

- 1.- MSIL es
  - (a) el lenguaje intermedio reconocido por el CLR
  - (b) un protocolo de comunicación
  - (c) una de las capas .NET
  - (d) ninguna de las anteriores
- 2.- En la arquitectura de capas propuesta
  - (a) Las entidades de negocio siempre acceden directamente a las bb.dd
  - (b) Las entidades de negocio solo en algunas ocasiones acceden directamente a las bb.dd
  - (c) Las entidades de negocio no acceden directamente a las bb.dd nunca
  - (d) Ninguna de las anteriores
- 3.- En el acceso conectado, recuperamos una fila de una consulta
  - (a) con el método nextRow de la clase Row
  - (b) con el método anotherRow de la clase DataReader
  - (c) con el método Read de la clase DataReader
  - (d) con el método Read de la clase DataRow
- 4.- Las tareas CRUD son llevadas a cabo por:
  - (a) la interfaz de usuario
  - (b) la lógica de negocio
  - (c) el componente de acceso a datos
  - (d) ninguna de las anteriores
- 5.- C# es un lenguaje
  - (a) débilmente tipado
  - (b) fuertemente tipado
  - (c) no tiene tipos
  - (d) ninguna de las anteriores
- 6.- en ASP.net...
  - (a) una aplicación Web está formada por un conjunto de páginas web
  - (b) Una aplicación web sólo existe en una localización que ha sido publicada por IIS como un Directorio Virtual
  - (c) Todo lo que requiera la aplicación web debe localizarse en el cliente
  - (d) Todas son ciertas
- 7.- En C# el enlace entre un evento y su controlador
  - (a) se implementa mediante un delegado
  - (b) se realiza automáticamente
  - (c) se implementa mediante un puntero a función
  - (d) a y c son ciertas
- 8.- En el acceso desconectado la clase que está entre nuestra aplicación y la conexión de las bb.dd se llama:
  - (a) DataReader
  - (b) DataSet
  - (c) DataAdapter
  - (d) Command
- 9.- En el acceso conectado, la cadena de conexión se pasa como parámetro
  - (a) al constructor de la clase SqlCommand
  - (b) al constructor de la clase SqlDataReader
  - (c) al constructor de la clase ExecuteReader
  - (d) al constructor de la clase SqlConnection
- 10.- En el objeto Application
  - (a) mantengo el contador de visitas del sitio web
  - (b) mantengo el nombre de usuario
  - (c) las dos anteriores son verdaderas
  - (d) ninguna de las anteriores
- 11.- En .NET cada lenguaje de programación define sus propios tipos de datos
  - (a) cierto
  - (b) falso
  - (c) solo coinciden los tipos de datos de C# y F#
  - (d) ninguna de las anteriores
- 12.- La propiedad que indica si un control de validación es correcto se llama
  - (a) causesValidation
  - (b) isOK
  - (c) isValid
  - (d) isCorrect

13.- Los objetos de la clase DataColumn

- (a) Contienen datos de nuestra aplicación
- (b) Esa clase no existe
- (c) **Contienen información sobre una columna**
- (d) Ninguna de las anteriores

14.- La información guardada en la sesión

- (a) perdura entre diferentes sesiones del mismo usuario
- (b) **perdura solo durante la sesión actual del usuario**
- (c) perdura hasta que el usuario la borre
- (d) ninguna de las anteriores

15.- La concurrencia en el método desconectado de ADO.net

- (a) se gestiona bloqueando las filas hasta que el usuario las libere
- (b) se gestiona con la estrategia Lastwin
- (c) debemos escribir el código necesario para gestionarla
- (d) **se guarda una copia del dataset original y se compara con la BBDD para ver si ha sido modificada**

16.- En el objeto Session

- (a) mantengo el contador de visitas del sitio web
- (b) **mantengo el nombre del usuario**
- (c) las dos anteriores son verdaderas
- (d) ninguna de las anteriores

17.- Un directorio virtual

- (a) debe ser implementado en la clase derivada
- (b) debe heredar de una interfaz
- (c) debe ser implementado y sobrescrito en la clase derivada
- (d) **se identifica por un alias que representa la localización física en el servidor**

18.- Las páginas maestras

- (a) **me permiten crear una plantilla común para todas las páginas de la aplicación web**
- (b) pueden ejecutarse sobre cualquier navegador html
- (c) deben contener controles de usuario
- (d) ninguna de las anteriores

19.- El término code behind

- (a) hace referencia a código asp
- (b) **hace referencia a código C#**
- (c) hace referencia a código asp y C#
- (d) ninguna de las anteriores

20.- En C# el tipo ArrayList

- (a) permite almacenar elementos de tipo string
- (b) **todos los elementos almacenados en un Arraylist son tratados como objetos**
- (c) todos los elementos deben ser del mismo tipo
- (d) ninguna de las anteriores

21.- En asp

- (a) no es posible trabajar con cookies
- (b) podemos crear una cookie como instancia de la clase Cookie
- (c) **podemos crear una cookie como instancia de la clase HttpCookie**
- (d) podemos crear una cookie como instancia de la clase AspNetCookie

22.- En asp la clase que representa un Email se llama

- (a) Email
- (b) SmtClient
- (c) **MailMessage**
- (d) Mail

23.- El objeto Response

- (a) **lo utiliza el servidor para enviar información al cliente**
- (b) lo utiliza el cliente para enviar información del servidor
- (c) Este objeto se utiliza para leer las cookies creadas en el cliente
- (d) Ninguna de las anteriores

24.- El atributo runat = "server"

- (a) se declara en los controles que requieren procesamiento en el cliente
- (b) **se declara en los controles que requieren procesamiento en el servidor**
- (c) tiene asociado una hoja de estilo CSS
- (d) optimiza el SEO del sitio web

25.- El objeto Session

- (a) mantiene las cookies del cliente
- (b) mantiene las preferencias del navegador
- (c) **hace que el navegador recuerde los datos de usuario**
- (d) ninguna de las anteriores

# Herramientas Avanzadas para el Desarrollo de Aplicaciones

## Parte II

### 1. En C#:

- = Es necesario inicializar las variables
- = Existe un recolector de basura
- = El bloque finally se ejecuta siempre
- = Con la conversión explícita perdemos datos

