

	<p>UNIVERSIDAD NACIONAL DE SAN AGUSTIN FACULTAD DE INGENIERÍA DE PRODUCCIÓN Y SERVICIOS ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMA</p>	
<p>Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación</p>		
<p>Aprobación: 2022/03/01</p>	<p>Código: GUIA-PRLE-001</p>	<p>Página: 1</p>


## INFORME DE LABORATORIO

INFORMACIÓN BÁSICA					
ASIGNATURA:	Estructuras de datos y algoritmos				
TÍTULO DE LA PRÁCTICA:	Revisión de elementos de programación (Parte I)				
NÚMERO DE PRÁCTICA:	01	AÑO LECTIVO:	2023-A	NRO. SEMESTRE:	III
FECHA DE PRESENTACIÓN	7/05/2023	HORA DE PRESENTACIÓN	5:00 pm		
<b>INTEGRANTE (s):</b> <ul style="list-style-type: none"> <li>- Borda Espinoza Gabriela</li> <li>- Condori Pinto Juan José</li> <li>- Gordillo Mendoza, Jose Alonzo</li> <li>- Lopez Arela, Ower Frank</li> </ul>				NOTA:	
<b>DOCENTE(s):</b> Mg. Edith Giovanna Cano Mamani					

SOLUCIÓN Y RESULTADOS
<p><b>1. SOLUCIÓN DE EJERCICIOS/PROBLEMAS</b></p> <ul style="list-style-type: none"> <li>Cree una cuenta de usuario en GitHub usando su correo institucional. (cada uno)</li> </ul> <p>Cuenta de Juan José Condori Pinto:</p> <div data-bbox="402 1449 1275 1948">  </div>

	<p>UNIVERSIDAD NACIONAL DE SAN AGUSTIN FACULTAD DE INGENIERÍA DE PRODUCCIÓN Y SERVICIOS ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMA</p>	
<p><b>Formato:</b> Guía de Práctica de Laboratorio / Talleres / Centros de Simulación</p>		
<p>Aprobación: 2022/03/01</p>	<p>Código: GUIA-PRLE-001</p>	<p>Página: 2</p>

### Cuenta Jose Gordillo:



**JoseGordilloMendoza (JoseGordilloMendoza)**  
Your personal account


- Public profile
- Account
- Appearance
- Accessibility
- Notifications

### Emails

**jgordillome@unsa.edu.pe** – Primary ⓘ

- Not visible in emails ⓘ
- Receives notifications ⓘ

### Cuenta de Gabriela Borda:



**gabrielaborda (gabrielaborda)**  
Your personal account


- Public profile
- Account
- Appearance
- Accessibility
- Notifications

### Emails

**gborda@unsa.edu.pe** – Primary ⓘ

- Not visible in emails ⓘ
- Receives notifications ⓘ

### Cuenta de Ower Lopez



**OwerFrankLopezArela (OwerFrankLopezArela)**  
Your personal account

- Public profile
- Account
- Appearance
- Accessibility
- Notifications

### Emails

**olopeza@unsa.edu.pe** – Primary ⓘ

- Not visible in emails ⓘ
- Receives notifications ⓘ

	<p>UNIVERSIDAD NACIONAL DE SAN AGUSTIN FACULTAD DE INGENIERÍA DE PRODUCCIÓN Y SERVICIOS ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMA</p>	
<p>Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación</p>		
<p>Aprobación: 2022/03/01</p>	<p>Código: GUIA-PRLE-001</p>	<p>Página: 3</p>

- Cree un nuevo proyecto personal y desarrolle el ejercicio resuelto en clase. Hace 3 commits como mínimo y muéstrellos. Commit para "¡Hola mundo!", otro para "Bienvenida al curso" y otro para imprimir su nombre.

