

Armando Molina Rincón – Examen 3

1. Describe un escenario donde creaste una rama más además de la master y modificaste el mismo archivo. Describe los pasos para solucionar el problema al hacer merge

- **git init** - Para inicializar el repositorio
- **touch notas.txt** - Ver el contenido de notas.txt
- escribir en el archivo
- **cat notas.txt** - Ver el contenido de notas.txt

```
USER@DESKTOP-NRTG9M9 MINGW64 ~/Documents/Desarrollo/solucionMerge (master)
$ cat notas.txt
15/08/22 Se realizó el programa hola_mundo.py
```

- **git add .** - Añadir el archivo notas.txt al index
- **git status** - Ver el estado de los archivos (si fueron modificados o no)

```
USER@DESKTOP-NRTG9M9 MINGW64 ~/Documents/Desarrollo/solucionMerge (master)
$ git add .

USER@DESKTOP-NRTG9M9 MINGW64 ~/Documents/Desarrollo/solucionMerge (master)
$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   notas.txt

USER@DESKTOP-NRTG9M9 MINGW64 ~/Documents/Desarrollo/solucionMerge (master)
$ |
```

- **git commit -m "primerCommit"** - Hacer un commit

Crear una nueva rama

- **git branch nuevaRama** - Crear una nueva rama
- **git branch** - Ver las ramas del proyecto
- **git switch nuevaRama** - Cambiar el área de trabajo a nuevaRama
- **git branch** - Ver las ramas del proyecto

```

USER@DESKTOP-NRTG9M9 MINGW64 ~/Documents/Desarrollo/solucionMerge (master)
$ git branch nuevaRama

USER@DESKTOP-NRTG9M9 MINGW64 ~/Documents/Desarrollo/solucionMerge (master)
$ git branch
* master
  nuevaRama

USER@DESKTOP-NRTG9M9 MINGW64 ~/Documents/Desarrollo/solucionMerge (master)
$ git switch nuevaRama
Switched to branch 'nuevaRama'

USER@DESKTOP-NRTG9M9 MINGW64 ~/Documents/Desarrollo/solucionMerge (nuevaRama)
$ git branch
  master
* nuevaRama

USER@DESKTOP-NRTG9M9 MINGW64 ~/Documents/Desarrollo/solucionMerge (nuevaRama)
$ |

```

Modificar el archivo

- **cat notas.txt** - Ver el contenido de notas.txt

```

USER@DESKTOP-NRTG9M9 MINGW64 ~/Documents/Desarrollo/solucionMerge (nuevaRama)
$ cat notas.txt
15/08/22 Se realizó el programa bubble_sort_2.py

```

- **git add notas.txt** - Agregar notas.txt al index
- **git commit -m "primerCommitNuevaRama"** - Hacer un commit

Merge:

- **git switch master** - Regresar a la rama master
- **git merge nuevaRama** - Unir el contenido de ambas ramas. Muestra un error ya que Git no sabe que contenido conservar

```

USER@DESKTOP-NRTG9M9 MINGW64 ~/Documents/Desarrollo/solucionMerge (master)
$ git merge nuevaRama
Auto-merging notas.txt
CONFLICT (content): Merge conflict in notas.txt
Automatic merge failed; fix conflicts and then commit the result.

```

- **cat notas.txt** - Ver el contenido de notas.txt
- Modificar el archivo
- **cat notas.txt** - Ver el contenido de notas.txt

```

USER@DESKTOP-NRTG9M9 MINGW64 ~/Documents/Desarrollo/solucionMerge (master|MERGI
NG)
$ cat notas.txt
<<<<<<< HEAD
15/08/22 Se realizó el programa hola_mundo.py
=====
15/08/22 Se realizó el programa bubble_sort_2.py
>>>>>> nuevaRama

USER@DESKTOP-NRTG9M9 MINGW64 ~/Documents/Desarrollo/solucionMerge (master|MERGI
NG)
$ cat notas.txt
15/08/22 Se realizó el programa hola_mundo.py
15/08/22 Se realizó el programa bubble_sort_2.py