### 2. En el modelo de capas:

- = La Interfaz llama al EN

### 3. Donde se guarda la aplicación WEB:

- = En un Directorio Virtual

### 4. RAD son las siglas de:

- = Rápido Desarrollo de Aplicaciones

### 5. Es obligatorio establecer la etiqueta runat = "server" en ASP.NET:

- = Si

### 6. Al establecer la etiqueta runat = "server":

- = El formulario se debe procesar en el servidor

### 7. Un proyecto WEB:

- = Sirve para aplicaciones web avanzadas

### 8. Las paginas maestras:

- = Proporcionan coherencia a la pagina

**9. En las master pages (paginas maestras):**

= Definimos el contenido común y los contenedores de contenido

**10. El enlace entre el mensaje del evento y el mensaje específico se lleva a cabo mediante:**

= Delegado de eventos

**11. Los delegados son:**

= Puntero a función

**12. Un Boton es de tipo:**

= Evento de envío

**13. Un Textbox es de tipo:**

= Ambas son Correctas (Evento de cache y no Postback)

**14. Se puede transformar No Postback a Postback:**

= Asignando AutoPostBack = true

**15. Se puede generar estilo de web sin usar archivo CSS:**

= Si, con código inline

**16. Con el comando Response.Redirect():**

= Podemos enviar parámetros dentro de la web

**17. Para leer parámetros en la Web usamos:**

= Request

**18. Un menú estático es aquel que:**

= El control del menú esta expandido completamente

**19. Un menú dinámico es aquel que:**

= Son estáticas las porciones especificadas

**20. Con RequiredFieldValidator Validamos:**

= La cadena no sea vacía

**21. Los controladores de validación pueden ser:**

= Los dos (Estático y dinámico)

**22. En los controladores de validación (marcar la falsa):**

= No se puede controlar un rango de datos

**23. La validación de un correo electrónico es:**

= \S+@\S+\.\S+

**24. En el objeto sesión:**

= Ambas son correctas

(Al acabar timeout se cierra sesión y al cerrar el navegador se borran los datos)

**25. Para cerrar sesión usamos:**

= Session.Abandon

**26. Cuales están implementados como colecciones o conjuntos de pares nombre-valor:**

= Ambas (Session y Application)

**27. Si queremos asignar variables para todos los usuarios usaremos:**

= Application

**28. Las variables application pueden ser bloqueadas:**

= Si

**29. El archivo Global.asax:**

= No contiene etiquetas ASP.NET

**30. Los cambios en Global.asax:**

= Requieren reinicio de aplicación

**31. En el entorno conectado usaremos:**

= Datareader

**32. En el entorno desconectado usaremos:**

= Dataset

**33. Los Objetos Connection y Command:**

= Tienen prefijo

**34. El archivo web.config:**

= Basado en XML

**35. Para realizar un SELECT:**

= ExecuteReader

**36. Con DataAdapter:**

= La gestión por abrir y cerrar la BBDD es automática

**37. Con DataAdapter:**

= La gestión por insertar, borrar y actualizar la BBDD es automática utilizando  
CommandBuilder

**38. GridView:**

= Los dos (Con asistente y con código)



**48. MSIL es:**

El lenguaje intermedio reconocido por el CLR

**49. En la arquitectura de capas propuesta:**

Las entidades de negocio no acceden directamente a las bb.dd nunca

**50. En el acceso conectado, recuperamos una fila de una consulta:**

Con el método Read de la clase DataReader

**51. Las tareas CRUD son llevadas a cabo por:**

El comportamiento de acceso a datos

**52. C# es un lenguaje:**

Fuertemente tipado

**53. En ASP.net:**

Una aplicación web sólo existe en una localización que ha sido publicada por IIS como un Directorio Virtual

**54. En C# el enlace entre un evento y su controlador:**

A y C son ciertas

(Implementa mediante un delegado y se implementa mediante un puntero a función)

**55. En el acceso desconectado de la clase que está entre nuestra aplicación y la conexión de las bb.dd se llama:**

DataAdapter

**56. En el acceso conectado, la cadena de conexión se pasa como parámetro:**

Al constructor de la clase SqlConnection

**57. En el objeto Application:**

Mantengo el contador de visitas del sitio web



**58. En .NET cada lenguaje de programación define sus propios tipos de datos**

Cierto

**59. La propiedad que indica si un control de validación es correcto se llama:**

IsValid

**60. Los objetos de la clase DataColumn:**

Contiene información sobre una columna

**61. La información guardada en la sesión:**

Perdura solo durante la sesión actual del usuario

**62. La concurrencia en el método desconectado de ADO.net:**

Se guarda una copia del dataset original y se compara con la BBDD para ver si ha sido modificada

**63. En el objeto Session:**

Mantengo el nombre de usuario

**64. Un directorio virtual:**

Se identifica por un alias que representa la localización física en el servidor

**65. Las páginas maestras:**

Me permiten crear una plantilla común para todas las páginas de la aplicación web

**66. El término code behind:**

Hace referencia a código C#

**67. En C# el tipo ArrayList:**

Todos los elementos almacenados en un ArrayList son tratados como objetos

**68. En asp:**

Podemos crear una cookie como instancia de la clase HttpCookie

**69. En asp la clase que representa un Email se llama:**

MailMessage

**70. El objeto Response:**

Lo utiliza el servidor para enviar información al cliente

**71. El atributo runat ="server":**

Se declara en los controles que requieren procesamiento en el servidor

**72. El objeto Session:**

Ninguna de las anteriores

(No mantiene las cookies, no mantiene las preferencias del navegador y no hace que el navegador recuerde los datos del usuario)

# Herramientas Avanzadas para el Desarrollo de Aplicaciones

## Parte I

**1. La autoconexión de señales en el marco Glade + Gtk+ se hace:**

Sobre un objeto de la clase Builde.

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

Utilizamos la operación Git stash.

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

En un directorio único por proyecto.

**4. SQLite:**

Ninguna de las anteriores.

(Ni requiere de un proceso servidor, ni requiere de un servidor y de un cliente, y ni en la versión 3 sí requiere un proceso servidor)

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

No tiene por qué producirse un conflicto.

**6. En Git la clave “SHA-1” está asociada a:**

Cada commit.

**7. En un sistema de control de versiones centralizado la operación “push”:**

No existe.

**8. Si en Git quisiéramos deshacer un commit sin perder la parte de la historia del proyecto donde aparece este commit:**

Usaremos la orden `git revert`.