*Primero creamos un nuevo repositorio en GitHub.*



*Luego, en la máquina personal creamos el archivo e inicializamos el repositorio que luego subiremos a la nube.*

```
Jose Alonzo@DESKTOP-EVNIKIE MINGW64 ~/Desktop/lab01/labEDA
$ cat > programa.java

Jose Alonzo@DESKTOP-EVNIKIE MINGW64 ~/Desktop/lab01/labEDA
$ git init
Initialized empty Git repository in C:/Users/Jose Alonzo/Desktop/lab01/labEDA/.git/
```

```
Jose Alonzo@DESKTOP-EVNIKIE MINGW64 ~/Desktop/lab01/labEDA (master)
$ git add programa.java
```

```
Jose Alonzo@DESKTOP-EVNIKIE MINGW64 ~/Desktop/lab01/labEDA (master)
$ git commit -m "creacion archivo java"
[master (root-commit) 7dbb748] creacion archivo java
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 programa.java
```

```
Jose Alonzo@DESKTOP-EVNIKIE MINGW64 ~/Desktop/lab01/labEDA (master)
$ git log
commit 7dbb7484b420da05a9c18eac41e53dbb77b26715 (HEAD -> master)
Author: Jose Gordillo Mendoza <jgordillome@unsa.edu.pe>
Date: Sun May 7 11:26:46 2023 -0500
```

```
creacion archivo java
```

*Luego haremos un commit para el “Hola mundo”, primero editando el archivo que creamos:*

```
labEDA > J programa.java
1 public class programa{
2     public static void main(String[] args){
3         System.out.println("Hola mundo");
4     }
5 }

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

Jose Alonzo@DESKTOP-EVNIKIE MINGW64 ~/Desktop/lab01/labEDA (master)
$ git add programa.java

Jose Alonzo@DESKTOP-EVNIKIE MINGW64 ~/Desktop/lab01/labEDA (master)
$ git commit -m "commit de 'Hola mundo'"
[master 8685918] commit de 'Hola mundo'
1 file changed, 5 insertions(+)

Jose Alonzo@DESKTOP-EVNIKIE MINGW64 ~/Desktop/lab01/labEDA (master)
$ git log --oneline
8685918 (HEAD -> master) commit de 'Hola mundo'
7dbb748 creacion archivo java
```

*Veamos el otro commit:*

```
labEDA > J programa.java
1 public class programa{
2     public static void main(String[] args){
3         System.out.println("Hola mundo");
4         System.out.println("Bienvenid@ al curso");
5     }
6 }

Jose Alonzo@DESKTOP-EVNIKIE MINGW64 ~/Desktop/lab01/labEDA (master)
$ git add programa.java

Jose Alonzo@DESKTOP-EVNIKIE MINGW64 ~/Desktop/lab01/labEDA (master)
$ git commit -m "Commt para la bienvenida al curso"
[master 7723b68] Commt para la bienvenida al curso
1 file changed, 1 insertion(+)

Jose Alonzo@DESKTOP-EVNIKIE MINGW64 ~/Desktop/lab01/labEDA (master)
$ git log --oneline
7723b68 (HEAD -> master) Commt para la bienvenida al curso
8685918 commit de 'Hola mundo'
7dbb748 creacion archivo java
```

*Último commit:*

```
labEDA > J programajava
1 public class programa{
2     public static void main(String[] args){
3         System.out.println("Hola mundo");
4         System.out.println("Bienvenid@ al curso");
5         System.out.println("Nombre: Jose Alonzo Gordillo Medndoza");
6     }
7 }
```

```
Jose Alonzo@DESKTOP-EVNIKIE MINGW64 ~/Desktop/lab01/labEDA (master)
$ git add .
```