```

- **git add notas.txt** - Añadir el archivo al área de index
- **git commit -m "mergeResuelto"** - Crear un commit en la rama master, donde se encuentra el archivo resuelto
- **git branch -d nuevaRama** - Eliminar la rama nuevaRama
- **git branch** - Ver las ramas del proyecto para verificar que se haya eliminado la otra rama

```

USER@DESKTOP-NRTG9M9 MINGW64 ~/Documents/Desarrollo/solucionMerge (master)
$ git branch -d nuevaRama
Deleted branch nuevaRama (was 042f31c).

USER@DESKTOP-NRTG9M9 MINGW64 ~/Documents/Desarrollo/solucionMerge (master)
$ git branch
* master

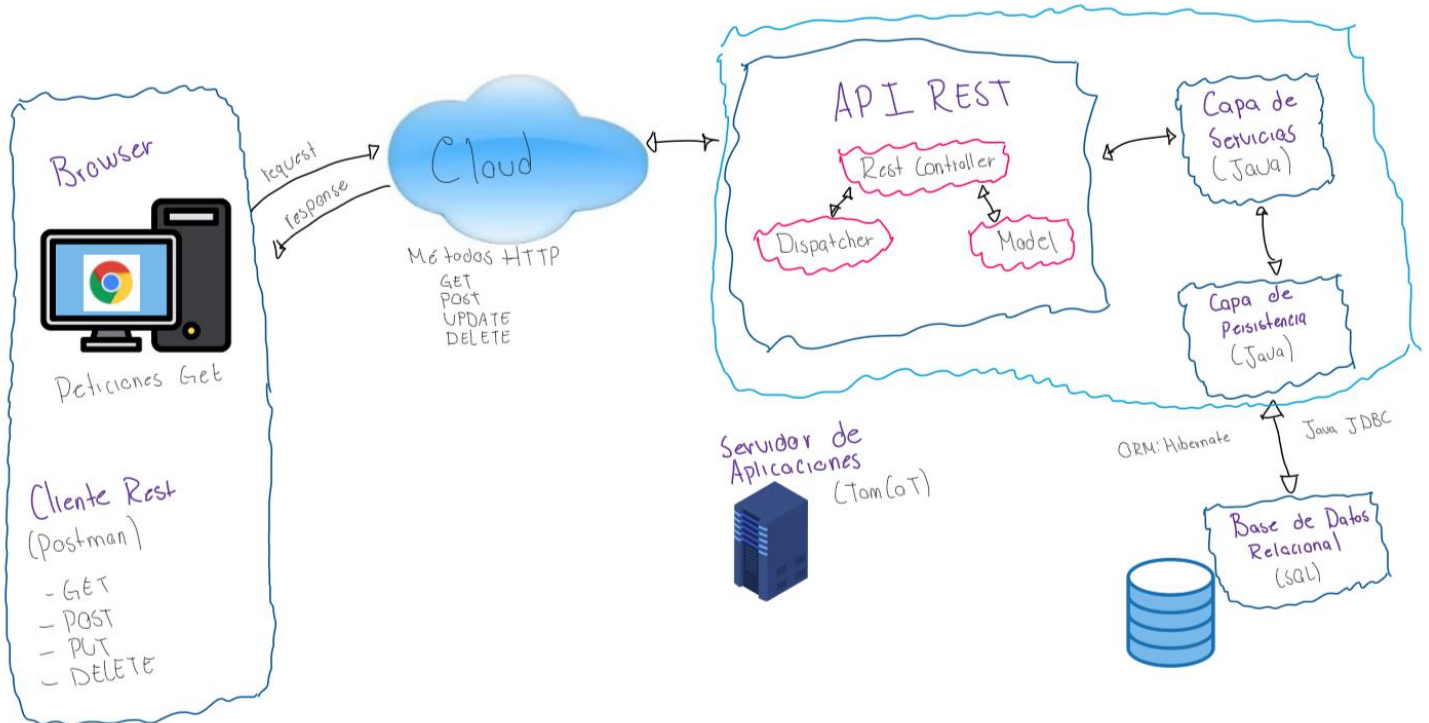
```

2. Realizar ejercicio de calculadora utilizando lambdas.

<https://github.com/Amolin10/BecaXideralExamenes>

Examen3Calc

3. Dibuja el diagrama de una aplicación de servicios REST



4. Explica en que consiste la inyección de dependencias

Se refiere a la inversión de responsabilidad con respecto a cómo el objeto obtiene las referencias de los objetos con los que colabora.

Consiste en el principio de pasar (inyectar) valores a los objetos cuando sea necesario y que no sean ellos mismos quienes tienen la responsabilidad de inicializarlos.

En el paradigma Orientado a Objetos existen 3 tipos de implementar la inyección de dependencias:

- **Inyección por una variable**
Un objeto cumple con la responsabilidad de asignar (inyectar) el valor de uno o más atributos a otro objeto.
- **Inyección por un setter**
Un objeto posee un método setter, a través del cual se le puede asignar el valor de un atributo.
- **Inyección por un constructor**
Al crear un objeto, se le indica a través de su constructor el valor de sus atributos.

La inyección de dependencias también es muy utilizada por los Frameworks, de esta manera, inyectando o añadiendo funcionalidades conforme sean requeridas. Pueden verse como módulos que se añaden a una base funcional.

5. **Implementa el proyecto CRUD utilizando SpringBoot, considera tanto la implementación de Hibernate como de JDBC para la base de datos.**

<https://github.com/Amolin10/BecaXideralExamenes>

SpringBootRestHibernateJdbc