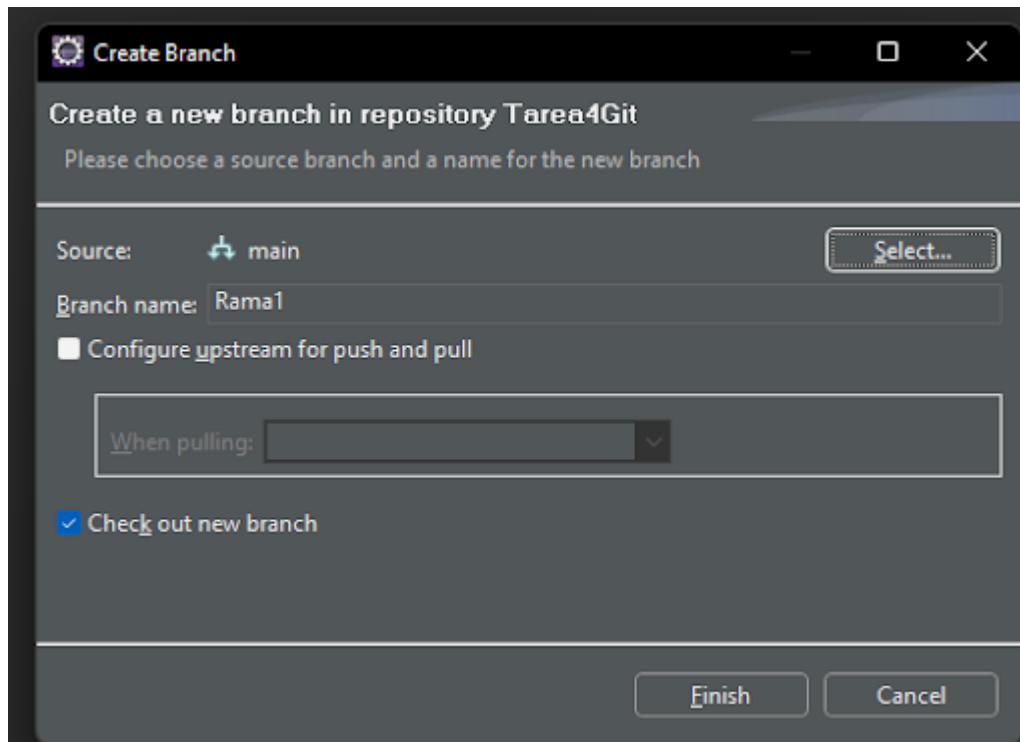
Click derecho en la carpeta > Team > Advanced > Create tag.



Para crear la Rama1, click derecho > Team > Switch to > New branch

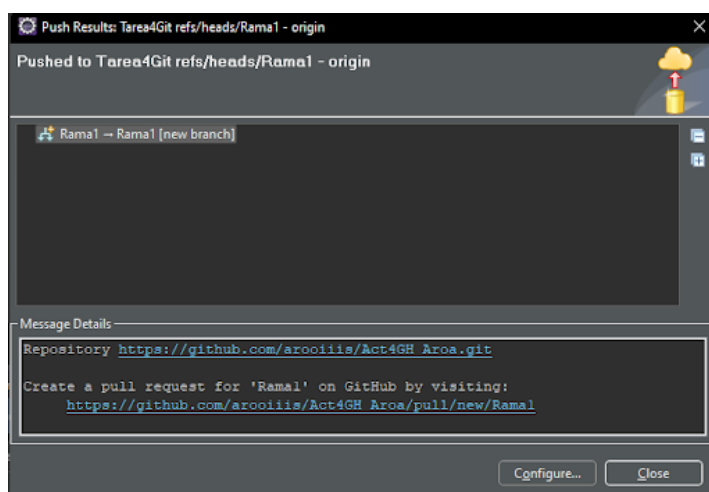


Le ponemos nombre

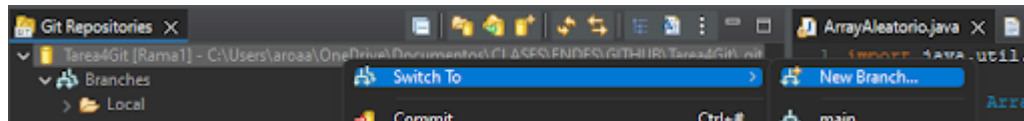


Añadimos la impresión del array en el programa y hacemos commit push en la rama1 como en los anteriores