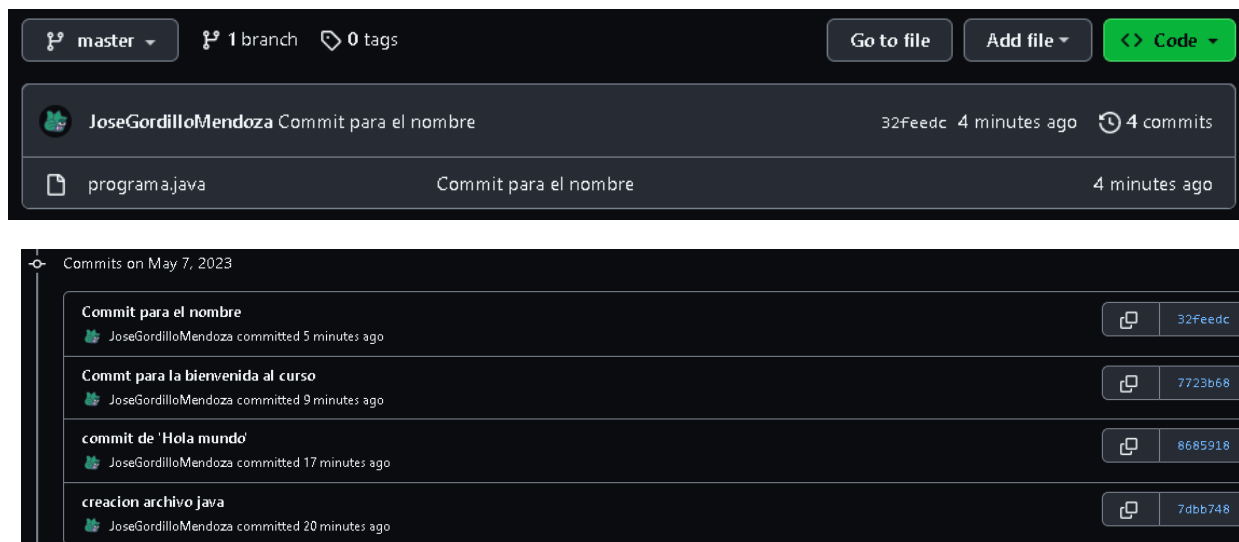
```
Jose Alonzo@DESKTOP-EVNIKIE MINGW64 ~/Desktop/lab01/labEDA (master)
$ git commit -m "Commit para el nombre"
[master 32feedc] Commit para el nombre
1 file changed, 1 insertion(+)
```

```
Jose Alonzo@DESKTOP-EVNIKIE MINGW64 ~/Desktop/lab01/labEDA (master)
$ git log --oneline
32feedc (HEAD -> master) Commit para el nombre
7723b68 Commt para la bienvenida al curso
8685918 commit de 'Hola mundo'
7dbb748 creacion archivo java
```

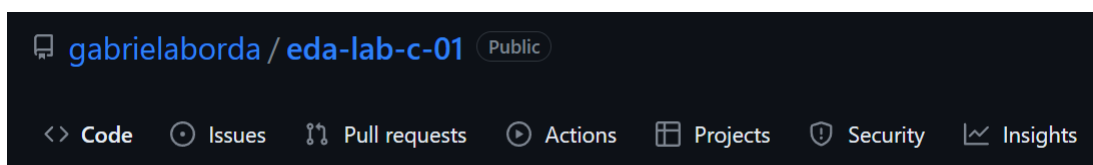
*Con todos los cambios subiremos el archivo a GitHub:*

```
Jose Alonzo@DESKTOP-EVNIKIE MINGW64 ~/Desktop/lab01/labEDA (master)
$ git remote add origin https://github.com/JoseGordilloMendoza/lab01EDA.git
```

