

Ejercicios UP3-1 y UP3-2 Formularios y su procesamiento

DWES - 2DAW

NOTA: Para las prácticas deberéis poner vuestro nombre en el código y el título de la página. Se deberá comprobar que los tipos de los datos introducidos son los esperados usando las validaciones vistas en la unidad.

1. Desarrolla un conversor entre euros y pesetas ($1\text{€}=166,386$ ptas) que convierta de euros a pesetas y de pesetas a euros según lo que elija el usuario (de forma excluyente) y por la cantidad que introduzca. Comprueba que los datos introducidos son los esperados.

RA1.b, RA1.f, RA2.a, RA2.b, RA2.c, RA2.d, RA2.e, RA2.f, RA2.g, RA2.h, RA3.a, RA3.b, RA3.c, RA3.d, RA3.e, RA3.f, RA3.g, RA4.f

2. Escribe un programa que calcule el salario semanal de un trabajador teniendo en cuenta que las horas ordinarias (40 primeras horas de trabajo) se pagan a 12 euros la hora. A partir de la hora 41, se pagan a 16 euros la hora. Por defecto se mostrará sólo el salario semanal pero se le dará la opción al usuario de marcar un check para mostrar también el salario mensual.

RA1.b, RA1.f, RA2.a, RA2.b, RA2.c, RA2.d, RA2.e, RA2.f, RA2.g, RA2.h, RA3.a, RA3.b, RA3.c, RA3.d, RA3.e, RA3.f, RA3.g, RA4.f

3. Elabora un programa que dado un carácter introducido por el usuario y determine si es:

- | | |
|-------------------------|------------------------------|
| 1. una letra mayúscula | 4. un carácter blanco |
| 2. una letra minúscula | 5. un carácter de puntuación |
| 3. un carácter numérico | 6. un carácter especial |

Se debe usar funciones para la comprobación de datos.

RA1.b, RA1.f, RA2.a, RA2.b, RA2.c, RA2.d, RA2.e, RA2.f, RA2.g, RA2.h, RA3.a, RA3.b, RA3.c, RA3.d, RA3.e, RA3.f, RA3.g, RA4.f

4. Diseña un programa que determine la cantidad total a pagar por 5 llamadas telefónicas de duración a introducir por el usuario de acuerdo a las siguientes premisas: Toda llamada que dure menos de 3 minutos tiene un coste de 10 céntimos. Cada minuto adicional a partir de los 3 primeros es un paso de contador y cuesta 5 céntimos. Muestra el tiempo total de cada llamada y su precio final así como el coste de las 5 llamadas en conjunto.

RA1.b, RA1.f, RA2.a, RA2.b, RA2.c, RA2.d, RA2.e, RA2.f, RA2.g, RA2.h, RA3.a, RA3.b, RA3.c, RA3.d, RA3.e, RA3.f, RA3.g, RA4.f

5. Dado un vector asociativo de trabajadores con su salario, crea usando funciones y a criterio del usuario, el salario máximo, el salario mínimo y el salario medio. (puede elegir uno de ellos, varios o todos). Además deberás permitir que el usuario elija si desea aplicar un porcentaje a los salarios indicando la cantidad del mismo y mostrando los salarios actuales y con el porcentaje ya aplicado.

RA1.b, RA1.f, RA2.a, RA2.b, RA2.c, RA2.d, RA2.e, RA2.f, RA2.g, RA2.h, RA3.a, RA3.b, RA3.c, RA3.d, RA3.e, RA3.f, RA3.g, RA4.f

6. Realiza el control de acceso a una caja fuerte. La combinación será un número de 4 cifras. El programa nos pedirá la combinación para abrirla. Si no acertamos, se nos mostrará el mensaje “Lo siento, esa no es la combinación” en color rojo y si acertamos se nos dirá “La caja fuerte se ha abierto satisfactoriamente” en color verde. Tendremos cuatro oportunidades para abrir la caja fuerte (muestra la combinación para poder acertar en las pruebas).

RA1.b, RA1.f, RA2.a, RA2.b, RA2.c, RA2.d, RA2.e, RA2.f, RA2.g, RA2.h, RA3.a, RA3.b, RA3.c, RA3.d, RA3.e, RA3.f, RA3.g, RA4.f

7. Crea un programa donde se le seleccione el curso 1 o 2 (radiobutton), los módulos (a seleccionar de un desplegable) y las horas (marcar o desmarcar) y genere un horario usando una tabla.

RA1.b, RA1.f, RA2.a, RA2.b, RA2.c, RA2.d, RA2.e, RA2.f, RA2.g, RA2.h, RA3.a, RA3.b, RA3.c, RA3.d, RA3.e, RA3.f, RA3.g, RA4.f

8. Escribe un formulario que solicite una dirección de correo, que la confirme e indique si acepta recibir publicidad. Añade botón Enviar y Borrar. El botón borrar se mantendrá en el mismo formulario inicial pero limpiará todos los campos. Cuando pulsemos Enviar, iremos a otra página donde se le indique el email y si ha aceptado recibir publicidad o no con un botón que volverá a la página inicial.

