**Examen Ras 2 y 3 de PHP**

**2º Desarrollo de Aplicaciones Web. Desarrollo de Aplicaciones Web en Entorno Servidor**

Duración: 3 horas

Instrucciones :

- Realiza cada ejercicio en archivos PHP separados, según las indicaciones.

- Asegúrate de subir los archivos al servidor local y que funcionen correctamente.

- Incluye comentarios para explicar cada bloque de código.

**Contexto de la empresa:**



Eres el nuevo desarrollador de la empresa **"HR Solutions"**, una firma que proporciona servicios de recursos humanos, incluyendo la búsqueda, selección y seguimiento del personal para diversas empresas. Hoy, tu misión es asistir al equipo de IT en el desarrollo de herramientas internas que automatizan procesos clave relacionados con el personal.

¡Tu desempeño puede abrirte las puertas para un contrato fijo en la empresa!

**Parte 1: Comandos básicos de PHP (30 minutos)**

**Ejercicio 1.1: Saludo personalizado según el día de la semana**

Situación:

Cuando un empleado ingresa al portal de "HR Solutions", debe recibir un saludo personalizado dependiendo del día de la semana y la hora del acceso.

Tarea:

1. Crea un archivo llamado `bienvenida.php`. Dentro de este archivo:

- Muestra tu nombre como desarrollador responsable.

- Muestra el día de la semana actual.

- Dependiendo del día, muestra un mensaje diferente:

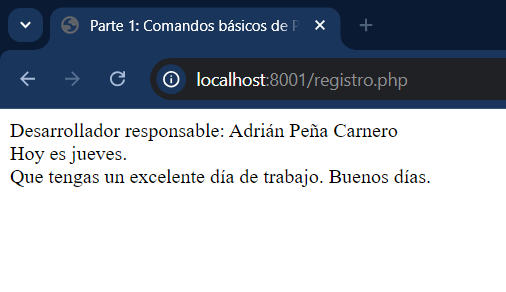
- Si es lunes, muestra "Feliz inicio de semana".

- Si es viernes, muestra "¡Feliz viernes, prepárate para el fin de semana!".

- Para el resto de los días, muestra "Que tengas un excelente día de trabajo".

- Si es antes de las 12:00, añade al mensaje "Buenos días". Si es después, añade "Buenas tardes".

Prueba localhost:



**Parte 2: Funciones en PHP (30 minutos)**

**Ejercicio 2.1: Conversión de horas trabajadas a minutos**

Situación:

El equipo de "HR Solutions" quiere llevar un registro preciso del tiempo trabajado por los empleados, y para ello necesitan convertir las horas trabajadas a minutos.

Tarea:

1. Crea un archivo llamado `horas.php`. Dentro de este archivo:

- Define una función llamada `convertirHorasAMinutos($horas)` que reciba como parámetro una cantidad de horas y las convierta a minutos.

- Usa esta función para convertir las siguientes horas de trabajo: 8 horas, 5.5 horas y 3.25 horas.

- Muestra el resultado de cada conversión en pantalla.

**Ejercicio 2.2: Cálculo de salario semanal**

Situación:

El equipo de RRHH necesita calcular el salario semanal de varios empleados. Cada uno tiene un número distinto de horas trabajadas.

Tarea:

1. En el mismo archivo `horas.php`, crea una función llamada `calcularSalarioSemanal($horasTrabajadas, $tarifaPorHora)` que reciba como parámetros el número de horas trabajadas en una semana y la tarifa por hora, y devuelva el salario total.