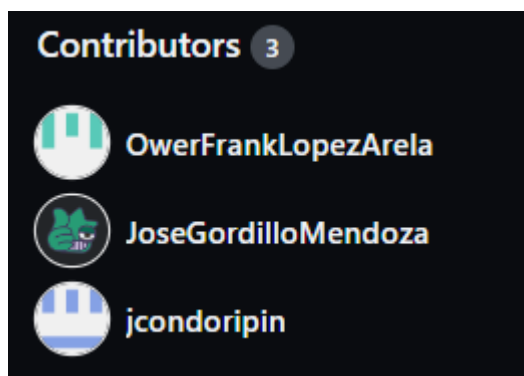
```
Jose Alonzo@DESKTOP-EVNIKIE MINGW64 ~/Desktop/lab01/labEDA (master)
$ git push origin -u master
Enumerating objects: 12, done.
Counting objects: 100% (12/12), done.
Delta compression using up to 8 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (12/12), 1.09 KiB | 1.09 MiB/s, done.
Total 12 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), done.
To https://github.com/JoseGordilloMendoza/lab01EDA.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
```



- Cree un proyecto grupal para trabajo colaborativo (de 3 a 5 integrantes).

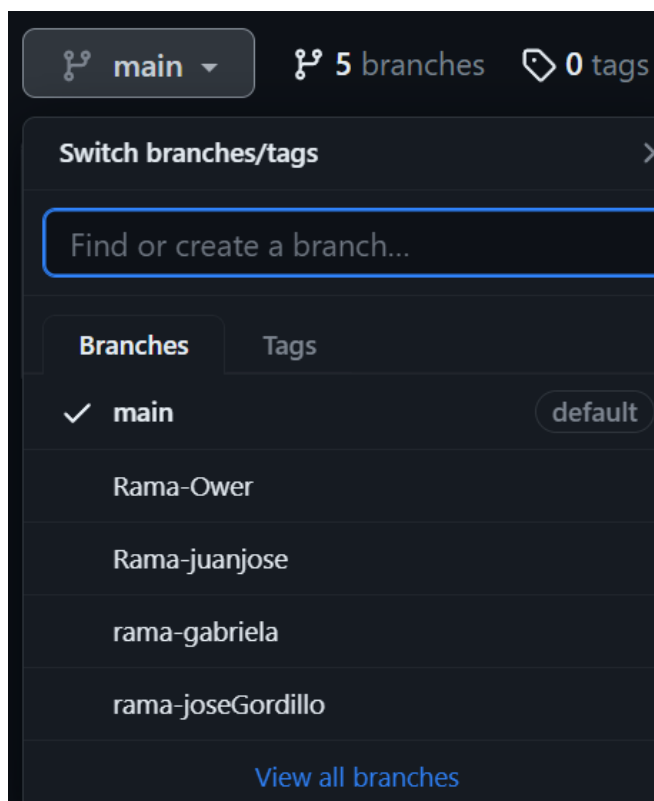


*Cada integrante es agregado como colaborador, excluyendo al dueño del repositorio, es decir Gabriela Borda en nuestro caso:*