**9. La biblioteca Gtk+...**

Aprovecha la tecnología de señales/manejadores de Glib/Gobject.

**10. En Git la operación pull equivale a:**

`fetch + merge`.

**11. Para modificar un mensaje de un commit:**

Utilizamos la opción `--amend` para modificar el último commit.

**12. La operación checkout de Git:**

La podemos utilizar para cambiar de rama.

**13. La signatura de un manejador conectado a una señal:**

Está condicionada por la signatura de la señal.

**14. En el patrón de la arquitectura MVC:**

Un modelo puede tener varias vistas sobre él.

**15. En la programación dirigida por eventos, la cola de eventos:**

No es necesaria.

**16. La operación commit de Git:**

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

**17. A una señal con visibilidad pública:**

Le podemos conectar cualquier manejador independientemente de su visibilidad.

**18. Los SVC, según la forma de almacenar los contenidos, se clasifican en:**

Centralizados y distribuidos.

**19. Las librerías dinámicas:**

Ninguna de las anteriores.

(Hay que compilarlas, no debe llevar el prefijo lib y la extensión y no hace crecer el ejecutable del final)

**20. Git es un sistema de control de versiones:**

Distribuido.

**21. Imagen:**

## Gtk+ ejemplo sencillo I

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

La inicialización de gtk+ es Application.Init();

**22. Imagen:**

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

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

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

**23. Imagen:**

```
// 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 ();
}
```

Gtk + termina con Application.Quit;

**24. Sqlite es para:**

Mas. (C, C# y muchas más)

**25. Compara MonoDevelop y Visual Studio:**

Pueden ejecutar el mismo tipo de proyectos.

**26. Para consultar los metadatos de una bb.dd. en sqlite:**

Consultamos la tabla sqlite\_master.

**27. En el interfaz de comandos sqlite para ejecutar en un fichero con comandos sql:**

Utilizamos la operación .exec

**28. La operacion branch de Git:**

Es la única que podemos usar para renombrar una rama.

//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

# Herramientas Avanzadas para el Desarrollo de Aplicaciones

Parte Escritorio marzo 2015

1. La autoconexión de señales en el marco Glade + Gtk+ se hace:
  - a) sobre un objeto de la clase Window
  - b) en un método estático de la clase Window
  - c) sobre un objeto de la clase Builder
  - d) en un método estático de la clase Builder
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
3. En Vala una función- $\lambda$  puede hacer de manejador...
  - a) Nunca
  - b) Siempre
  - c) Cuando coincida su lista de parámetros con los de la señal
  - d) Cuando no tenga parámetros
4. Git guarda toda la meta-información de un proyecto en:
  - a) Una base de datos relacional
  - b) Un directorio único para cada usuario
  - c) En un directorio único por proyecto
  - d) En un archivo de configuración
5. En Vala el operador `as`:
  - a) Es un operador matemático
  - b) Es un operador de conversión de tipos en tiempo de compilación
  - c) Es un operador de conversión de tipos en tiempo de ejecución
  - d) Ninguna de las anteriores
6. En el lenguaje Vala colocar el símbolo “(?)” tras el nombre de un tipo:
  - a) Es un error
  - b) No significa nada
  - c) Indica que la variable que se declare de ese tipo puede contener el valor NULL
  - d) Indica que la variable que se declare de ese tipo sólo podrá contener el valor NULL

7. SQLite:
- a) requiere de un proceso servidor
  - b) requiere de un servidor y de un cliente
  - c) en la versión 3 (SQLite 3) sí requiere un proceso servidor
  - d) Ninguna de las anteriores
8. En el patrón de la arquitectura MVC aplicado en Vala usando Gtk+:
- a) El modelo es representado por una clase y el controlador sería implementado por la librería Gtk+
  - b) El modelo es representado por una clase y el controlador por una clase interfaz
  - c) La vista se implementa siempre con Glade en formato XML
  - d) B y C son ciertas
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 en un mismo archivo
  - d) Ninguna de las anteriores
10. En Vala los espacios de nombres sólo se pueden crear así
- a) ¿Espacios de nombres...? ¿Qué es eso?
  - b) `using namespace name;`
  - c) `namespace name {...}`
  - d) Ninguna de las anteriores
- (Puedes crear un espacio de nombres con `class Hada.Ejemplo {...}`)
11. En Vala la clase `ArrayList<T>` forma parte...
- a) Del lenguaje, es un tipo de datos básico
  - b) De nada. Esa clase no existe
  - c) De una biblioteca externa
  - d) Ninguna de las anteriores
12. 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
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`



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`
15. La biblioteca Gtk+...
- a) Implementa su tecnología de señales/manejadores
  - b) Aprovecha la tecnología de señales/manejadores de Glib/Gobject
  - c) No usa el concepto de señales/manejadores
  - d) Ninguna de las anteriores
16. Para poder usar la biblioteca Gtk+ desde Vala...
- a) Debemos llamar al compilador con la opción "`--pkg gtk+-2.0`"
  - b) No se necesita ninguna operación especial de compilación
  - c) Debemos llamar al compilador con la opción "`--pkg gmodule-2.0`"
  - d) Es necesario construir previamente un interfaz gráfico de la aplicación con Glade
17. En Git la operación `pull` equivale a:
- a) `fetch + merge`
  - b) `merge`
  - c) `push`
  - d) `rebase`
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 de hacer un checkout del archivo y luego usamos `-amend`
  - d) Ninguna de las anteriores
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
20. La signatura de un manejador conectado a una señal:
- a) Está condicionada por la signatura de la señal
  - b) No está delimitada por la signatura de la señal
  - c) Depende de si la clase a la cual pertenece está dentro de un espacio de nombres
  - d) Ninguna de las anteriores


21. En el patrón de la arquitectura MVC:

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

22. 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

23. Cuando compilamos en Vala una aplicación que hace uso de Gtk y glade:

- a) Debemos añadir la opción del compilador `-with-gtk`
- b) Debemos añadir la opción del compilador `-with-glade`
- c) Debemos añadir la opción del compilador `-pkg gtk-2.0`
- d) Debemos añadir la opción del compilador `-pkg gmodule2.0` 

24. La operación 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

25. A una señal con visibilidad pública:

- a) Sólo le podemos conectar manejadores públicos
- b) Sólo le podemos funciones independientes de cualquier clase
- c) Le podemos conectar cualquier manejador independientemente de su visibilidad
- d) Ninguna de las anteriores

