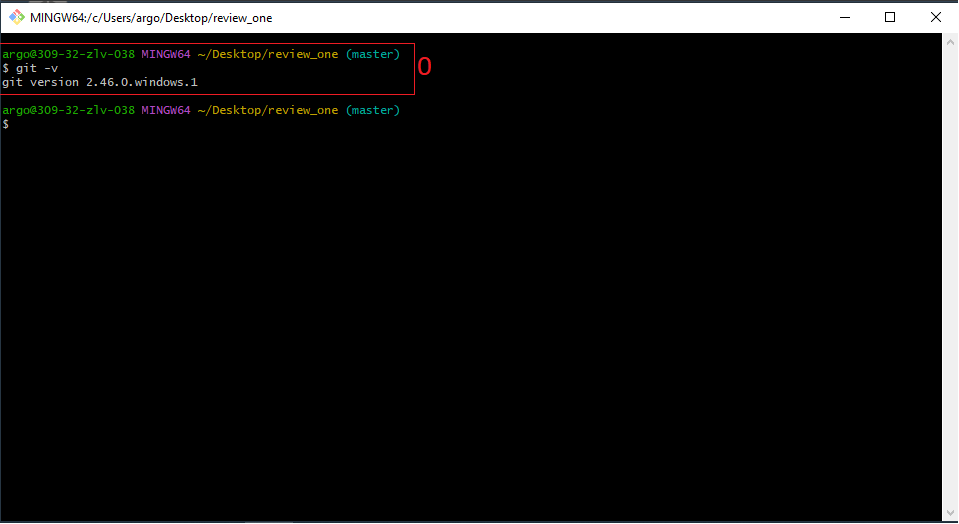
# T0. Activitats de repàs I

1. **Crea la carpeta de projecte “review\_one” en la teva unitat i afegeix-la com a repositori de git.**

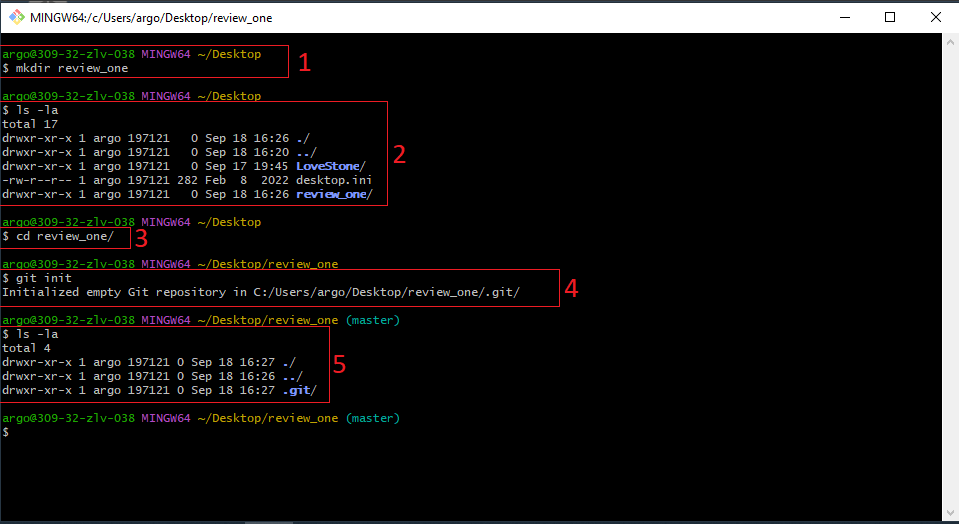
**PASOS**

1. Para el correcto desarrollo de la actividad, primero verificamos si tenemos instalado Git en nuestro computador, para esto ejecutamos el comando mostrado en la terminal el cuál nos dará la versión instalada de Git, en caso de no reconocer git como un comando significa que tenemos que realizar el proceso de instalacion.



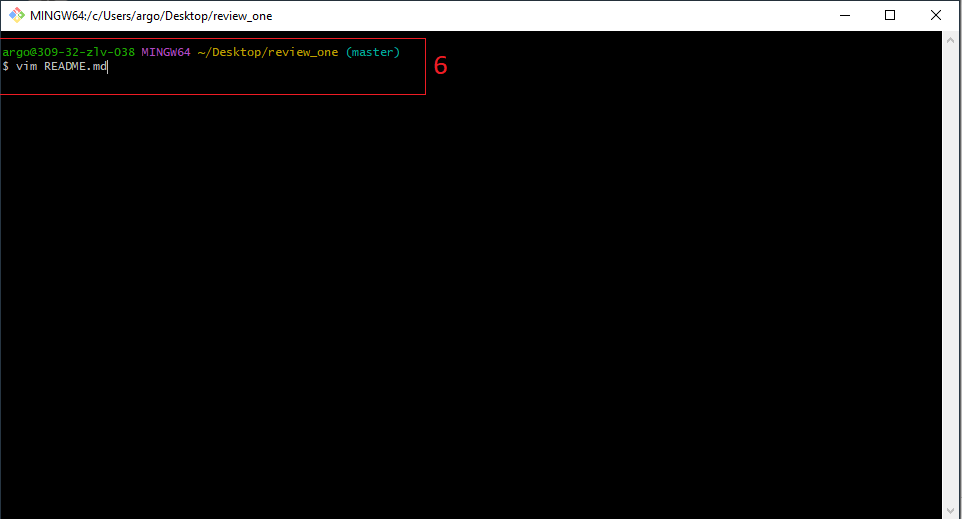
Una vez descargado Git en nuestro computador, veremos que nos descarga algunas dependencias para su correcto funcionamiento y una de ellas es la terminal Git bash, en esta terminal tendremos un entorno muy parecido a la terminal de Linux por lo que, si ya hemos usado linux previamente nos resultará amigable en cuanto a comandos e interfaz