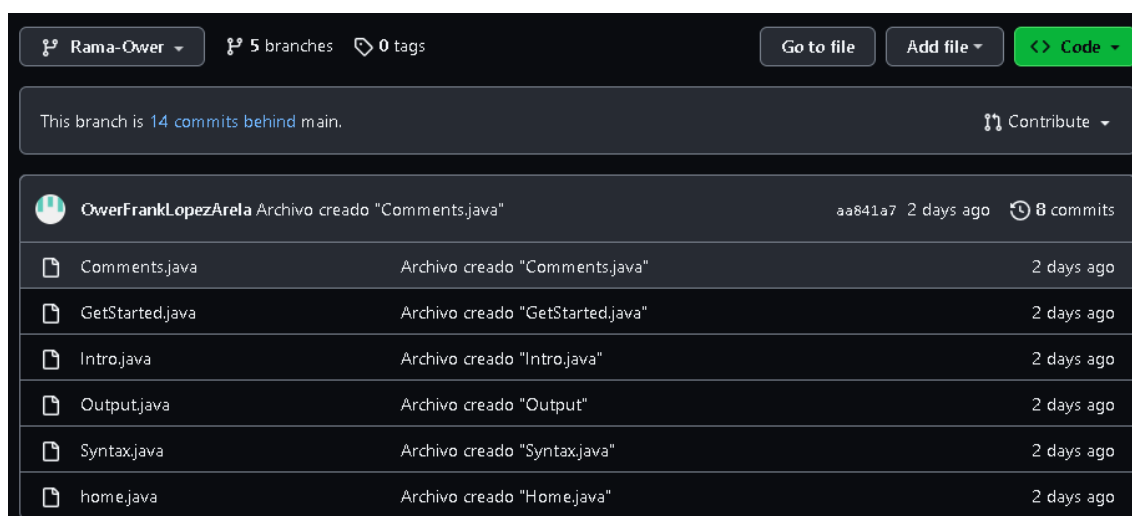


- Cree un archivo por cada tema del manual de java (<https://www.w3schools.com/java/default.asp>), haga commit e incluyalo en su informe grupal (Dividanse los temas).
  - o Java Tutorial
  - o Java Methods
- Cree ramas para cada integrante y cada cierto tiempo una las ramas al main. No elimine nada para evidenciar ramas, main y commits.

*Son un total de 4 ramas adicionales, una por integrante:*

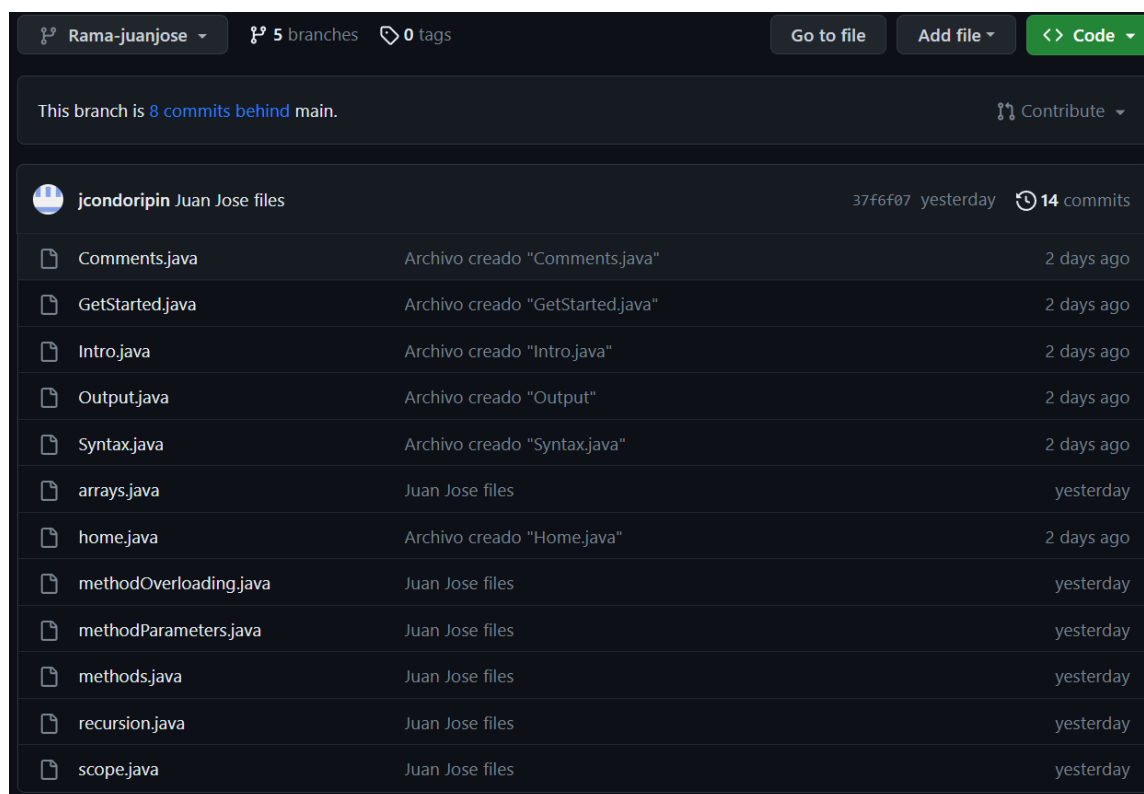


*Primero veamos la rama del compañero Ower que creó su rama a partir de la “línea de tiempo” en donde la rama principal no cuenta con archivos:*