Quien programa mejor:

```
{  
~ Lexus  
= GT++  
~ Jose el del Tupe  
}
```

En C#:

```
{  
~ Es correcta la expresion float a = 3,5;  
= Es necesario inicializar las variables  
~ La conversion implicita tiene perdida de informacion  
}
```

En C#:

```
{  
= Existe un recolector de basura  
~ Debemos destruir las variables manualmente  
~ Microsoft dice que esta basado en Java  
}
```

En C#:

```
{  
~ No se pueden manejar excepciones  
~ Se puede hacer un try sin catch  
= El bloque finally se ejecuta siempre  
}
```

En C#:

```
{  
~ No se puede hacer casting  
~ Con la conversion implicita perdemos datos  
= Con la conversion explicita perdemos datos  
}
```

En el modelo de capas:

```
{  
= La Interfaz llama al EN  
~ El CAD llama al EN  
~ El EN llama a la Interfaz  
}
```

Donde se guarda la aplicacion WEB:

```
{  
~ En una Carpeta  
= En un Directorio Virtual  
~ En LocalHost  
}
```

RAD son las siglas de:

```
{
```

- = Rapido Desarrollo de Aplicaciones
- ~ Rapido Analisis y Desarrollo
- ~ Rapido Dise ⊕ de Aplicaciones

}

Es obligatorio establecer la etiqueta runat = "server" en ASP.NET:

- {
- ~ NO
- = SI
- ~ Depende de la etiqueta

}

Al establecer la etiqueta runat = "server":

- {
- ~ El formulario no debe ser procesado en el servidor
- ~ El formulario lo envia el servidor
- = El formulario se debe procesar en el servidor

}

Un proyecto WEB:

- {
- = Sirve para aplicaciones web avanzadas
- ~ Sirve para paginas web sencillas
- ~ No podemos referenciar DLLs

}

Las paginas maestras:

- {
- ~ Solo puede haber 1 pagina maestra
- ~ Proporcionan seguridad a la pagina
- = Proporcionan coherencia a la pagina

}

En las master pages (paginas maestras):

- {
- = Definimos el contenido comun y los contenedores de contenido
- ~ Hacemos referencia a otras paginas maestras
- ~ Creamos el contenido de los contenedores

}

El enlace entre el mensaje del evento y el mensaje especifico se lleva a cabo mediante:

- {
- ~ Emisor de eventos
- = Delegado de eventos
- ~ Receptor de eventos

}

Los delegados son:

- {
- = Puntero a funcion
- ~ Puntero a mensaje

```
~ Puntero a destino  
}
```

Un Boton es de tipo:

```
{  
~ No Postback  
~ Evento de cache  
= Evento de envio  
}
```

Un Textbox es de tipo:

```
{  
~ Evento de cache  
~ No Postback  
= Ambas son Correctas  
}
```

Se puede transformar No Postback a Postback:

```
{  
~ Los eventos No Postback ya son Postback  
= Asignando AutoPostBack = true  
~ Asignando PostBack = true  
}
```

Se puede generar estilo de web sin usar archivo CSS:

```
{  
= Si, con codigo inline  
~ No, tiene que existir este archivo  
~ Ninguna de las anteriores  
}
```

Con el comando Response.Redirect():

```
{  
= Podemos enviar parametros dentro de la web  
~ Solo podemos acceder a paginas en el directorio raiz  
~ Ninguna de las anteriores  
}
```

Para leer parametros en la Web usamos:

```
{  
~ Commander  
= Request  
~ Application  
}
```

Un menu estatico es aquel que:

```
{  
~ Son estaticas las porciones especificadas  
~ El control del menu es variable  
= El control del menu esta expandido completamente  
}
```

Un menu dinamico es aquel que:

```
{
= Son estaticas las porciones especificadas
~ Toda la estructura es siempre visible
~ El control del menu esta expandido completamente
}
```

Con RequiredFieldValidator Validamos:

```
{
~ La cadena sea numerica
= La cadena no sea vacia
~ La cadena sea alfabetica
}
```

Los controladores de validacion pueden ser:

```
{
~ Estaticos
~ Dinamicos
= Los dos
}
```

En los controladores de validacion (marcar la falsa):

```
{
= No se puede controlar un rango de datos
~ Se puede controlar una expresion
~ Podemos definir nuestro propio control
}
```

La validacion de un correo electronico es:

```
{
~ \w{1,3}\.\w{1,3}@(\w{1,3}\.){1,3}\w{1,3}
= \w{1,3}\.\w{1,3}@(\w{1,3}\.){1,3}\w{1,3}
~ \w{1,3}\.\w{1,3}@(\w{1,3}\.){1,3}\w{1,3}
}
```

En el objeto sesion:

```
{
~ Al acabar el timeout se cierra sesion
~ Al cerrar el navegador se borran los datos
= Ambas son correctas
}
```

Para cerrar sesion usamos:

```
{
= Session.Abandon
~ Session.Remove
~ Session.Close
}
```

Cuales estan implementados como colecciones o conjuntos de pares nombre-valor:

```
{
~ Session
~ Application
= Ambas
}
```

Si queremos asignar variables para todos los usuarios usaremos:

```
{
~ Session
= Application
~ Ambas
}
```

Las variables application pueden ser bloqueadas:

```
{
= Si
~ No
~ Depende del criterio a seguir
}
```

El archivo Global.asax:

```
{
~ Contiene etiquetas HTML
~ Contiene etiquetas XML
= No contiene etiquetas ASP.NET
}
```

Los cambios en Global.asax:

```
{
= Requieren reinicio de aplicacion
~ Requieren compilacion de aplicacion
~ Se ejecutan con normalidad
}
```

En el entorno conectado usaremos:

```
{
~ Dataset
= Datareader
~ Dataconnect
}
```

En el entorno desconectado usaremos:

```
{
= Dataset
~ Datareader
~ Dataconnect
}
```

Los Objetos Connection y Command:

```
{
= Tienen prefijo
```

- ~ No tienen prefijo
- ~ Tienen prefijo, como dataset

```
}
```

El archivo web.config:

```
{  
~ Basado en ASP.NET  
~ Basado en HTML  
= Basado en XML  
}
```

Para realizar un SELECT:

```
{  
~ ExecuteNonQuery  
~ ExecuteCommand  
= ExecuteReader  
}
```

