



MATERIAL UTILIZABLE:

- para la parte TEST, ningún material
- para la parte práctica: todo el material de este módulo profesional y consulta a internet
- la parte práctica se realizará sobre el entorno de desarrollo Visual Studio Code, disponible en una máquina virtual (se podrá elegir entre entorno en Windows o entorno en GNU/Linux Ubuntu, en caso de máquina virtual).
- si se desea, se puede traer portátil personal para la realización de la prueba, que se podrá conectar a la red local del aula (debe disponer de interfaz de cable Ethernet) (debe tener software instalado de grabación de pantalla, por ejemplo OBS): se pedirá la grabación de toda la duración del examen. Si se trabaja con 2 pantallas, se deben grabar las 2 pantallas.

PARTE 1: TEST

(2 puntos)

PREGUNTA 1: (ver modelos en fpadistancia)

PARTE 2: PREGUNTAS PRÁCTICAS

(8 puntos)

Antes de comenzar a trabajar con el entorno de desarrollo integrado, VSC, se recomienda:

- 1) Configurar la variable de entorno PATH para que apunte también al directorio de PHP de Xampp
- 2) Instalar composer (recuerda que posiblemente haya que deshabilitar xdebug en el php.ini durante la instalación).

Criterios de puntuación:

- 1) CORRECCIÓN DEL CÓDIGO: **no importa en absoluto que haya mucho código o poco código**: lo que importa es que el código que desarrolles cumpla con las especificaciones indicadas en el enunciado, como mínimo
- 2) MEJORAS: cualquier prestación adicional que añadas, que no habiendo sido especificada explícitamente, pero no vaya contra las especificaciones indicadas, se puntuará positivamente.
- 3) LEGIBILIDAD del código (sangrado, espaciado de elementos, ...)
- 4) DOCUMENTACIÓN DEL CÓDIGO: comentarios correctos

Para cada pregunta: crea 2 carpetas (pregunta2, pregunta3) y guarda todo el código y recursos necesarios correspondientes a cada pregunta en cada carpeta correspondiente. Luego entrega, por favor, las 2 carpetas en un .zip, JUNTO CON ESTE FICHERO CON TUS RESPUESTAS Y COMENTARIOS.

PREGUNTA 2, problema de programación en PHP, contra base de datos y con clases: (5 puntos)

NOTA: En todo este ejercicio se debe respetar el criterio seguido durante el curso de que:

- las páginas "públicas" (=a servir a internet) se guardan en el directorio "public"
- las páginas de soporte en "src"

a) (2 puntos)

Desarrolla una página web "pagina1.php" que solicite del usuario una frase que solo tenga letras minúsculas sin acentuar y sin signos de puntuación ni otro carácter extraño (es decir, abcdefghijklmnopqrstuvwxyz).

- a.1) si la frase no cumple con lo anterior: se le indicará al usuario por qué, y se solicitará de nuevo su introducción.
- a.2) si la frase valida correctamente:
 - se escribirá la frase del revés
 - se indicará si la frase es un palíndromo o si no lo es.
- a.3) además, se mostrará en una tabla html (TABLA) las frases que ese usuario ha ido introduciendo hasta ese momento, incluida la actual (después de darle a "Enviar")

Por ejemplo: (las entradas aparecen en fondo gris, las salidas calculadas en fondo azul)

(Enviar y Reset representan botones html)

Introduzca una frase con palabras con sólo letras minúsculas sin acentuar: dabale arroz a la zorra el abad

Enviar Reset

Escrita al revés: daba le arroz al a zorra elabad
Sí es un palíndromo!!!



Frases probadas hasta el momento:

Palíndromo	Frase
no	refrescar no molestar
no	la casa era muy bonita
sí	dabale arroz a la zorra el abad

Introduzca una frase con palabras con sólo letras minúsculas sin acentuar: **reconocer**

Enviar **Reset**

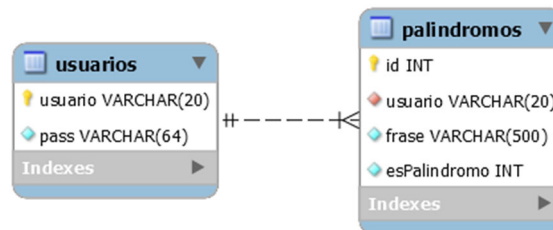
Escrita al revés: **reconocer**
Sí es un palíndromo!!!