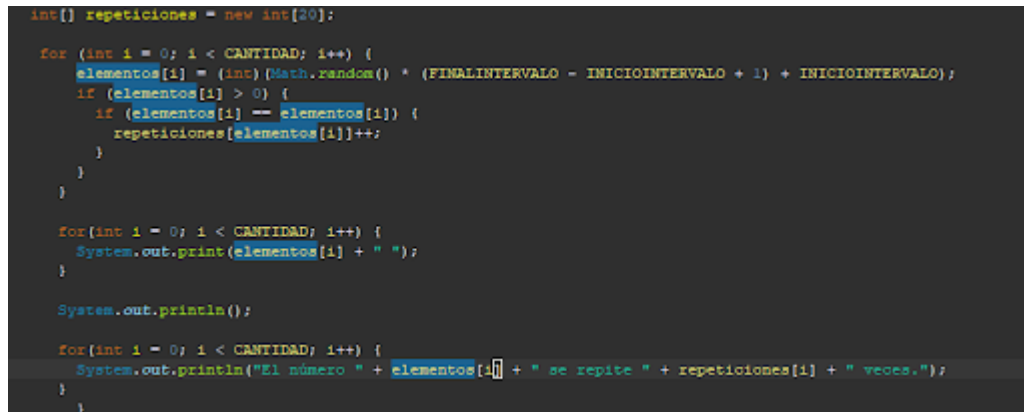
```
}  
for (int i = 0; i < CANTIDAD; i++) {  
    System.out.println(elementos[i]);  
}
```



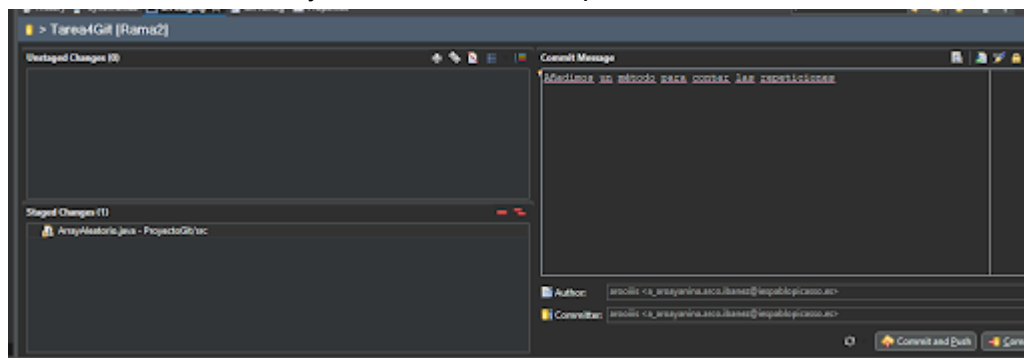
Creamos la Rama2 (mismos pasos que la rama1)



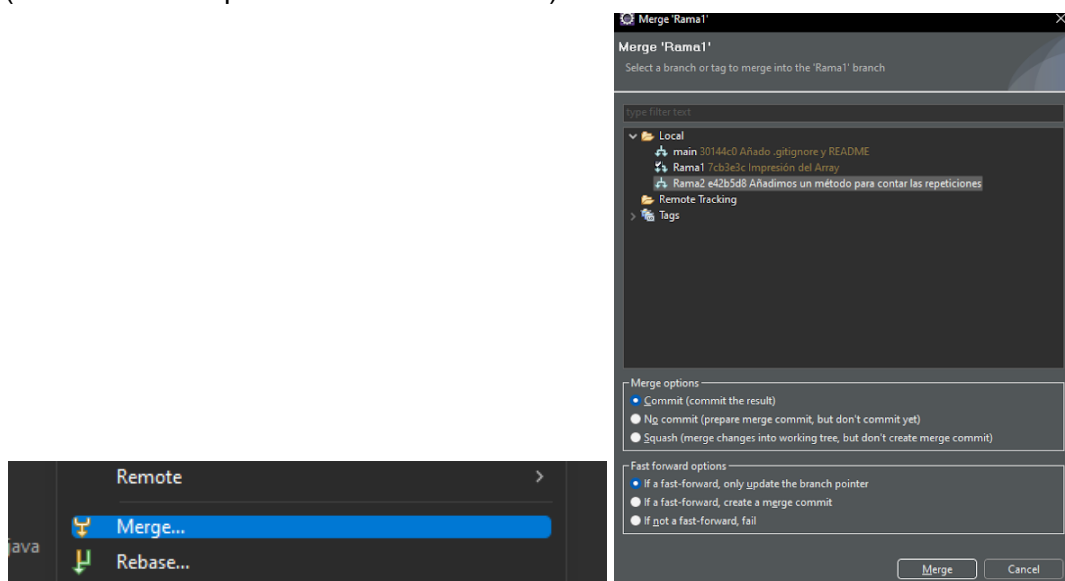
Añadimos un método que cuente la repetición de los números en el programa