Con DataAdapter:

```
{  
~ Debemos gestionar en todo momento la apertura y cierre de la BBDD  
= La gestion por abrir y cerrar la BBDD es automatica  
~ La gestion por abrir y cerrar la BBDD es automatica utilizando CommandBuilder  
}
```

Con DataAdapter:

```
{  
~ Debemos gestionar en todo momento la apertura y cierre de la BBDD  
~ La gestion por insertar, borrar y actualizar la BBDD es automatica  
= La gestion por insertar, borrar y actualizar la BBDD es automatica utilizando  
CommandBuilder  
}
```

GridView:

```
{  
~ Con asistente  
~ Con codigo  
= Los dos  
}
```

La concurrencia pesimista es:

```
{  
= Cuando una fila es leida, esta queda bloqueada para su lectura para cualquier otro  
que la demande hasta que aquel que la posee la libere.  
~ Las filas estan disponibles para su lectura en todo momento, estas pueden ser leidas  
por distintos usuarios al mismo tiempo.  
~ Esta tecnica implica que no existe control. El ultimo cambio en escribirse es el que  
permanece.  
}
```

La concurrencia last win es:



```

{
~ Cuando una fila es leida, esta queda bloqueada para su lectura para cualquier otro
que la demande hasta que aquel que la posee la libere.
~ Las filas estan disponibles para su lectura en todo momento, estas pueden ser leidas
por distintos usuarios al mismo tiempo.
= Esta tecnica implica que no existe control. El ultimo cambio en escribirse es el que
permanece.
}

```

La concurrencia positiva es:

```

{
~ Cuando una fila es leida, esta queda bloqueada para su lectura para cualquier otro
que la demande hasta que aquel que la posee la libere.
= Las filas estan disponibles para su lectura en todo momento, estas pueden ser leidas
por distintos usuarios al mismo tiempo.
~ Esta tecnica implica que no existe control. El ultimo cambio en escribirse es el que
permanece.
}

```

La concurrencia optimista es:

```

{
= Una cota de ADA que nos raya en este tipo de preguntas
~ Las filas estan disponibles para su lectura en todo momento, estas pueden ser leidas
por distintos usuarios al mismo tiempo.
~ Esta tecnica implica que no existe control. El ultimo cambio en escribirse es el que
permanece.
}

```

Cuando tenemos que hacer un acceso complicado usaremos:

```

{
~ DataReader
~ DataAdapter
= DataSet
}

```

Si trabajamos con mas de una BBDD usaremos:

```

{
= DataSet
~ DataReader
~ DataAdapter
}

```

En las Cookies:

```

{
~ Los datos se borran siempre cuando el usuario cierra la ventana del navegador
= Las cookies no se pierden cuando se cierra el navegador (a no ser que el usuario las
borre)
~ Una cookie se representa por la clase Cookie
}

```

En las Cookies:

```
{
~ Los datos se borran siempre cuando el usuario cierra la ventana del navegador
~ Las cookies no se pueden borrar con fechas ya expiradas
= Una cookie se representa por la clase HttpCookie
}
```

La extension de los controles de usuario es:

```
{
~ aspx
= ascx
~ asdx
}
```

- Las tareas CRUD son llevadas a cabo por:
  - a) la interfaz de usuario
  - b) la logica de negocio
  - c) el componente de acceso a datos
  - d) ninguna de las anteriores
- C# es un lenguaje
  - a) debilmente tipado
  - b) fuertemente tipado
  - c) no tiene tipos
  - d) ninguna de las anteriores
- La paginas maestras
  - a) me permiten crear una plantilla comun para todas las paginas de la aplicacion web
  - b) pueden ejecutarse sobre cualquier navegador html
  - c) deben contener controles de usuario
  - d) ninguna de las anteriores
- En C# el tipo ArrayList
  - a) permite almacenar elementos del tipo string
  - b) todos los elementos almacenados en un ArrayList son tratados como objetos
  - c) todo los elementos deben ser del mismo tipo
  - d) ninguna de las anteriores
- El objeto Response
  - a) lo utiliza el servidor para enviar informacion al cliente
  - b) lo utiliza el cliente para enviar informacion del servidor
  - c) Este objeto se utiliza para leer las cookies creadas en el cliente
  - d) ninguna de las anteriores
- En el acceso conectado, recuperamos una fila de una consulta
  - a) con el metodo nextRow de la clase Row
  - b) con el metodo anotherRow de la clase DataReader
  - c) con el metodo Read de la clase DataReader
  - d) con el metodo Read de la clase DataRow
- La informacion guardada en la sesion
  - a) perdura entre diferentes sesiones del mismo usuario
  - b) perdura solo durante la sesion actual del usuario
  - c) perdura hasta que el usuario la borre
  - d) ninguna de las anteriores
- La propiedad que indica di un control de validacion es correcto se llama
  - a) causesValidation
  - b) isOk
  - c) isValid
  - d) isCorrect
- Los objetos de clase DataColumn
  - a) Contienen datos de nuestra aplicacion
  - b) Esa clase no existe
  - c) Contienen informacion sobre una columna
  - d) Ninguna de las anteriores
- La concurrencia en el metodo desconectado de ADO.net
  - a) se gestiona bloqueando las filas hasta que el usuario las libere
  - b) se gestiona con la estrategia Lastwin
  - c) debemos escribir el codigo necesario para gestionarla
  - d) se guarda una copia del dataset original y se compara con la BBDD para ver si ha sido modificada
- En la arquitectura de capa propuesta
  - a) Las entidades de negocio siempre acceden direcctamente a las bb.dd
  - b) Las entidades de negocio solo en algunas ocasiones acceden direcctamente a las bb.dd
  - c) no acceden directamente a las bb.dd nunca
  - d) ninguna de las anteriores
- MSIL es
  - a) el lenguaje intermedio reconocido por el CLR
  - b) un protocolo de comunicacion
  - c) una de las capas de .NET
  - d) ninguna de las anteriores
- Un directorio virtual
  - a) debe ser implementado en la clase derivada
  - b) debe heredar de una intefaz
  - c) debe ser implementado y sobrescrito en la clase derivada
  - d) se identifica por un alias que representa la localizacion fisica en el servidor