RA1.b, RA1.f, RA2.a, RA2.b, RA2.c, RA2.d, RA2.e, RA2.f, RA2.g, RA2.h, RA3.a, RA3.b, RA3.c, RA3.d, RA3.e, RA3.f, RA3.g, RA4.a, RA4.b, RA4.f

9. Escribe un formulario de recogida de datos que conste de dos páginas: En la primera página se solicitan los datos y se muestran errores tras validarlos (botones Validar y Enviar). En la segunda página se muestra toda la información introducida por el usuario si no hay errores (tendrá botón Volver a la página inicial). Los datos a recoger son datos personales, nivel de estudios (desplegable), situación actual (selección múltiple: estudiando, trabajando, buscando empleo, desempleado) y hobbies (marcar de varios mostrados y poner otro con opción a introducir texto).

RA1.b, RA1.f, RA2.a, RA2.b, RA2.c, RA2.d, RA2.e, RA2.f, RA2.g, RA2.h, RA3.a, RA3.b, RA3.c, RA3.d, RA3.e, RA3.f, RA3.g, RA4.a, RA4.b, RA4.f

10. Escribe un formulario de recogida de datos que conste de dos páginas: En la primera página se solicitan los datos y se muestran errores tras validarlos (botones Borrar, Validar y Enviar). En la segunda página se muestra toda la información introducida por el usuario si no hay errores (con botón Volver a la página inicial). Los datos a introducir son: Nombre, Apellidos, Edad, Peso (entre 10 y 150), Sexo, Estado Civil (Soltero, Casado, Viudo, Divorciado, Otro) Aficiones: Cine, Deporte, Literatura, Música, Cómic, Series, Videojuegos.

RA1.b, RA1.f, RA2.a, RA2.b, RA2.c, RA2.d, RA2.e, RA2.f, RA2.g, RA2.h, RA3.a, RA3.b, RA3.c, RA3.d, RA3.e, RA3.f, RA3.g, RA4.a, RA4.b, RA4.f

11. Crea una página Web para obtener los siguientes datos: Nombre completo, Contraseña (mínimo 6 caracteres), Nivel de Estudios(Sin estudios, Educación Secundaria Obligatoria, Bachillerato, Formación Profesional, Estudios Universitarios), Nacionalidad (Española, Otra), Idiomas (Español, Inglés, Francés, Alemán Italiano), Email, Adjuntar Foto (sólo extensiones jpg, gif y png, tamaño máximo 50 KB). Además de las comprobaciones de validación, se debe comprobar que sube fichero, que el fichero tiene extensión (puedes usar `explode()`) y ésta es válida, que hay directorio donde guardarlo y que se genera con nombre único. (botones Limpiar, Validar y Enviar) Si todo ha ido bien, redirige al usuario a una página donde se le indique que se ha procesado con éxito e incluye tu nombre y grupo de clase (con botón Volver).

RA1.b, RA1.f, RA2.a, RA2.b, RA2.c, RA2.d, RA2.e, RA2.f, RA2.g, RA2.h, RA3.a, RA3.b, RA3.c, RA3.d, RA3.e, RA3.f, RA3.g, RA4.a, RA4.b, RA4.f

12. Crea un script con funciones de validación donde uses expresiones regulares con `preg_match` donde muestres la prueba con una cadena que cumpla el patrón (cadena válida) y otra cadena no válida. Las funciones de validación se aplicarán a:

- a) Genera el patrón para códigos postales de la Comunidad Valenciana
- b) Genera el patrón para un NIF
- c) Genera el patrón para fecha en formato dd/mm/aaaa o bien dd-mm-aaaa
- d) Genera el patrón para una cadena que contenga enviado tanto en mayúsculas como en minúsculas
- e) Genera el patrón para texto con letras mayúsculas/minúsculas y espacios
- f) Genera el patrón para solamente números, sin espacios
- g) Genera el patrón para números con espacios
- h) Genera el patrón para texto en blanco, números, mayúsculas/minúsculas y caracteres acentuados
- i) Genera el patrón para el caso anterior añadiendo los signos de puntuación: comillas simples, coma, punto, punto y coma, dos puntos y guiones
- j) Genera el patrón para validar una dirección de email
- k) Genera el patrón para validar una URL sencilla (<http://www.ieslasenia.org/ejercicio?16>)
- l) Genera el patrón para validar una contraseña con al menos un carácter en minúscula, una mayúscula, un número y al menos 6 caracteres de longitud
- m) Genera el patrón para validar una IPv4
- n) Genera el patrón para validar una MAC separada por :

RA2.d, RA2.e, RA2.f, RA2.g, RA2.h, RA3.a, RA3.b, RA3.c, RA3.d, RA3.g

13. Crea una página Web con los campos necesarios para poder probar las funciones llevadas a cabo en el ejercicio anterior en el momento de validar. Cuando todo haya sido validado, redirigir al usuario a una página indicando el valor de cada campo y si el resultado ha sido válido o si no cumple la función de validación.

RA1.b, RA1.f, RA2.a, RA2.b, RA2.c, RA2.d, RA2.e, RA2.f, RA2.g, RA2.h, RA3.a, RA3.b, RA3.c, RA3.d, RA3.e, RA3.f, RA3.g, RA4.a, RA4.b, RA4.f