- Usa esta función para calcular el salario de los siguientes empleados:

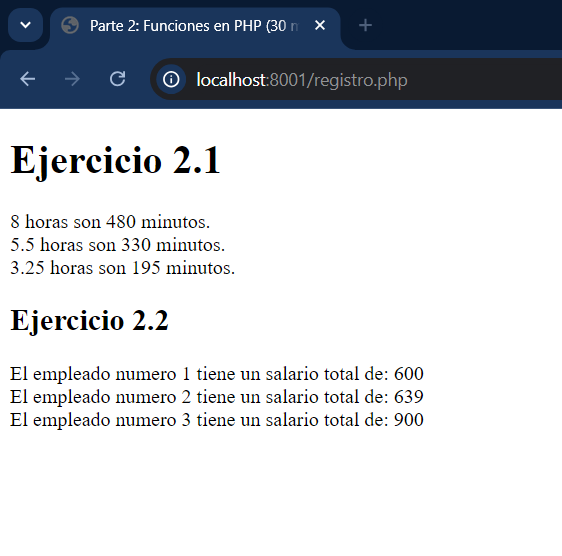
- Empleado 1: 40 horas trabajadas, tarifa de $15/hora.

- Empleado 2: 35.5 horas trabajadas, tarifa de $18/hora.

- Empleado 3: 45 horas trabajadas, tarifa de $20/hora.

- Muestra los salarios calculados en pantalla.

Prueba en localhost:



**Parte 3: Arrays en PHP (30 minutos)**

**Ejercicio 3.1: Gestión de candidatos en espera**

Situación:

El equipo de "HR Solutions" tiene una lista de candidatos que están en espera para ser llamados a entrevistas. Necesitan gestionar esta lista de manera eficiente.

Tarea:

1. Crea un archivo llamado `candidatos.php`. Dentro de este archivo:

- Crea un array llamado `$candidatosEnEspera` con los siguientes nombres: "Carlos", "Sofía", "Andrés", "María", "Daniel".

- Un nuevo candidato, "Isabel", se une a la lista de espera. Agrégalo al inicio del array.

- Muestra el total de candidatos en espera.

- Elimina el primer candidato de la lista (el que ya fue entrevistado) y muestra nuevamente la lista completa en pantalla.

**Ejercicio 3.2: Inventario de equipos de oficina**

Situación:

El equipo de administración quiere llevar un control del inventario de equipos de oficina, con los nombres de los productos y sus cantidades disponibles.

Tarea:

1. En el mismo archivo `candidatos.php`, crea un array asociativo llamado `$inventario` con los siguientes productos y cantidades:

- "Impresora" => 5

- "Escritorio" => 12

- "Silla" => 20

- "Teléfono" => 8

2. Muestra cada producto con su cantidad en pantalla.

3. Calcula y muestra el total de equipos disponibles (sumando todas las cantidades).

Prueba en el localhost:

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

**Parte 4: Formularios en PHP (30 minutos)**

**Ejercicio 4.1: Registro de empleado nuevo**

Situación:

"HR Solutions" necesita un formulario en su portal para registrar a nuevos empleados en su sistema de recursos humanos.

Tarea:

1. Crea un archivo llamado `registro.php` que contenga un formulario HTML con los siguientes campos:

- Nombre completo (campo de texto)

- Edad (campo de número)

- Correo electrónico (campo de texto)

- Botón de envío

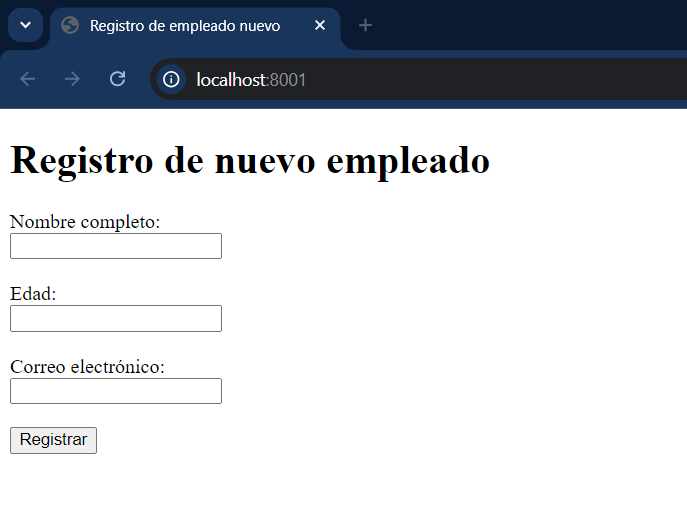
2. Cuando el formulario se envíe, debe procesarse en el mismo archivo utilizando el método POST:

- Verifica si el correo tiene un formato válido.