- En asp la clase que representa un Email de llama
  - a) Email
  - b) SmtpClient
  - c) MailMessage
  - d) Mail
- En el acceso conectado, la cadena de conexion se pasa como parametro
  - a) al constructor de la clase SqlCommand
  - b) al constructor de la clase SqlDataReader
  - c) al constructor de la clase ExecuteReader
  - d) al constructor de la clase SqlConnection
- En .NET cada lenguaje de programacion define sus propios tipos de datos
  - a) cierto
  - b) falso
  - c) solo coinciden los tipos de datos de C# y F#
  - d) ninguna de las anteriores
- El termino "code behind"
  - a) hace referencia a codigo asp
  - b) hace referencia a codigo C#
  - c) hace referencia a codigo asp y C#
  - d) ninguna de las anteriores
- En el objeto Session
  - a) mantengo el contador de visitas del sitio web
  - b) mantengo el nombre de usuario
  - c) las dos anteriores son verdaderas
  - d) ninguna de las anteriores
- En el objeto Application
  - a) mantengo el contador de visitas del sitio web
  - b) mantengo el nombre de usuario
  - c) las dos anteriores son verdaderas
  - d) ninguna de las anteriores
- En el acceso desconectado la clase que esta entre nuestra aplicacion y la conexion con la bb.dd se llama:
  - a) DataReader
  - b) DataSet
  - c) DataAdapter
  - d) Command
- En asp
  - a) no es posible trabajar con cookies
  - b) podemos crear una cookie como instancia de la clase Cookie
  - c) podemos crear una cookie como instancia de la clase Http-Cookie
  - d) podemos crear una cookie como instancia de la clase AspNetCookie
- El atributo 'runat="server"'
  - a) se declara en los controles que requieren procesamiento en el cliente
  - b) se declara en los controles que requieren procesamiento en el servidor
  - c) tiene asociado una hoja de estilos CSS
  - d) optimiza el SEO del sitio web
- En C# el enlace entre un evento y su controlador
  - a) se implementa mediante un delegado
  - b) se realiza automaticamente
  - c) se implementa mediante un puntero a funcion
  - d) a y c son ciertas
- En el objeto Session
  - a) mantiene las cookies del cliente
  - b) mantiene las preferencias del navegador
  - c) hace que el navegador recuerde los datos del usuario
  - d) ninguna de las anteriores
- En ASP.net...
  - a) Una aplicacion Web esta formada por un conjunto de paginas web
  - b) Una aplicacion web solo existe en una localizacion que ha sido publicada por IIS como un Directorio Virtual
  - c) Todo lo que requiera la aplicacion web debe localizarse en el cliente
  - d) todas son ciertas

# Herramientas Avanzadas para el Desarrollo de Aplicaciones

marzo 2012

- El examen tiene una duración de 30 minutos.
- Todas las preguntas tienen una (y sólo) una respuesta válida.
- **Dos** respuestas erróneas anulan una buena.
- Es preferible que utilices lápiz en lugar de bolígrafo.
- Debes entregar esta hoja de enunciado al profesor una vez hayas terminado.
- **Modalidad: A**

Nombre: \_\_\_\_\_

1. En Vala los espacios de nombres sólo se pueden crear así:
  - A. ¿Espacios de nombres...? ¿Qué es eso?.
  - B. `using namespace name;`
  - C. `namespace name {...}`.
  - D. **Ninguna de las anteriores.**
2. En Vala la clase `ArrayList<T>` forma parte ...
  - A. Del lenguaje, es un tipo de datos básico.
  - B. De nada. Esa clase no existe.
  - C. **De una biblioteca externa.**
  - D. Ninguna de las anteriores.
3. En Vala la cláusula `requires` representa:
  - A. Una excepción.
  - B. Una postcondición.
  - C. **Una precondición.**
  - D. Ninguna de las anteriores.
4. 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.
5. La operación `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.
6. La operación `branch` de Git:
  - A. Es la única que podemos usar para crear ramas.
  - B. Es la única que podemos usar para cambiar de rama.
  - C. **Es la única que podemos usar para renombrar una rama.**
  - D. Ninguna de las anteriores.
7. Una **señal**:
  - A. Sólo puede tener conectado un manejador.
  - B. Sólo puede estar conectada con métodos de una clase.
  - C. **Puede tener conectados varios manejadores.**
  - D. Ninguna de las anteriores.
8. La signature de un manejador conectado a una **señal**:
  - A. **Está delimitada por la signature de la señal.**
  - B. No está delimitada por la signature de la señal.
  - C. Depende de si la clase a la cual pertenece está dentro de un espacio de nombres.
  - D. Ninguna de las anteriores.
9. A una **señal** con visibilidad pública:
  - A. Sólo le podemos conectar manejadores públicos.
  - B. Sólo le podemos conectar funciones independientes de cualquier clase.
  - C. **Le podemos conectar cualquier manejador independientemente de su visibilidad.**
  - D. Ninguna de las anteriores.
10. En Vala para que funcione el mecanismo de **señales/manejadores**:
  - A. No es necesario hacer nada, el lenguaje lo proporciona.
  - B. **Debemos derivar de la clase `GLib.Object`.**
  - C. Se debe compilar el código con una opción especial.
  - D. Debemos derivar de la clase `Gtk.Object`.
11. En Vala una función- $\lambda$  puede hacer de manejador...
  - A. Nunca.
  - B. Siempre.
  - C. **Cuando coincida su lista de parámetros con los de la señal.**
  - D. Cuando no tenga parámetros.
12. En Vala una señal...
  - A. Nunca puede tener implementación.
  - B. Siempre puede tener implementación.
  - C. **Sólo podrá tener implementación cuando se declare virtual.**
  - D. Ninguna de las anteriores.
13. La biblioteca `Gtk+`...
  - A. Implementa su tecnología de **señales/manejadores**.
  - B. **Aprovecha la tecnología de señales/manejadores de `GLib/GObject`.**
  - C. No usa el concepto de **señales/manejadores**.
  - D. Ninguna de las anteriores.
14. Para poder usar la biblioteca `Gtk+` desde Vala...
  - A. **Debemos llamar al compilador con la opción `--pkg gtk+-2.0`.**

- B. No se necesita ninguna opción especial de compilación.
  - C. Debemos llamar al compilador con la opción “`--pkg gmodule -2.0`”.
  - D. Es necesario construir previamente el interfaz gráfico de la aplicación con `glade`.
15. Los interfaces de usuario generados por `glade`...
- A. Se guardan en archivos binarios.
  - B. Se guardan como código fuente Vala.
  - C. Constituyen la única manera de dotar de interfaz gráfico a un aplicación que use `Gtk+`.
  - D. Son archivos de texto en formato XML.**

