

//PREGUNTAS QUE ES MUY POSIBLE QUE SEAN IGUALES

-En Git la operación pull equivale a:

- a) Fetch+merge
- b) Merge
- c) Push
- d) Rebase

-En el patrón de arquitectura MVC

- a) Un modelo puede tener varias vistas sobre él
- b) Una vista puede tener varias vistas sobre él
- c) Una vista puede referirse más de un modelo siempre
- d) Ninguna de las anteriores

-Los SVC, según la forma de almacenar los contenidos, se clasifican en:

- a) Centralizados y colaborativos
- b) Distribuidos y exclusivos
- c) Centralizados y exclusivos
- d) Centralizados y distribuidos

-Sqlite:

- a) Requiere de un proceso servidor
- b) Requiere de un servidor de un servidor y de un cliente
- c) En la versión 3 (Sqlite3) si requiere de un proceso servidor
- d) Ninguna de las anteriores

-Git guarda toda la meta-información de un proyecto en:

- a) Una base de datos relacional
- b) En un directorio único para cada usuario
- c) En un directorio único por proyecto

d) En un archivo configuración

-Las librerías dinámicas :

- a) No hay que compilarlas
- b) Debe llevar el prefijo lib y la extensión .a
- c) Hace crecer el ejecutable del final
- d) Ninguna de las anteriores

-Git es un sistema de control de versiones:

- A. Centralizado.
- B. Distribuido.
- C. Centralizado en unos casos, distribuido en otros.
- D. Ninguna de las anteriores.

-La operacion commit de Git:

- A. Traslada los datos del repositorio local a la copia maestra del mismo.
- B. Traslada los cambios hechos en la copia de trabajo al repositorio local.
- C. Traslada los cambios hechos en la copia de trabajo a un repositorio remoto.
- D. Clona repositorios.

-La inicializacion de gtk+ es Application.Init();

Gtk+ ejemplo sencillo I

```
// gtkhw.cs
using Gtk;
using System;
class Hello {
    static void Main() {
        Application.Init (); // Iniciacion de la biblioteca Gtk+
```

-El bucle de espera de eventos se inicia con Application.run();

```

Window window = new Window ("helloworld");
window.Show();

Application.Run ();    // Bucle de espera de eventos
}

```

-Gtk + termina con Application.Quit;

```

// runs when the user deletes the window using
// the "close window" widget in the window frame.
static void delete_event (object obj,
                          DeleteEventArgs args) {
    Application.Quit ();
}

```

-Sqlite es para:

- a) C
- b) C#
- c) Las dos
- d) Mas

Compara MonoDevelop y Visual Studio ,

- a) Mismo producto distinto nombre
- b) dos desarrollados por microsoft
- c) Pueden ejecutar el mismo tipo de proyectos
- d) Todas correctas

¿Qué es MonoDevelop?

- Se trata de un IDE ligero multiplataforma (*Linux, Windows y Mac OS*).
- Permite escribir código en diversos lenguajes de programación (C#, F# (1 y 2), C/C++, etc...).
- Dispone de completado de código, plantillas de código y ocultación de bloques de código.
- Tiene depurador a nivel de código fuente así como diseñador de interfaces de aplicaciones de escritorio integrados.
- Es de código abierto y está escrito en C#.
- Permite abrir y trabajar con *soluciones* creadas con VisualStudio en Windows, tanto de escritorio como de ASP.NET.

//FIN DE PREGUNTAS FILTRADAS POR MR.PUCHE

//Preguntas 2016

2. Para modificar el un mensaje de un commit:
 - a) Un mensaje de un commit no se puede modificar
 - b) Utilizamos la opción –amend para modificar el último commit**
 - c) Debemos hace un checkout del archivo y luego usamos amend
 - d) Ninguna de las anteriores
3. En la programación dirigida por eventos, la cola de eventos:
 - a) Es necesaria.
 - b) No es necesaria**
 - c) Depende del lenguaje de programación usado
 - d) Ninguna de las anteriores
4. Para consultar los metadatos de una bb.dd. en sqlite:
 - a) Utilizamos la operación .tables
 - b) Utilizamos la operación .databases
 - c) Consultamos la tabla sqlite_master**
 - d) Ninguna de las anteriores