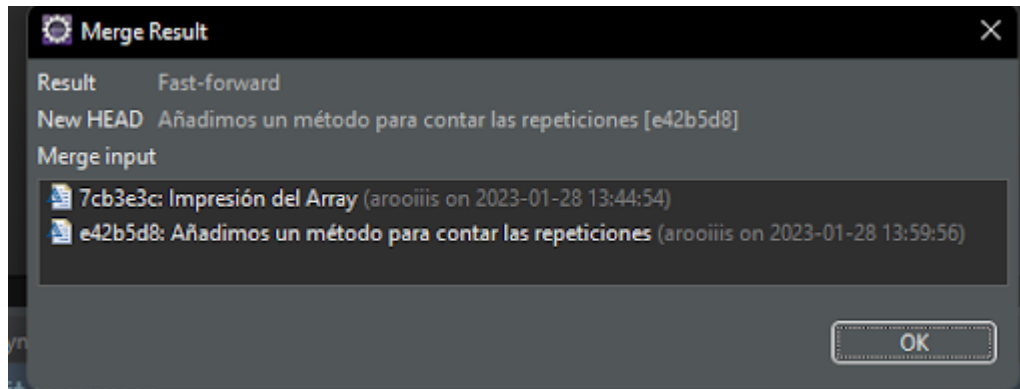


Añadimos los ficheros, y realizamos commit push

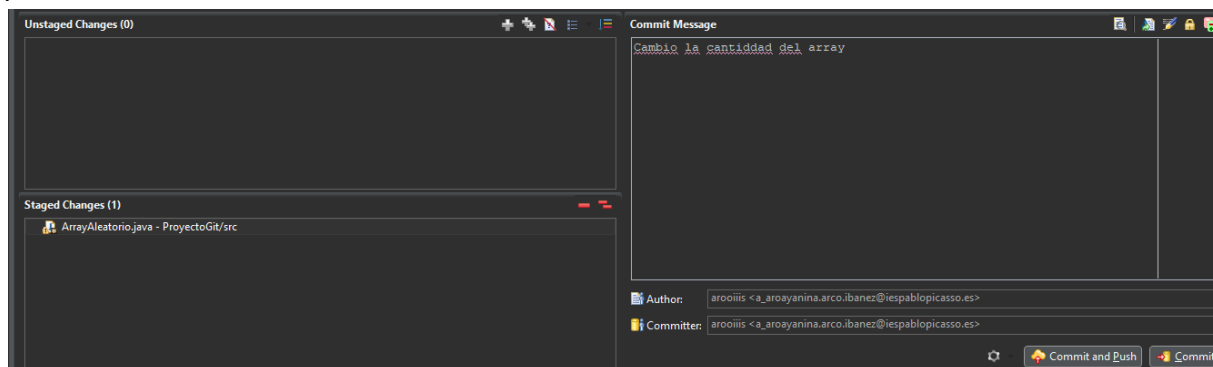


Para realizar la fusión entre las ramas > click derecho > team > merge  
(debíamos estar previamente en la rama1)