# Herramientas Avanzadas para el Desarrollo de Aplicaciones

marzo 2012

- El examen tiene una duración de 30 minutos.
- Todas las preguntas tienen una (y sólo) una respuesta válida.
- **Dos** respuestas erróneas anulan una buena.
- Es preferible que utilices lápiz en lugar de bolígrafo.
- Debes entregar esta hoja de enunciado al profesor una vez hayas terminado.
- **Modalidad: B**

Nombre: \_\_\_\_\_

1. Una **señal**:
  - A. Sólo puede tener conectado un manejador.
  - B. Sólo puede estar conectada con métodos de una clase.
  - C. Puede tener conectados varios manejadores.**
  - D. Ninguna de las anteriores.
2. La **signatura** de un manejador conectado a una **señal**:
  - A. Está delimitada por la signatura de la señal.**
  - B. No está delimitada por la signatura de la señal.
  - C. Depende de si la clase a la cual pertenece está dentro de un espacio de nombres.
  - D. Ninguna de las anteriores.
3. A una **señal** con visibilidad **pública**:
  - A. Sólo le podemos conectar manejadores públicos.
  - B. Sólo le podemos conectar funciones independientes de cualquier clase.
  - C. Le podemos conectar cualquier manejador independientemente de su visibilidad.**
  - D. Ninguna de las anteriores.
4. En Vala los espacios de nombres sólo se pueden crear así:
  - A. ¿Espacios de nombres...? ¿Qué es eso?.
  - B. `using namespace name;`
  - C. `namespace name {...}`.
  - D. Ninguna de las anteriores.**
5. En Vala la clase `ArrayList<T>` forma parte ...
  - A. Del lenguaje, es un tipo de datos básico.
  - B. De nada. Esa clase no existe.
  - C. De una biblioteca externa.**
  - D. Ninguna de las anteriores.
6. En Vala la cláusula **requires** representa:
  - A. Una excepción.
  - B. Una postcondición.
  - C. Una precondición.**
  - D. Ninguna de las anteriores.
7. 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.
8. La operación **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.
9. La operación **branch** de Git:
  - A. Es la única que podemos usar para crear ramas.
  - B. Es la única que podemos usar para cambiar de rama.
  - C. Es la única que podemos usar para renombrar una rama.**
  - D. Ninguna de las anteriores.
10. En Vala para que funcione el mecanismo de **señales/manejadores**:
  - A. No es necesario hacer nada, el lenguaje lo proporciona.
  - B. Debemos derivar de la clase `GLib.Object`.**
  - C. Se debe compilar el código con una opción especial.
  - D. Debemos derivar de la clase `Gtk.Object`.
11. En Vala una función- $\lambda$  puede hacer de manejador...
  - A. Nunca.
  - B. Siempre.
  - C. Cuando coincida su lista de parámetros con los de la señal.**
  - D. Cuando no tenga parámetros.
12. En Vala una **señal**...
  - A. Nunca puede tener implementación.
  - B. Siempre puede tener implementación.
  - C. Sólo podrá tener implementación cuando se declare virtual.**
  - D. Ninguna de las anteriores.
13. La biblioteca `Gtk+`...
  - A. Implementa su tecnología de **señales/manejadores**.
  - B. Aprovecha la tecnología de señales/manejadores de `GLib/GObject`.**
  - C. No usa el concepto de **señales/manejadores**.
  - D. Ninguna de las anteriores.
14. Para poder usar la biblioteca `Gtk+` desde Vala...
  - A. Debemos llamar al compilador con la opción `--pkg gtk+-2.0`.**

- B. No se necesita ninguna opción especial de compilación.
  - C. Debemos llamar al compilador con la opción “`--pkg gmodule -2.0`”.
  - D. Es necesario construir previamente el interfaz gráfico de la aplicación con `glade`.
15. Los interfaces de usuario generados por `glade`...
- A. Se guardan en archivos binarios.
  - B. Se guardan como código fuente Vala.
  - C. Constituyen la única manera de dotar de interfaz gráfico a un aplicación que use `Gtk+`.
  - D. Son archivos de texto en formato XML.**



# Herramientas Avanzadas para el Desarrollo de Aplicaciones

marzo 2012

- El examen tiene una duración de 30 minutos.
- Todas las preguntas tienen una (y sólo) una respuesta válida.
- **Dos** respuestas erróneas anulan una buena.
- Es preferible que utilices lápiz en lugar de bolígrafo.
- Debes entregar esta hoja de enunciado al profesor una vez hayas terminado.
- **Modalidad: C**

Nombre: \_\_\_\_\_

1. La biblioteca Gtk+...
  - A. Implementa su tecnología de señales/manejadores.
  - B. Aprovecha la tecnología de señales/manejadores de GLib/GObject.**
  - C. No usa el concepto de señales/manejadores.
  - D. Ninguna de las anteriores.
2. Para poder usar la biblioteca Gtk+ desde Vala...
  - A. Debemos llamar al compilador con la opción "--pkg gtk+-2.0".**
  - B. No se necesita ninguna opción especial de compilación.
  - C. Debemos llamar al compilador con la opción "--pkg gmodule -2.0".
  - D. Es necesario construir previamente el interfaz gráfico de la aplicación con glade.
3. Los interfaces de usuario generados por glade...
  - A. Se guardan en archivos binarios.
  - B. Se guardan como código fuente Vala.
  - C. Constituyen la única manera de dotar de interfaz gráfico a un aplicación que use Gtk+.
  - D. Son archivos de texto en formato XML.**
4. Una **señal**:
  - A. Sólo puede tener conectado un manejador.
  - B. Sólo puede estar conectada con métodos de una clase.
  - C. Puede tener conectados varios manejadores.**
  - D. Ninguna de las anteriores.
5. La signature de un manejador conectado a una **señal**:
  - A. Está delimitada por la signature de la señal.**
  - B. No está delimitada por la signature de la señal.
  - C. Depende de si la clase a la cual pertenece está dentro de un espacio de nombres.
  - D. Ninguna de las anteriores.
6. A una **señal** con visibilidad pública:
  - A. Sólo le podemos conectar manejadores públicos.
  - B. Sólo le podemos conectar funciones independientes de cualquier clase.
  - C. Le podemos conectar cualquier manejador independientemente de su visibilidad.**
  - D. Ninguna de las anteriores.
7. En Vala los espacios de nombres sólo se pueden crear así:
  - A. ¿Espacios de nombres...? ¿Qué es eso?.
  - B. `using namespace name;`
  - C. `namespace name {...}`.
  - D. Ninguna de las anteriores.**
8. En Vala la clase `ArrayList<T>` forma parte ...
  - A. Del lenguaje, es un tipo de datos básico.
  - B. De nada. Esa clase no existe.
  - C. De una biblioteca externa.**
  - D. Ninguna de las anteriores.
9. En Vala la cláusula **requires** representa:
  - A. Una excepción.
  - B. Una postcondición.
  - C. Una precondición.**
  - D. Ninguna de las anteriores.
10. 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.
11. La operación **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.
12. La operación **branch** de Git:
  - A. Es la única que podemos usar para crear ramas.
  - B. Es la única que podemos usar para cambiar de rama.
  - C. Es la única que podemos usar para renombrar una rama.**
  - D. Ninguna de las anteriores.
13. En Vala para que funcione el mecanismo de **señales/manejadores**:
  - A. No es necesario hacer nada, el lenguaje lo proporciona.
  - B. Debemos derivar de la clase GLib.Object.**
  - C. Se debe compilar el código con una opción especial.
  - D. Debemos derivar de la clase Gtk.Object.
14. En Vala una función- $\lambda$  puede hacer de manejador...
  - A. Nunca.