16. En el interfaz de comandos sqlite para ejecutar en un fichero con comandos sql;
- a) Utilizamos la operación .file
 - b) Utilizamos la operación .exec
 - c) Utilizamos la operación .read
 - d) Ninguna de las anteriores
21. En Git la clave "SHA-1" está asociada a:
- a) Cada copia del repositorio
 - b) Cada usuario que puede hacer commits en el proyecto
 - c) Un conjunto de commits
 - d) Cada commit
22. La biblioteca Gtk+
- a) Implementa su tecnología de señales/manejadores.
 - b) Aprovecha la tecnología de señales/manejadores de GLib/Object
 - c) No usa el concepto de señales/manejadores
 - d) Ninguna de las anteriores

//PREGUNTAS VARIADAS DE OTROS AÑOS QUE NO SON DE VALA

5. La operacion commit de Git:

A. Traslada los datos del repositorio local a la copia maestra del mismo.

B. Traslada los cambios hechos en la copia de trabajo al repositorio local.

C. Traslada los cambios hechos en la copia de trabajo a un repositorio

6. La operacion branch de Git:

A. Es la unica que podemos usar para crear ramas.

B. Es la unica que podemos usar para cambiar de rama.

C. Es la unica que podemos usar para renombrar una rama.

D. Ninguna de las anteriores.

2. Para guardar una copia temporal de tu directorio de trabajo:

- (a) Utilizamos la operación `Git bisect`.
- (b) Utilizamos la operación `Git clone`.
- (c) Utilizamos la operación `Git stash`.
- (d) No se puede realizar una copia, el directorio de trabajo es único.

B) GIT CLONE

4. Git guarda toda la meta-información de un proyecto en:

- (a) Una base de datos relacional.
- (b) En un directorio único para cada usuario.
- (c) En un directorio único por proyecto.
- (d) En un archivo de configuración.

C) EN UN DIRECTORIO UNICO POR PROYECTO

7. Sqlite:

- (a) requiere de un proceso servidor
- (b) requiere de un servidor y de un cliente.
- (c) en la versión 3 (Sqlite3) sí requiere de un proceso servidor
- (d) Ninguna de las anteriores.

9. Trabajando con Git si dos programadores modifican el mismo archivo:

- (a) no tiene por qué producirse un conflicto.
- (b) siempre se producirá un conflicto.
- (c) Git no permite modificar simultáneamente un mismo archivo.
- (d) ninguna de las anteriores.

- **Conflicto**: Situación que surge cuando dos desarrolladores hacen un *commit* con cambios en la *misma región del mismo fichero*. El **scv** lo detecta, pero es el programador el que debe corregirlo.

CREO QUE ES A) NO TIENE PORQUE PRODUCIRSE UN CONFLICTO

13. En un sistema de control de versiones *centralizado* la operación "push":
- (a) No existe.
 - (b) Hace el mismo papel que en uno *distribuido*.
 - (c) Es equivalente a *pull*.
 - (d) Es equivalente a *rebase*.

CREO QUE ES EQUIVALENTE A PULL PERO NO LO SE JEJEJEJE

14. Si en Git quisiéramos deshacer un *commit* sin perder la parte de la historia del proyecto donde aparece este *commit*:
- (a) No podemos.
 - (b) Usaremos la orden `"git revert"`.
 - (c) Usaremos la orden `"git reset"`.
 - (d) Usaremos la orden `"git rewind"`.

B) GIT REVERT (ESTY BASTANTE SEGURO)

18. Para modificar un mensaje de un *commit*:
- (a) Un mensaje de un *commit* no se puede modificar
 - (b) Utilizamos la opción `--amend` para modificar el último *commit*.
 - (c) Debemos hacer un *checkout* del archivo y luego usamos `--amend`.
 - (d) Ninguna de las anteriores.

B)UTILIZAMOS LA OPCION --AMEND PARA MODIFICAR EL ULTIMO COMMIT

19. La operación checkout de Git:

- (a) Sirve para cambiarse de una rama a otra con la opción -b.
- (b) La podemos usar para cambiar de rama.
- (c) Sirve únicamente para actualizar la copia de trabajo con la versión del repositorio local.
- (d) Ninguna de las anteriores.

B) LA PODEMOS USAR PARA CAMBIAR DE RAMA