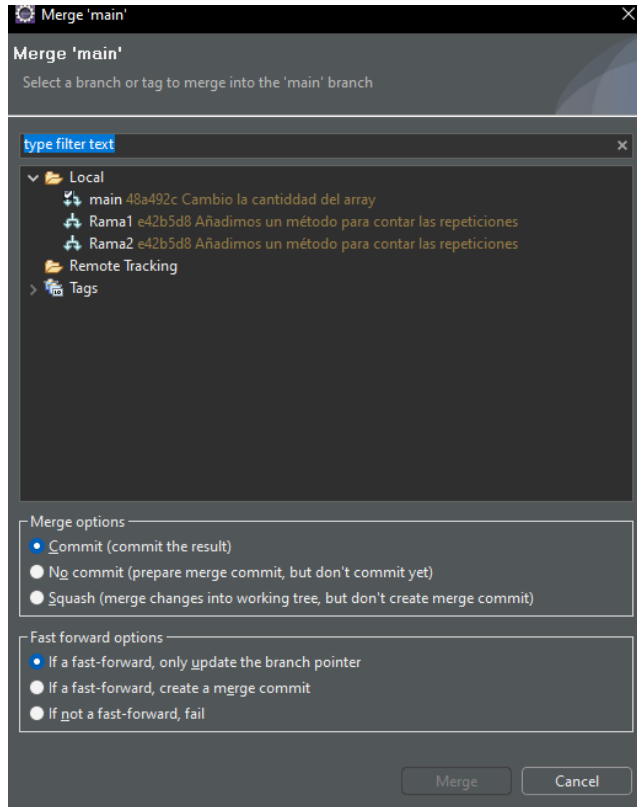


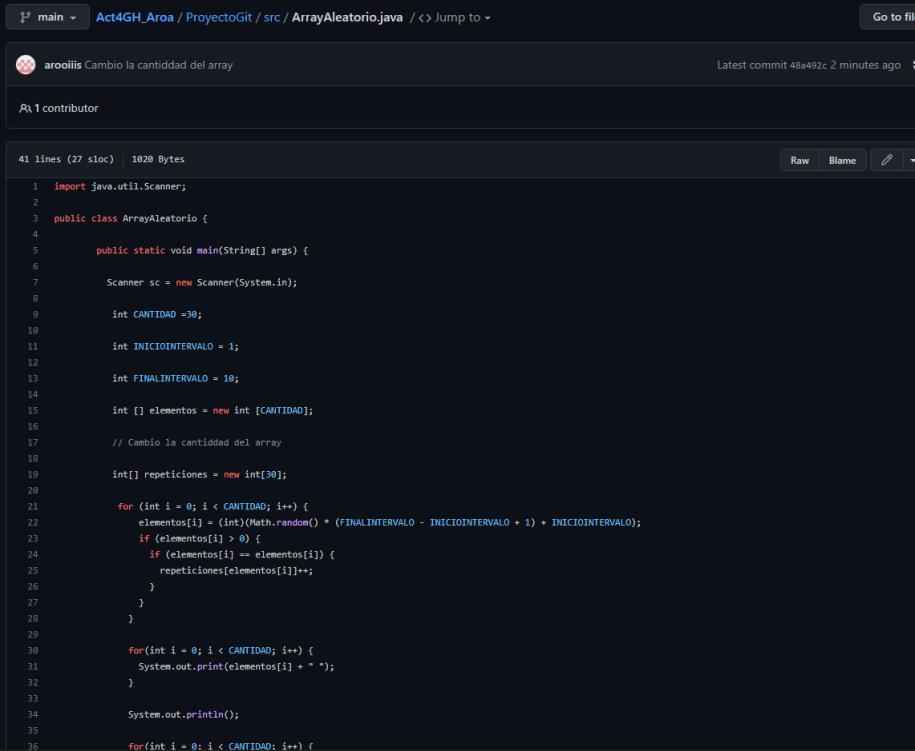


Después de fusionar las ramas, creamos la etiqueta v.2.(se me olvidó hacer captura)  
Nos vamos a la rama Main y editamos la cantidad del array, después realizamos commit y push.



Click derecho > merge.

