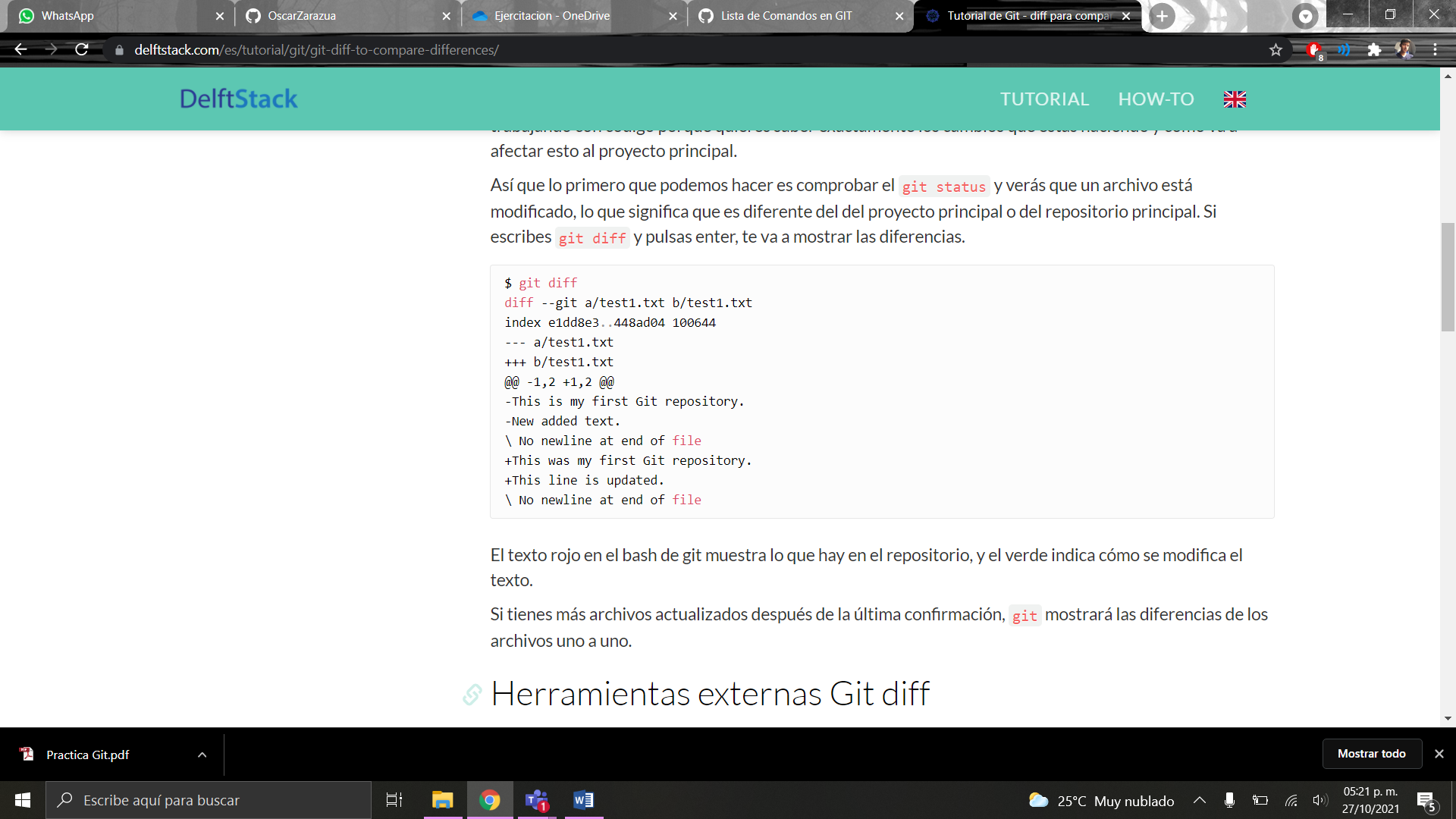
Practica Git

1. Investigar 2 comandos no vistos en clase (Ej: Git rebase, Git Fetch)

**Git Diff**:

Una vez que hemos guardado un fichero, este fichero en una copia de trabajo es diferente del que está en nuestro repositorio. ¿Pero cómo vemos exactamente estos cambios? Porque a veces hacemos algunos cambios en un fichero y queremos ver el que está en el repositorio. Así que lo primero que podemos hacer es comprobar el git status y verás que un archivo está modificado, lo que significa que es diferente del del proyecto principal o del repositorio principal. Si escribes git diff y pulsas enter, te va a mostrar las diferencias.



