# PRUEBA TEORICA DESARROLLADOR WEB

1. **Cual es el comando de git para descargar un repositorio completo de git.**
   1. Push
   2. Fork
   3. Clone
   4. Commit
2. **Cual es el comando para subir los cambios realizados en su repositorio local al repositorio central**
   1. Push
   2. Fork
   3. Clone
   4. Commit
3. **Como agrego un mensaje a un commit**
   1. message “Mensage”
   2. add “Mensage”
   3. commit “Mensage”
   4. commit -m “Mensage”
4. **Como cambio de rama en git.**
   1. Pull
   2. Merge
   3. checkout
   4. rebase
5. **Con cual comando mezclo los cambios de otra rama (Seleccione 2)**
   1. checkout
   2. merge
   3. rebase
   4. pull
6. **Como veo las modificaciones que he realizado a un archivo antes de hacer un commit.**
   1. Log
   2. status
   3. diff
   4. tag
7. **Para que sirve el constructor de una clase**
   1. Manejar la seguridad
   2. Definir atributos
   3. Inicializar atributos
   4. Instanciar atributos
8. **Como defino parámetros opcionales**
   1. Inicializándolo en la firma del método
   2. Anteponiendo el signo “?”
   3. En PHP no es necesario
   4. Por medio del type hinting
9. **Que es la inyección de dependencias**
   1. Una función
   2. una recomendación de symfony
   3. Un patrón de diseño
   4. Un método
10. **Como se llama el patrón de diseño donde siempre se retorna la misma instancia.**
    1. DI
    2. Facade
    3. Factory
    4. Singleton Patron de diseño que siempre que se invoque un método devuelve el mismo objeto.
11. **Cual es la manera correcta de importar PDO**
    1. use PDO; $pdo = new PDO; independiente de la base de datos, se implementan las mismas funciones.
    2. require\_once 'PDO'; $pdo = new PDO;
    3. include PDO; $pdo = new PDO;
    4. use PDO; $pdo = new PDO();
12. **Para que sirven los traits**
    1. Dotar de características o rasgos las clases
    2. Para limitar la ejecución reducir la limitación de la herencia.
    3. Para optimizar la carga
    4. Los traits no aportan nada
13. **Marque dos de los principales beneficios de composer**
    1. Ayuda a extender otros objetos
    2. Maneja las dependencias
    3. Maneja la auto carga de clases
    4. Maneja compatibilidad con versiones anteriores
14. **Para que sirve el autowiring en symfony**
    1. Gestionar la memoria del sistema
    2. Configurar las conexiones
    3. Preconfigurar las dependencias requeridas Permite manejar o configurar ciertas dependencias
    4. Gestiona toda la inyección de dependencias
15. **Para que sirven los los voters**
    1. Para generar encuestas
    2. Para generar contenido en una entidad
    3. Para gestionar permisos granularmente
    4. Para gestionar usuarios
16. **Que es symfony flex**
    1. La nueva version de symfony
    2. Un Plug-in de composer
    3. Un nuevo maquetador basado en css
    4. Un componente de symfony
17. **Para que sirve el bundle maker**
    1. Para compilar la aplicación
    2. Para generar componentes vacios y evitar la repetición
    3. Para automatizar el despliegue de la aplicación
    4. Para generar binarios

# PRUEBA TÉCNICA DESARROLLADOR WEB

**1. Se debe crear un CRUD para una tienda online:**

* Debe tener las siguientes entidades base: Product, Customer, Sale.
* Puede agregar las entidades adicionales que considere conveniente.
* El proyecto debe ser versionado en git.
* Puede utilizar cualquier framework PHP orientado a objetos, o si desea con PHP puro. (Preferible Symfony 4 o superior)
* Puede utilizar cualquier sistema de base de datos SQL relacional.

**Reglas de negocio.**

* La tienda puede tener muchos productos
* La tienda puede tener muchos Clientes
* Los clientes pueden comprar 1 ó muchos productos
* Si compra más de 5 productos de la misma referencia se aplica un 10% de descuento al precio de venta del producto.
* Debe simular un stock, es decir no puede vender más productos de los que hay en existencia.
* Agregar validaciones con javascript a los formularios.

**2. Informes**

* Clasificación de los primeros 5 clientes con más compras en el mes.
* Clasificación primeros 5 productos más vendidos en el mes
* Mostrar el valor total por producto y por cliente comprados en un rango de fechas.
* Ej:

|  |  |  |
| --- | --- | --- |
| Cliente | Producto | Total |
| Pepito Perez | Producto 1 | $1200000 |
| Pepito Perez | Producto 2 | $5200000 |
| Paco Lopez | Producto 3 | $530000 |
| Paco Lopez | Producto 1 | $230000 |

**3. API REST**

* Debe crear un endpoint que me devuelva la lista de productos en stock.
* Crear otra vista que consuma este endpoint y muestra la lista de de productos
* Crear un formulario para filtrar por código de producto, y me devuelva la información del producto y el stock sin recargar la página utilizando javascript. *Ej endpoint: api/product/{idProduct} 77023445323*

**4. GIT**

* Debe crear un repositorio en gitlab o github y subir el código versionado.