*En todo caso, al momento de unir esta rama con la principal, el main contiene estos archivos, esto GitHub nos lo indica, indicando el merge con un color morado.*

*La siguiente rama contiene los siguientes 6 archivos del tutorial de Java:*



File Name	Description	Last Commit
Comments.java	Archivo creado "Comments.java"	2 days ago
GetStarted.java	Archivo creado "GetStarted.java"	2 days ago
Intro.java	Archivo creado "Intro.java"	2 days ago
Output.java	Archivo creado "Output"	2 days ago
Syntax.java	Archivo creado "Syntax.java"	2 days ago
arrays.java	Juan Jose files	yesterday
home.java	Archivo creado "Home.java"	2 days ago
methodOverloading.java	Juan Jose files	yesterday
methodParameters.java	Juan Jose files	yesterday
methods.java	Juan Jose files	yesterday
recursion.java	Juan Jose files	yesterday
scope.java	Juan Jose files	yesterday


























*Como podemos observar, esta rama posee los 6 archivos de la anterior rama, los cuales fueron extraídos mediante un pull, para luego añadir los 6 nuevos y realizar un push en la nueva rama y unirla al main del repositorio remoto.*

*Entonces a la hora de la siguiente rama, después de hacer uso del comando git pull, actualizamos el contenido, donde ahora cuenta con los archivos de las otras dos ramas abordadas, cabe recalcar que el pull es muy importante, ya que no es conveniente clonar a cada momento dependiendo de los cambios, aunque podría ser una alternativa por si no se manejan de manera correcta los comandos en git.*

*Ahora bien, veamos la 3er rama, donde se incluyen otros 6 archivos pertenecientes a lo que se indicó*



*Vemos que la cantidad ahora ha aumentado, esta rama se sitúa en el momento en el que ya existen 12 archivos, por lo que al agregar/unir ramas, los otros 6 archivos, se nos forma la lista de archivos anterior.*

rama-gabriela 5 branches 0 tags			Go to file	Add file +	Code >
This branch is 1 commit behind main.			Contribute >		
 gabrielaborda Temas 07 - 12			81525cd 1 minute ago 23 commits		
	Comments.java	Archivo creado "Comments.java"	2 days ago		
	GetStarted.java	Archivo creado "GetStarted.java"	2 days ago		
	Intro.java	Archivo creado "Intro.java"	2 days ago		
	Output.java	Archivo creado "Output"	2 days ago		
	Syntax.java	Archivo creado "Syntax.java"	2 days ago		
	arrays.java	Juan Jose files	yesterday		
	booleans.java	base booleans.java	yesterday		
	breakContinue.java	base "breakContinue.java"	yesterday		
	dataTypes.java	Temas 07 - 12	1 minute ago		
	forLoop.java	base "forLoop.java"	yesterday		
	home.java	Archivo creado "Home.java"	2 days ago		
	ifElse.java	base "ifElse.java"	yesterday		
	math.java	Temas 07 - 12	1 minute ago		
	methodOverloading.java	Juan Jose files	yesterday		
	methodParameters.java	Juan Jose files	yesterday		
	methods.java	Juan Jose files	yesterday		
	operators.java	Temas 07 - 12	1 minute ago		
	recursion.java	Juan Jose files	yesterday		
	scope.java	Juan Jose files	yesterday		
	strings.java	Temas 07 - 12	1 minute ago		
	switch.java	base "switch.java"	yesterday		
	typeCasting.java	Temas 07 - 12	1 minute ago		
	variables.java	Temas 07 - 12	1 minute ago		
	whileLoop.java	base "whileLoop.java"	yesterday		