**Git Tag**

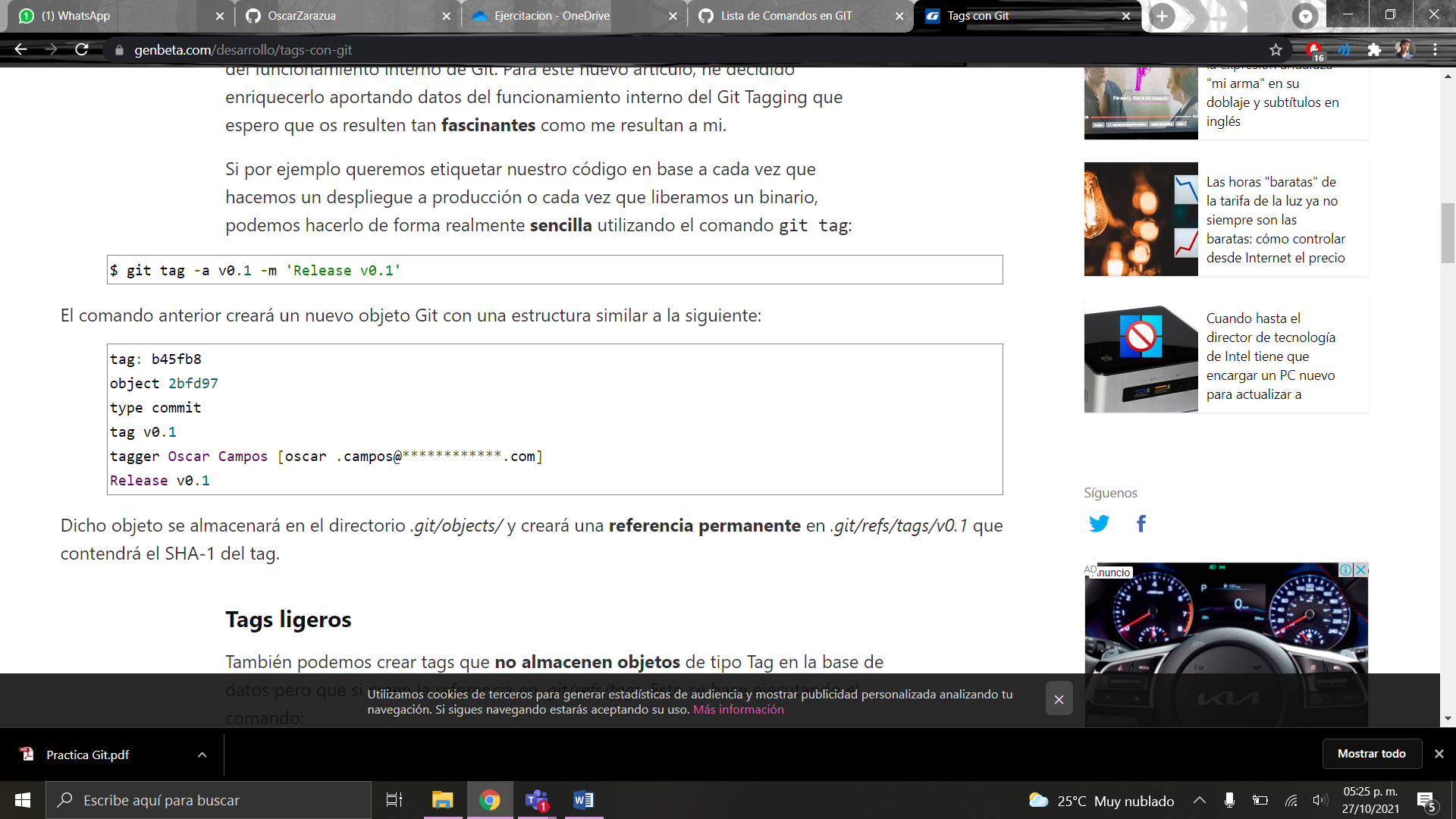
En Git, una etiqueta o tag sirve básicamente como una rama firmada que no permuta, es decir, siempre se mantiene inalterable. Sencillamente es una cadena arbitraria que apunta a un commit específico. Puede decirse que un tag es un nombre que puedes usar para marcar un punto específico en la historia de un repositorio.

Los tags pueden ser muy útiles para marcar hitos de progreso sobre todo si trabajamos con metodologías de desarrollo ágil.

Si por ejemplo queremos etiquetar nuestro código en base a cada vez que hacemos un despliegue a producción o cada vez que liberamos un binario, podemos hacerlo de forma realmente sencilla utilizando el comando git tag:



El comando anterior creará un nuevo objeto Git con una estructura similar a la siguiente:



Dicho objeto se almacenará en el directorio .git/objects/ y creará una referencia permanente en .git/refs/tags/v0.1 que contendrá el SHA-1 del tag.

1. Investigar Git Fork y cómo funciona

**Fork**

Cuando hacemos un fork de un repositorio, se hace una copia exacta en crudo (en inglés «bare») del repositorio original que podemos utilizar como un repositorio git cualquiera.

Cuando haces un fork de un repositorio, se crea un nuevo repositorio en tu cuenta de Github o Bitbucket, con una URL diferente (fork). Acto seguido tienes que hacer un clon de esa copia sobre la que empiezas a trabajar de forma que cuando haces push, estás modificando TU COPIA (fork). El repositorio original sigue intacto.

¿Para qué sirve Fork?

Permite a los desarrolladores contribuir a un proyecto de forma segura.

Crear un archivo de texto con la investigación realizada y subirla a su repositorio en GitHub utilizando los pasos mencionados en clase y a través de un pull request (puede ser un PR a Master solo en esta ocasión)