- B. Siempre.
  - C. Cuando coincida su lista de parámetros con los de la señal.**
  - D. Cuando no tenga parámetros.
15. En Vala una señal...
- A. Nunca puede tener implementación.
  - B. Siempre puede tener implementación.
  - C. Sólo podrá tener implementación cuando se declare virtual.**
  - D. Ninguna de las anteriores.

# Herramientas Avanzadas para el Desarrollo de Aplicaciones Marzo 2016

## Modalidad 2

### 1. En Vala una *señal*:

- A. Sólo puede tener conectado un manejador.
- B. Sólo puede estar conectada con métodos de una clase.
- C. Puede tener conectados varios manejadores.
- D. Ninguna de las anteriores.

### 2. 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.

### 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.

### 5. En Git la operación *pull* equivale a:

- A. `fetch+merge`.
- B. `merge`.
- C. `push`.
- D. `rebase`.

### 6. En el patrón de arquitectura MVC:

- A. un modelo puede tener varias vistas sobre él.
- B. una vista puede referirse a varios modelos si estos lo permiten.
- C. una vista puede referirse más de un modelo siempre.
- D. ninguna de las anteriores.

### 7. Los interfaces de usuario generados por *glade*...

- A. Se guardan en archivos binarios.
- B. Se guardan como código fuente Vala.
- C. Constituyen la única manera de dotar de interfaz gráfico a una aplicación que use Gtk+.
- D. Son archivos de texto en formato XML.

**8. En Gtk el modo de leer un elemento de interfaz de usuario creado en glade es:**

- A. Con el método *get\_ui\_element*.
- B. Con el método *get\_widget* o *get\_button*, etc...
- C. Con el método *read\_object*.
- D. Con el método *get\_object*.

**9. En Vala la clase *ArrayList<T>* forma parte...**

- A. Del lenguaje, es un tipo de datos básico.
- B. De nada. Esa clase no existe.
- C. De una biblioteca externa.
- D. Ninguna de las anteriores.

**10. Los SCV, 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.

**11. El método *connect\_after*:**

- A. Sólo admite funciones lambda manejadoras de la señal.
- B. Siempre tiene que ser declarado virtual.
- C. a y b son verdaderas.
- D. a y b son falsas.

**12. En Vala la conexión de las señales con los manejadores:**

- A. debe realizarse siempre en el *main*.
- B. debe realizarse siempre en el constructor.
- C. no es obligatoria su conexión.
- D. ninguna de las anteriores.

**13. 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.

**14. 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.

**15. Las librerías dinámicas**

- A. No hay que compilarlas.
- B. Deben llevar el prefijo lib y la extensión .a.
- C. Hacen crecer el ejecutable final.
- D. Ninguna de las anteriores.

**16. En el interfaz de comandos sqlite para ejecutar 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.

**17. La autoconexión de señales en el marco Glade +Gtk+ se hace:**

- A. sobre un objeto de la clase Window
- B. en un método estático de la clase Window
- C. sobre un objeto de la clase Builder**
- D. en un método estático de la clase Builder

**18. La clase Window**

- A. Pertenece al espacio de nombre GLib
- B. Pertenece al espacio de nombre Gtk**
- C. Pertenece al espacio de nombre Glade
- D. Pertenece al espacio de nombre Widget

**19. Para que una aplicación escrita en Vala que hace uso de Gtk pueda funcionar una vez compilada:**

- A. Hemos tenido que añadir una línea así: `"using Gtk;"`.
- B. Hemos tenido que iniciar Gtk: `"Gtk.init (ref args)"`.**
- C. Basta con añadir la opción del compilador: `"--pkg gtk-3.0"`.
- D. No hay que hacer nada especial.

**20. En Vala una señal...**

- A. Nunca puede tener implementación.
- B. Siempre puede tener implementación.
- C. Sólo podrá tener implementación cuando se declare *virtual*.**
- 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/GObject.**
- C. No usa el concepto de *señales/manejadores*.
- D. Ninguna de las anteriores.

**23. En el patrón de arquitectura MVC aplicado en Vala usando Gtk+:**

- A. El modelo es representado por una clase y el controlador sería implementado por la librería Gtk+.
- B. El modelo es representado por una clase, y el controlador por una clase interfaz.
- C. La vista se implementa siempre con Glade, en formato XML.
- D. B y C son ciertas.

**24. A una *señal* con visibilidad *pública*:**

- A. Sólo le podemos conectar manejadores públicos.
- B. Sólo le podemos conectar funciones independientes de cualquier clase.
- C. Le podemos conectar cualquier manejador independientemente de su visibilidad.
- D. Ninguna de las anteriores.

**25. ¿Cuál de las siguientes afirmaciones es falsa?**

- A. A una señal podemos conectarle una función lambda.
- B. A una señal podemos conectarle una función anónima.
- C. A una señal no le podemos conectar una función lambda.
- D. Una señal puede ser desconectada de su callback.