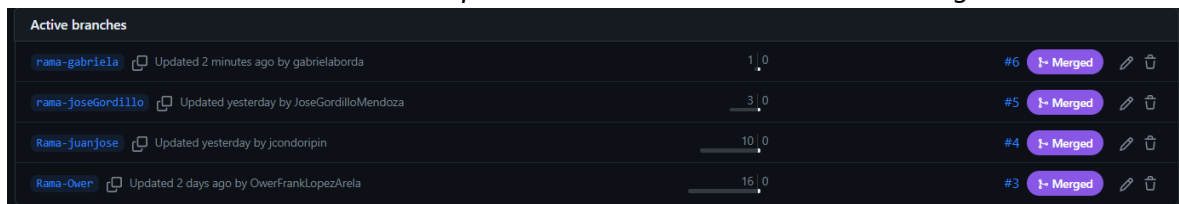
*En esta rama se encuentran los últimos 6 archivos del tutorial de java, lo que significa que esta rama se crea con 18 archivos existentes.*

<span>main</span> <span>5 branches</span> <span>0 tags</span> <span>Go to file</span> <span>Add file</span> <span>Code</span>		
<span>gabrielaborda</span> Merge pull request #6 from gabrielaborda/rama-gabriela <span>55adsab now</span> <span>24 commits</span>		
	Comments.java	Archivo creado "Comments.java" 2 days ago
	GetStarted.java	Archivo creado "GetStarted.java" 2 days ago
	Intro.java	Archivo creado "Intro.java" 2 days ago
	Output.java	Archivo creado "Output" 2 days ago
	Syntax.java	Archivo creado "Syntax.java" 2 days ago
	arrays.java	Juan Jose files yesterday
	booleans.java	base booleans.java yesterday
	breakContinue.java	base "breakContinue.java" yesterday
	dataTypes.java	Temas 07 - 12 2 minutes ago
	forLoop.java	base "forLoop.java" yesterday
	home.java	Archivo creado "Home.java" 2 days ago
	ifElse.java	base "ifElse.java" yesterday
	math.java	Temas 07 - 12 2 minutes ago
	methodOverloading.java	Juan Jose files yesterday
	methodParameters.java	Juan Jose files yesterday
	methods.java	Juan Jose files yesterday
	operators.java	Temas 07 - 12 2 minutes ago
	recursion.java	Juan Jose files yesterday
	scope.java	Juan Jose files yesterday
	strings.java	Temas 07 - 12 2 minutes ago
	switch.java	base "switch.java" yesterday
	typeCasting.java	Temas 07 - 12 2 minutes ago
	variables.java	Temas 07 - 12 2 minutes ago
	whileLoop.java	base "whileLoop.java" yesterday

*Como podemos ver,todos los archivos fueron enlazados a la rama main, esto a través del repositorio remoto y la instrucción Pull Request.*

	<p style="text-align: center;"><b>UNIVERSIDAD NACIONAL DE SAN AGUSTIN</b>  <b>FACULTAD DE INGENIERÍA DE PRODUCCIÓN Y SERVICIOS</b>  <b>ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMA</b></p>	
<p style="text-align: center;"><b>Formato:</b> Guía de Práctica de Laboratorio / Talleres / Centros de Simulación</p>		
<p><b>Aprobación:</b> 2022/03/01</p>	<p><b>Código:</b> GUIA-PRLE-001</p>	<p><b>Página:</b> 12</p>

*Por último veamos que todas las ramas han hecho un merge:*



<https://github.com/gabrielaborda/eda-lab-c-01.git>

(Repositorio remoto)

## 2. SOLUCIÓN DEL CUESTIONARIO

- **¿Por qué Git y GitHub son herramientas importantes para el curso?**

Porque son herramientas que permiten guardar y mantener un seguimiento de las versiones de un trabajo ya sea tanto grupal como individual.