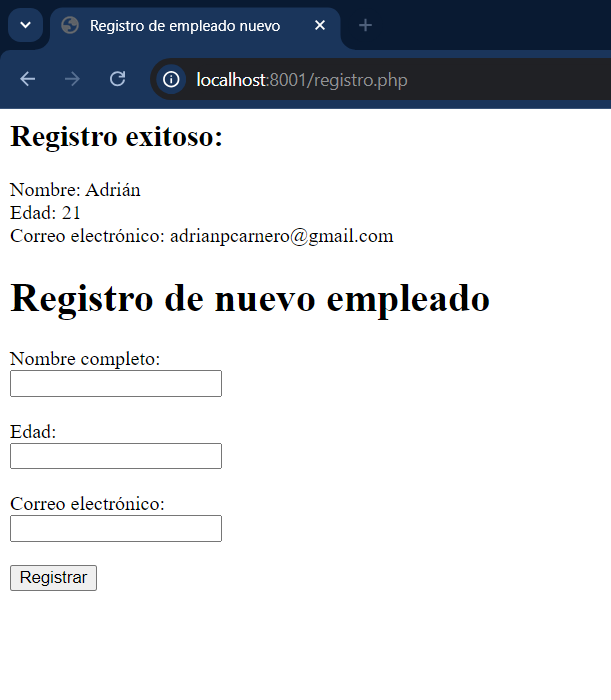
- Verifica si la edad es un número positivo.

- Si todos los datos son válidos, muestra un mensaje de éxito con los datos ingresados. Si no, muestra un mensaje de error adecuado.

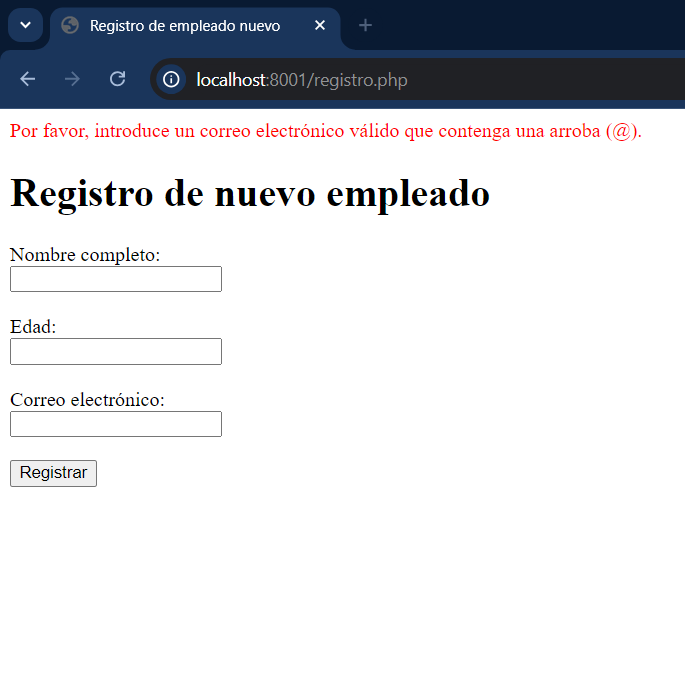
Prueba en el localhost:



Si introducimos todo correctamente:



Correo si el @:



Edad con número negativo:



Evaluación:

- Comandos básicos: 10 puntos

- Funciones: 20 puntos

- Arrays: 30 puntos

- Formularios: 30 puntos

- Calidad del código (comentarios, buenas prácticas) : 10 puntos

Total: 100 puntos