1. Se crea la carpeta “review\_one” mediante comandos (mkdir <nombre\_directorio>).
2. Se verifica que la carpeta se haya creado exitosamente, esto se realiza mediante el comando “ls –la” el cuál nos generará una lista detallada de los directorios y archivos existentes en nuestra ubicacion
3. Se accede a la carpeta creada (a partir de ahora la llamaremos directorio) mediante el comando “cd” el cual nos permite cambiar de ruta
4. Usando el comando “git init” generamos los archivos necesarios para que el Git bash reconozca el directorio como un repositorio Git, en este caso se crea el directorio “.git”.
5. Si usamos de nuevo el comando “ls –la” podremos ver el directorio creado .git. De igual manera un indicativo de que creamos un repositorio es que despues de la ruta en la que estamos ubicados aparecerá un “(master)”.

****

**2. Crea dins el repositori el document README.md amb el contingut següent:**

|  |
| --- |
| **# Primera activitat**  **Aquest repositori conté les activitats de repàs I del mòdul de projectes** |



**6.** Dentro del paquete que descargamos e instalamos de Git, viene por defecto (si no cambiamos nada al momento de la instalacion) el editor de texto “Vim”. Para usarlo sólo tenemos que poner su nombre seguido del archivo que deseamos abrir, si no existe lo crea y lo abre para su edición.

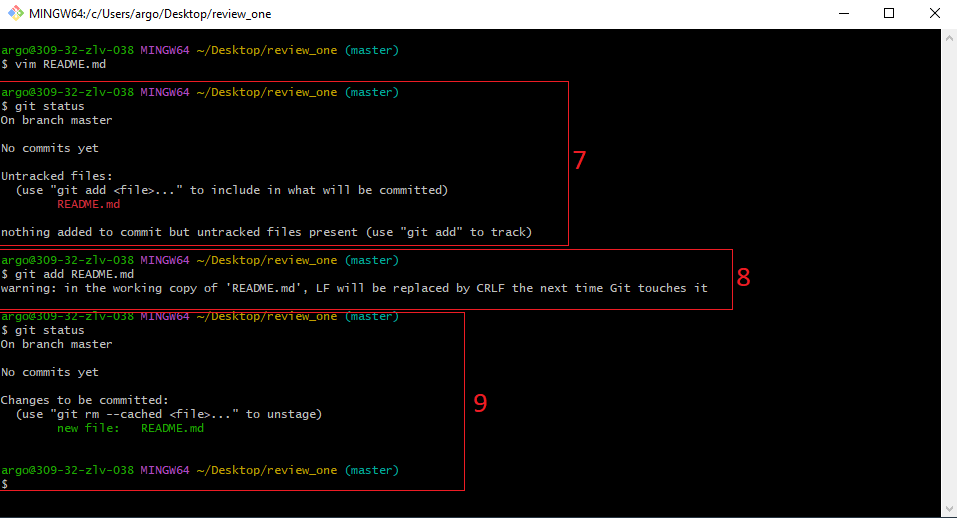


**3. Afegeix-lo al staging area.**

**7.** Para identificar un archivo o directorio que no se haya agregado al staging area, usamos el comando “git status” el cual nos indicará en color rojo lo que nos hace falta para añadir.

**8.** Una vez identificado, en este caso el archivo README.md, lo añadimos con el comando “git add” seguido del nombre del archivo/directorio