Por parte de git tendremos la herramienta de versionado en nuestra máquina, para nuestro propio avance y configuración dentro de parámetros locales.

Por parte de github nos permitirá subir nuestro trabajo a la nube ya sea para un trabajo grupal o simplemente para presentarlo al público, además de que al estar en la nube tenemos un grado de seguridad al guardar nuestro trabajo pues ya no dependerá solo de nuestro propio computador.

- **¿Qué conductas éticas deberían promocionarse cuando se usa un Sistema de Control de Versiones?**

Las conductas deberían ser:

- Respeto por los derechos de autor: No usar ni compartir código, archivos u otros recursos sin el debido permiso del propietario.
- Respetar la propiedad intelectual: No copiar ni plagiar el trabajo de otra persona sin dar el debido crédito u obtener los permisos necesarios.
- Ser claro en los cambios: proporcionar descripciones precisas y claras de los cambios realizados en cada confirmación.
- Respetar las políticas y lineamientos del proyecto: apegarse a las reglas establecidas por el proyecto o el equipo, como el flujo de trabajo, las convenciones de nomenclatura y las pautas de contribución.

- **¿Qué son los estándares de codificación?**

Son pautas o normas establecidas para escribir código en un lenguaje de programación particular. Estas reglas describen una serie de convenciones y prácticas que deben seguirse para mantener la coherencia, la legibilidad y la calidad del código en un proyecto u organización.

Los estándares de codificación pueden afectar la estructura, la denominación de variables y funciones, la organización de la codificación, la documentación y el uso de comentarios, entre

	<p style="text-align: center;"><b>UNIVERSIDAD NACIONAL DE SAN AGUSTIN</b>  <b>FACULTAD DE INGENIERÍA DE PRODUCCIÓN Y SERVICIOS</b>  <b>ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMA</b></p>	
<p style="text-align: center;"><b>Formato:</b> Guía de Práctica de Laboratorio / Talleres / Centros de Simulación</p>		
<p><b>Aprobación:</b> 2022/03/01</p>	<p><b>Código:</b> GUIA-PRLE-001</p>	<p><b>Página:</b> 13</p>

otros aspectos del código. Estas pautas facilitan a los programadores la comprensión y el trabajo conjunto en proyectos, así como el mantenimiento y la evolución del código a lo largo del tiempo.

### 3. CONCLUSIONES

- Con Git tenemos una gestión eficiente de versiones, ya que nos permite realizar un seguimiento completo de los cambios realizados en su proyecto a lo largo del tiempo.
- Git permite el desarrollo de manera muy buena el trabajo en equipo y colaboración, pues facilita la colaboración en proyectos entre diferentes miembros del equipo. Varios desarrolladores pueden trabajar y luego fusionar sus cambios.
- Git permite el desarrollo y control de sucursales y ramificaciones, al permitirnos crear sucursales independientes para desarrollar nuevas funciones o solucionar problemas sin afectar la sucursal principal.
- Aprovechamos la flexibilidad y velocidad de git, que nos ayuda en la programación al ser rápido y eficiente en el manejo de grandes cantidades de código y archivos.
- Git permite la búsqueda a través de un historial de cambios, lo que facilita el desarrollo continuo a través de distintas etapas de nuestro proyecto.

## RETROALIMENTACIÓN GENERAL

## REFERENCIAS Y BIBLIOGRAFÍA

- <https://www.atlassian.com/es/git/glossary> - Enlace empleado en la resolución de los ejercicios
- <https://docs.github.com/es/pull-requests/collaborating-with-pull-requests/incorporating-changes-from-a-pull-request/merging-a-pull-request> - Información útil al momento de unir ramas en la resolución de los ejercicios
- Enlaces proporcionados por el docente:
  - <https://git-scm.com/book/es/v2>
  - <https://guides.github.com/>
  - <https://www.w3schools.com/java/default.asp>
- <https://www.w3schools.com/java/default.asp> - Enlace base para la creación de los archivos