Frases probadas hasta el momento:

Palíndromo	Frase
no	refrescar no molestar
no	la casa era muy bonita
sí	dabale arroz a la zorra el abad
sí	reconocer

b) (1 punto)

Descarga de FPADISTANCIA la base de datos "pregunta2.zip" y cárgala en tu entorno de desarrollo del examen.



Desarrolla una página "login.php" que permita autenticar a un usuario contra la tabla "usuarios" de la base de datos "pregunta2".

- si el usuario se autentica correctamente, será redirigido a la página pagina1.php, pasándole a esta (pagina1.php) los **datos de sesión** con el nombre de usuario validado.
- modificar la pagina1.php para que muestre el nombre del usuario "logueado"

El nombre de usuario será una cadena sin espacios, de 10 caracteres como máximo, y los caracteres serán letras minúsculas o mayúsculas del alfabeto castellano (debe validarse la entrada).

c) (2 puntos)

Añade un nuevo botón a la pagina1.php "Persistir" (y toda la lógica (=página/s php) subyacente) que permita almacenar en la base de datos, en la tabla "palindromos" la información que en ese momento el usuario tenga en la tabla TABLA.

- la actualización se realizará usando AJAX (librería Jaxon-php), sin que se tenga que recargar la página pagina1.php
- no se enviarán, en una misma sesión, aquellas filas de la tabla TABLA que ya fueron enviadas.

**PREGUNTA 3, problema de modificación y/o corrección de código en PHP: (3 puntos)**

Como resultado de esta pregunta:

- a) se completarán los apartados "**INFORME DEL TRABAJO REALIZADO**" (una lista de pasos principales seguidos)
- b) se entregará el código final desarrollado tal como se indica en las instrucciones al comienzo del enunciado de este examen.

Dada la siguiente aplicación que ofrece y utiliza un servicio SOAP en PHP (descargar código de la pregunta en formato .zip de la web de FPADISTANCIA, fichero **pregunta3.zip**) que se corresponde con una solución posible al ejercicio trabajado en la Tarea 06, te solicitan realizar las siguientes tareas.

a) (0,25 puntos)

Carga la base de datos (esquema y datos) en MySQL (MariaDB) de Xampp y arregla lo que haya que arreglar para que la página **public/index.php** funcione correctamente.

INFORME DEL TRABAJO REALIZADO: (muestra también pantallazo de la página index.php mostrando resultado correcto)

b) (0,75 puntos)

Añade al servicio SOAP una nueva función que permita obtener la ciudad de una tienda dada. **(0,75 puntos)**

CONSULTA SQL: `SELECT ciudad FROM tiendas WHERE id = <id de la tienda>`

INFORME DEL TRABAJO REALIZADO: (también, captura de pantalla donde index.php muestra que la función funciona correctamente).

c) (1 punto)

Añade a la página un formulario que permita al usuario introducir:

- un código de tienda
- un código de producto
- un código de familia
- la fecha actual

Requisitos:

- c.1) Deben validarse los datos, como mínimo en el php. Los rangos válidos de dichos códigos se obtengan mediante las oportunas consultas a la base de datos.
- c.2) Además cuando se haga clic en el botón "Enviar", se mostrará la ciudad para el código de tienda introducido usando el servicio SOAP añadido en b).
- c.3) Cuando el usuario haga clic en el botón "Enviar", la página se recargará mostrando la información solicitada y manteniendo el valor de los datos introducidos en los campos del formulario (mantener la sesión)

d) (1 punto)

Añade la prestación de AJAX (usando Jaxon-php) a la aplicación (fichero index.php), consistente en que aparezca un botón "Clima" que al hacer clic en él, permita obtener pequeño informe del tiempo (obtenido usando un **servicio REST** de tu elección) en la ciudad de la tienda introducida por el usuario en ese momento, sin recargar la página index.php

INFORME DEL TRABAJO REALIZADO: