

Nombre y apellidos:	Nota:
---------------------	-------

Hasta 8 puntos.

Se entregará una carpeta comprimida con los ficheros generados en los ejercicios.

El formato de la carpeta será: DWES_231121_NombreApellido_2DAW.zip

1) Piedra, papel o tijeras (3 puntos)

Realizar un script que simule el tradicional juego de piedra papel o tijeras. **(1,5puntos)**

- Habrá un único jugador que jugará contra la máquina y podrán jugar todas las veces que se desee. Al jugador se le mostrarán las tres opciones posibles entre las que deberá seleccionar una, y un botón “Jugar” para enviar al servidor la jugada realizada. Una vez terminada la partida, se volverán a mostrar las opciones y el botón para volver a jugar.
- En el servidor se generará una jugada aleatoria de las tres posibles y se comparará con la introducida por el usuario.
- La página que se devuelve al usuario mostrará, el texto “Has elegido XXX y la máquina ha elegido YYY” y a continuación, si el resultado ha sido que ha habido empate, ha ganado el jugador o ha ganado la máquina.

Se dará la posibilidad de realizar una partida larga (al mejor de 5 por ejemplo), para ello el sistema debe ser capaz de mostrar el número de partidas ganadas, empatadas y perdidas que lleva el usuario y la máquina en cada una de las rondas. **(1,5punto)**

2) Tienda online (5 puntos)

Crea la parte de identificación (login.php) y registro (registro.php) para una tienda online. El formulario de login debe tener un enlace al de registro y viceversa.

Utilizaremos la tabla Alumnos para la realización del ejercicio (alumnos.sql), creada sobre la BBDD dwes

- **Parte de login: (2punto)**

- Si un usuario se identifica correctamente como usuario, el programa recibe el nombre del usuario y muestra el mensaje de ‘Bienvenid@ + nombre de usuario’. La contraseña del alumno siempre se corresponderá con su nombre todo en minúsculas. Ejemplo: **(1p)**

User: IGNACIO

Pass: ignacio

*En caso de no logar esta funcionalidad la contraseña siempre será fija con el valor 1234.

*Es necesario realizar la validación con el valor del registro devuelto por la consulta.

- Si un usuario no se identifica correctamente, en el mismo formulario se muestra un mensaje diciendo "Identificación incorrecta" y se vuelca el mensaje a un fichero externo con un timestamp (hora y tiempo actual del acceso indebido). **(1p)**

- **Parte de registro: (3puntos)**

Para el tratamiento de los datos se creará la clase Alumno que contendrá como propiedades los campos de la tabla (nombre, apellidos, teléfono, correo). Estos campos serán privados por lo que se deberán crear los métodos getter/setter correspondientes. En la parte de registro se deberá trabajar con esta estructura de datos para trabajar con los campos del alumno. **(1p)**

- Si una persona se registra, los datos se envían a la misma página del registro, donde se comprueba que han llegado todos los datos rellenos. Es necesario comprobar al menos que el email contiene un @ y un . antes de hacer la inserción. La validación se realizará mediante un método de la clase Alumno que recibirá el correo y devolverá true/false si se cumple o no el requisito **(1p)**
- Si los datos son correctos, se insertará el registro y posteriormente se redirigirá a la página de login.php en cuyo formulario se mostrará el usuario que se introdujo en el registro para loguearse directamente. **(1p)**