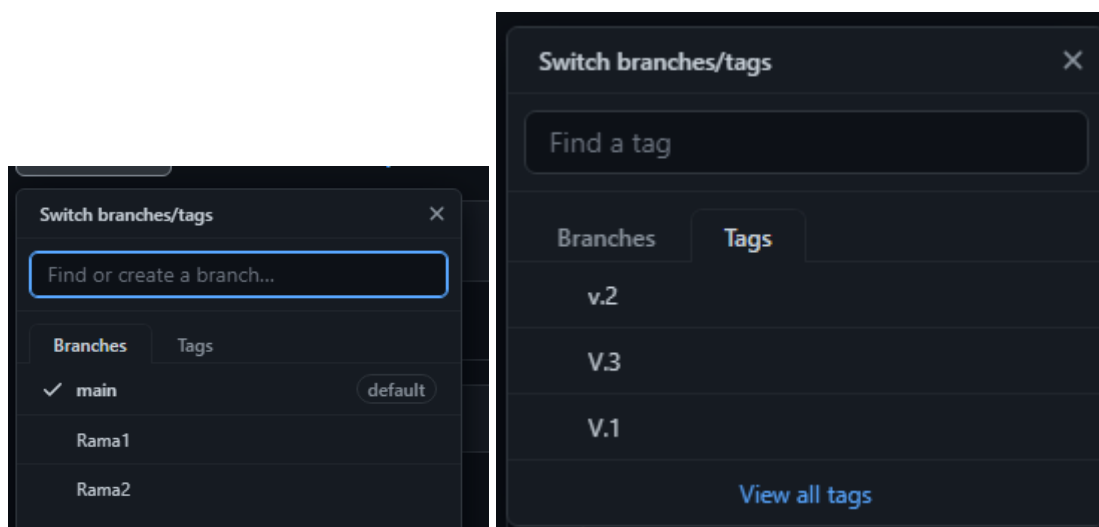
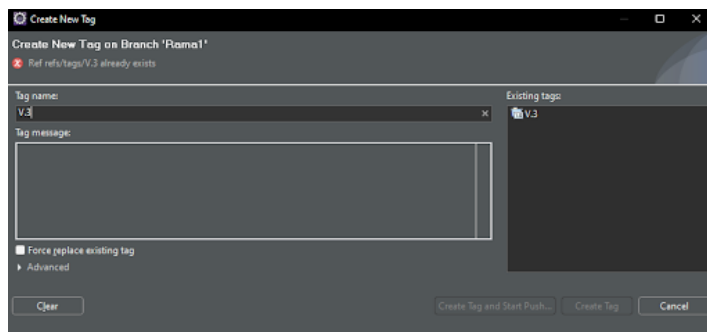




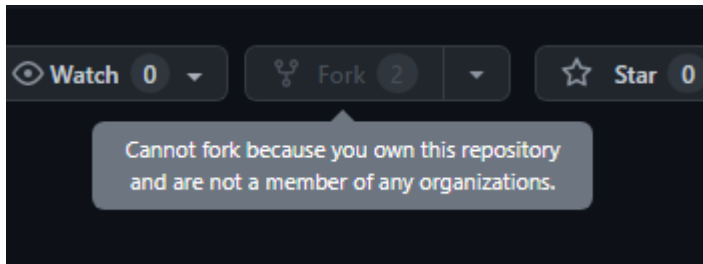
The screenshot shows a GitHub repository page for the file `ArrayAleatorio.java`. The repository is named `Act4GH_Aroa / ProyectoGit`. The file is 41 lines long (27 sloc) and 1020 Bytes. The commit history shows a single commit by `aroolis` titled "Cambio la cantidad del array" with the latest commit hash `48a492c` made 2 minutes ago. The code is as follows:

```
1 import java.util.Scanner;
2
3 public class ArrayAleatorio {
4
5     public static void main(String[] args) {
6
7         Scanner sc = new Scanner(System.in);
8
9         int CANTIDAD = 30;
10
11         int INICIOINTERVALO = 1;
12
13         int FINALINTERVALO = 10;
14
15         int [] elementos = new int [CANTIDAD];
16
17         // Cambio la cantidad del array
18
19         int[] repeticiones = new int[30];
20
21         for (int i = 0; i < CANTIDAD; i++) {
22             elementos[i] = (int)(Math.random() * (FINALINTERVALO - INICIOINTERVALO + 1) + INICIOINTERVALO);
23             if (elementos[i] > 0) {
24                 if (elementos[i] == elementos[i]) {
25                     repeticiones[elementos[i]]++;
26                 }
27             }
28         }
29
30         for(int i = 0; i < CANTIDAD; i++) {
31             System.out.print(elementos[i] + " ");
32         }
33
34         System.out.println();
35
36         for(int i = 0; i < CANTIDAD; i++) {
```

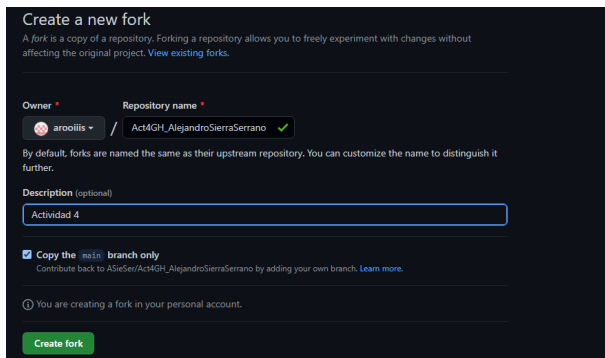
Ahora creamos la etiqueta V.3



Para realizar el fork, nos vamos al repositorio de un compañero, y le damos a fork, arriba a la derecha del repositorio.



Añadimos una descripción del repositorio



Y añadimos la modificación