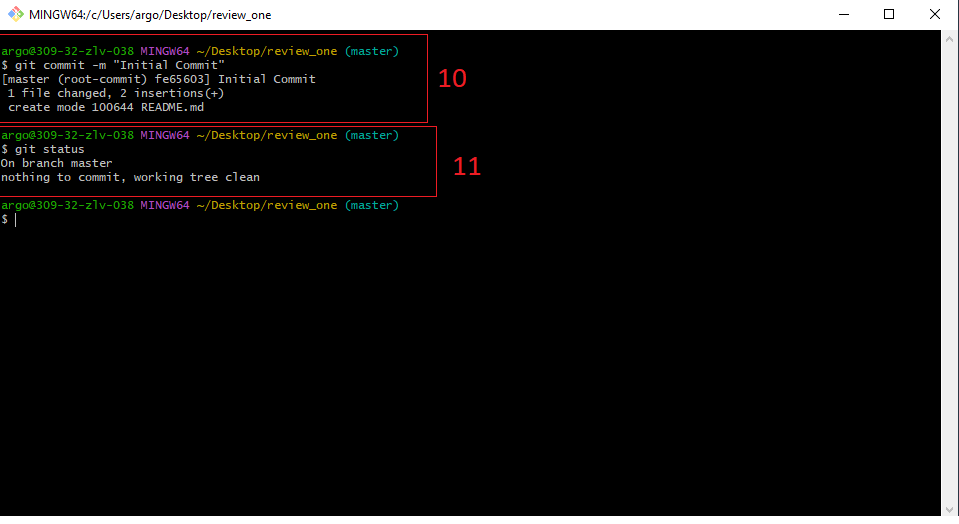
**9.** Si volvemos a usar el git status, este nos indicará en color verde que se ha subido pero hace falta el commit para confirmar



**4. Afegeix-lo a la commit area amb el comentari "Initial commit".**

**10.** Para realizar los commit y “confirmar” los add’s, usamos el comando “git commit –m <Mensaje>”

**11.** Si usamos nuevamente git status veremos que no tenemos nada para realizar commit, este es un buen momento para realizar un push a nuestro repositorio remoto

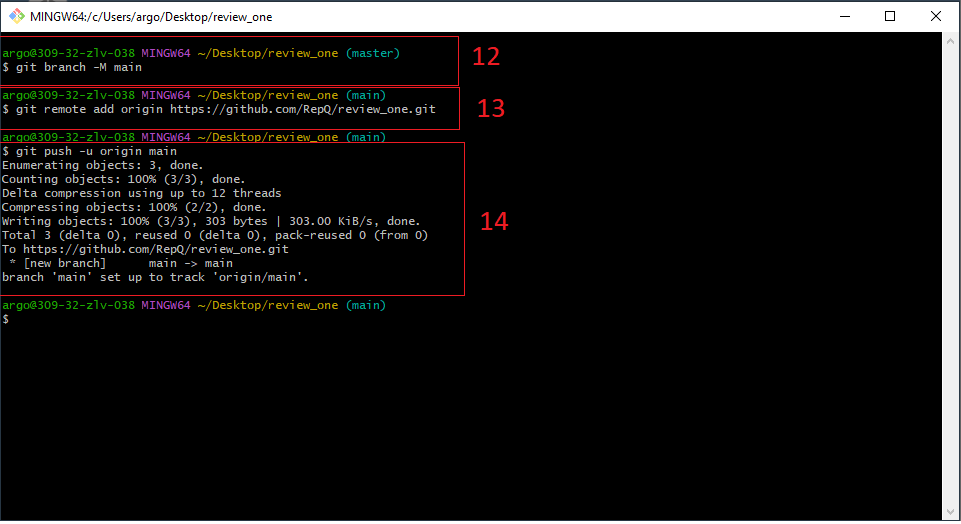


**5. Puja el teu repositori local a un repositori remot (ha de tenir el mateix nom).**

**12.** Usamos “git branch –M main” basicamente para seguir la convencion mas reciente de Git para cambiar de “master” a “main” como rama principal

**13.** Vinculamos el repositorio local a uno remoto. Acá creamos una referencia llamada en este caso origin que apunta a nuestro repositorio remoto (la url)

**14.** Estamos subiendo a nuestro repositorio remoto, vinculado con origin, nuestra rama main creada previamente en el paso 12



**6. Aplica les tècniques de refacció i optimització en el codi següent:**

|  |
| --- |
| using System;  class Program  {  static void Main(string[] args)  {  int nota1 = 0;  int nota2 = 0;  int nota3 = 0;  Console.WriteLine("Introdueix la primera qualificació");  nota1 = Convert.ToInt32(Console.ReadLine());  Console.WriteLine("Introdueix la segona qualificació: ");  nota2 = Convert.ToInt32(Console.ReadLine());  Console.WriteLine("Introdueix la tercera qualificació: ");  nota3 = Convert.ToInt32(Console.ReadLine());  int suma = nota1 + nota2 + nota3;  float promig = suma / 3;  Console.WriteLine("El promig de les notes es: " + promig );  if (promig > 6)  {  Console.WriteLine("Aprovat");  }  else if (promig < 6)  {  Console.WriteLine("Suspès");  }  }  } |

**Crea el projecte dins del mateix repositori i puja’l al repositori remot.**

**7. Afegeix una funcionalitat al programa anterior que permeti emmagatzemar ciutats i els seus respectius codis postals. El programa ha de demanar el nom i el codi fins que l’usuari introdueixi un valor per finalitzar la introducció de dades i mostrar un llistat amb format dels valors introduïts.**