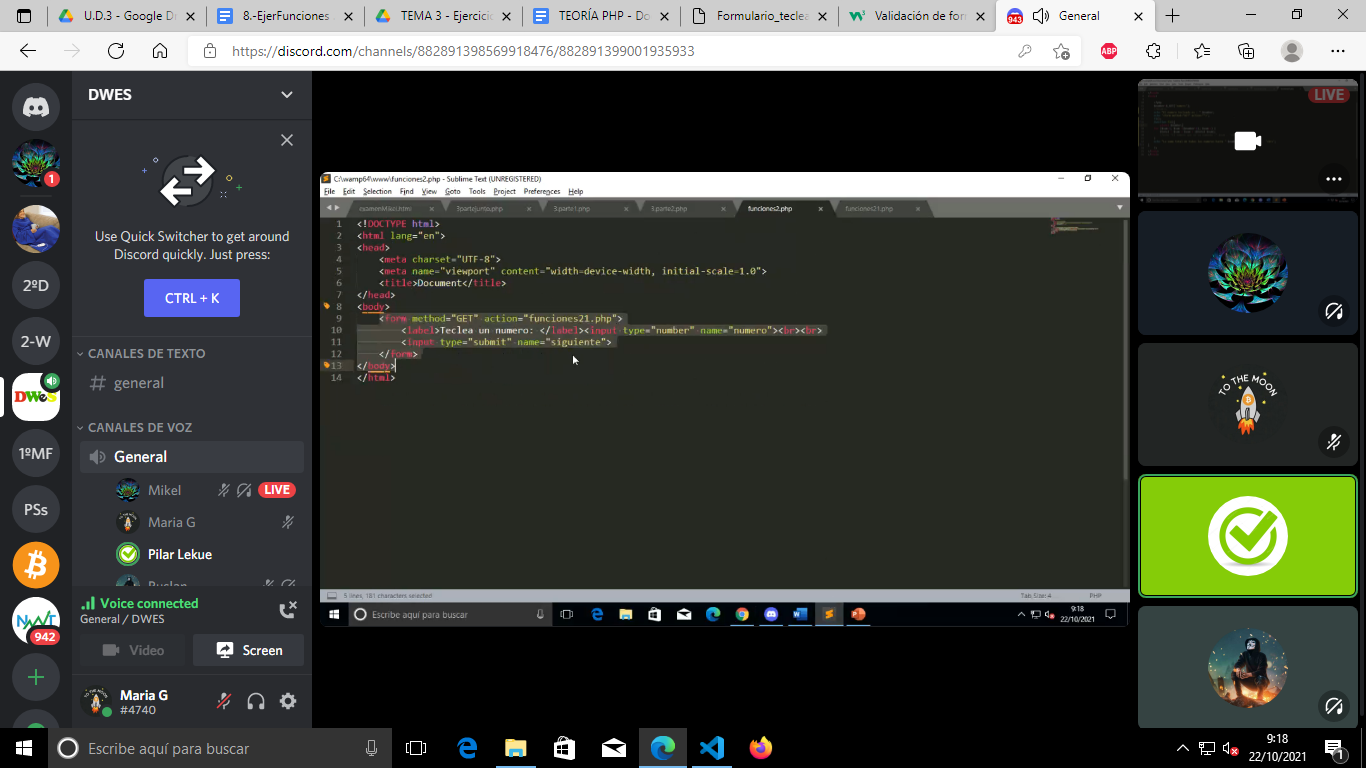
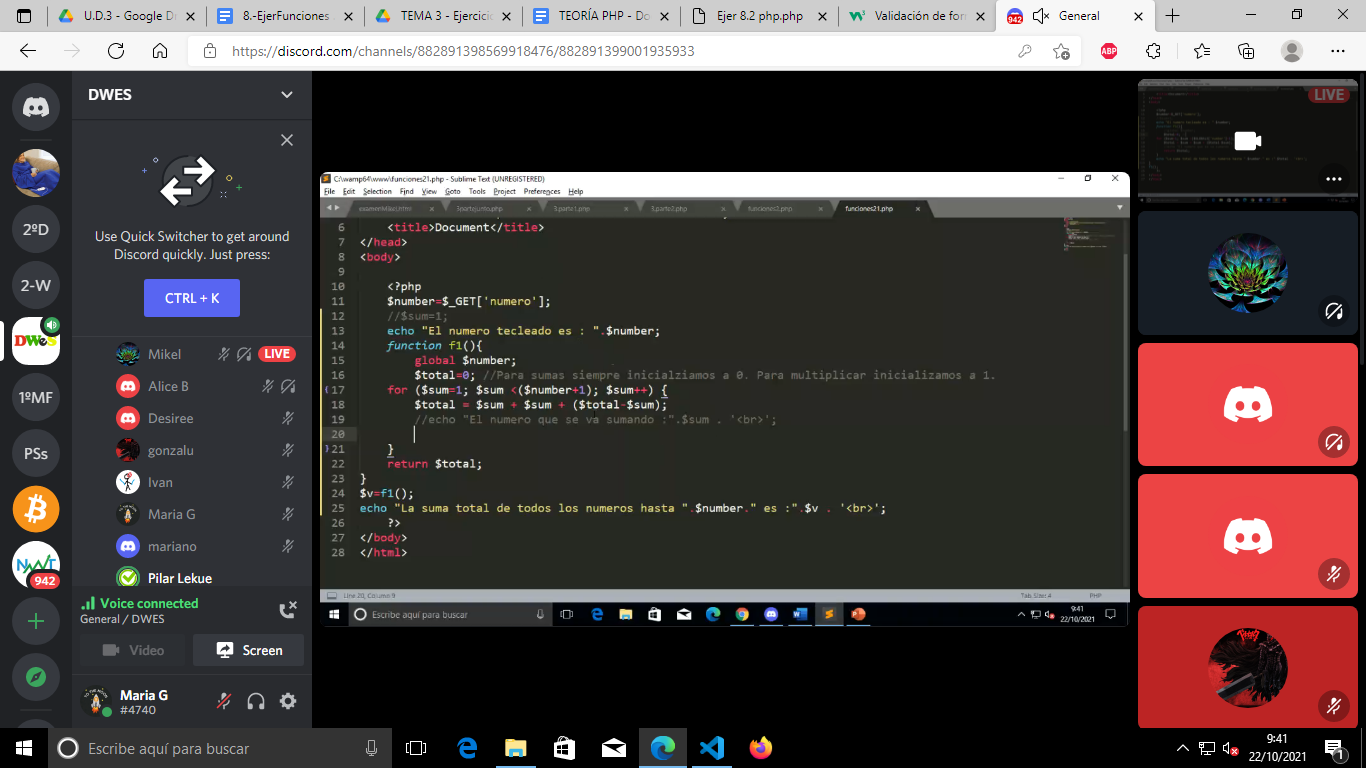
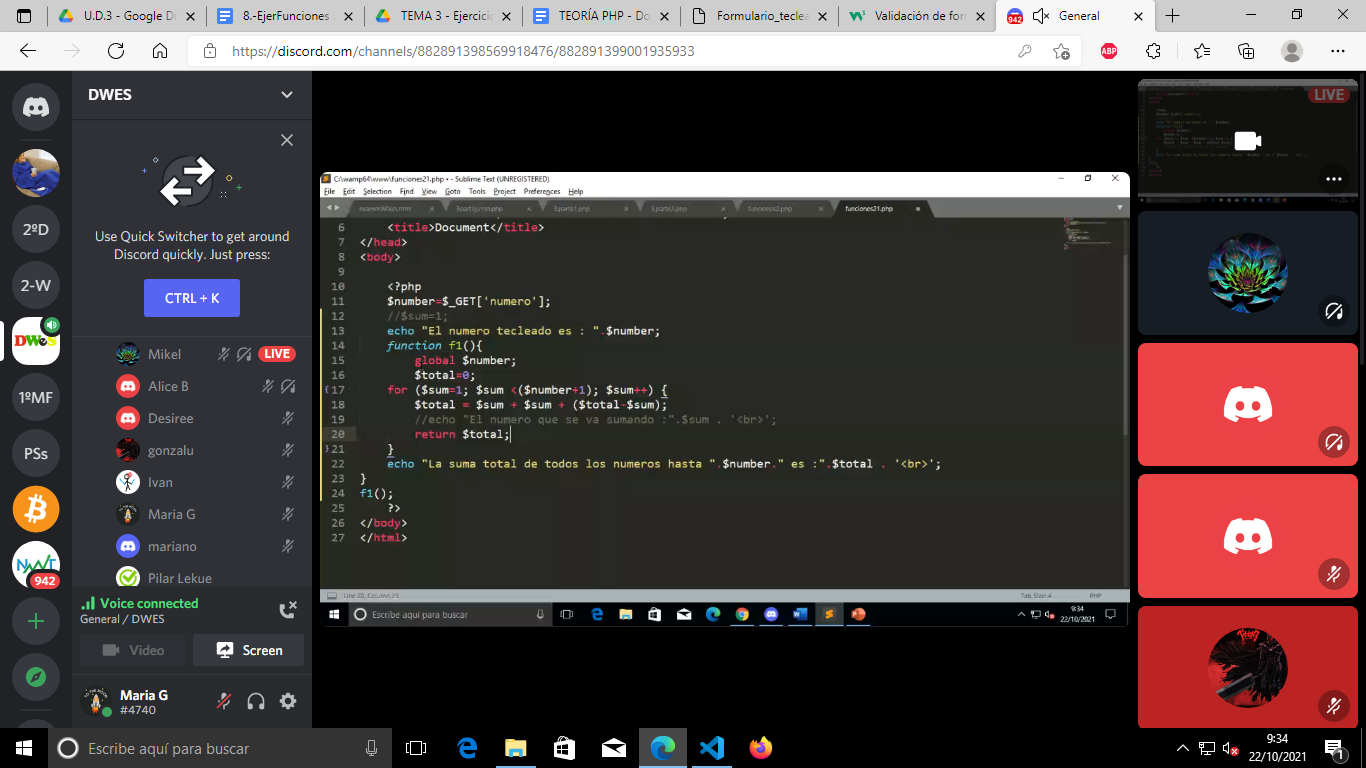
Unidad 3. Programación basada en lenguajes de marcas con código embebido.

Ejercicios:

1. Calculadora.  
    Crear un formulario HTML que consista en pedir dos operandos y un desplegable (<select>) que permita elegir el operador.
2. Formulario para teclear un número.  
   Visualizar el resultado de sumar todos los números desde el 1 hasta el número tecleado.  
   Crear una función.  
   Realizar el ejercicio en 2 páginas y luego en una sola página.

****

****

1. Formulario para introducir 2 números y se debe visualizar el número mayor o el mensaje: son iguales.
2. Escribir el código necesario para que, dado un numero n, calcule su cuadro (n2) y su cubo (n3). (en funciones independientes)

pow(base, exponente), y si tienes la versión P.H.P 5.6 ó superior , puedes utilizar el operador \*\*, es decir: base\*\*exponente.



Resultado: hola-que-tal

function en\_guiones($a){

return str\_replace(" ","-",$a);

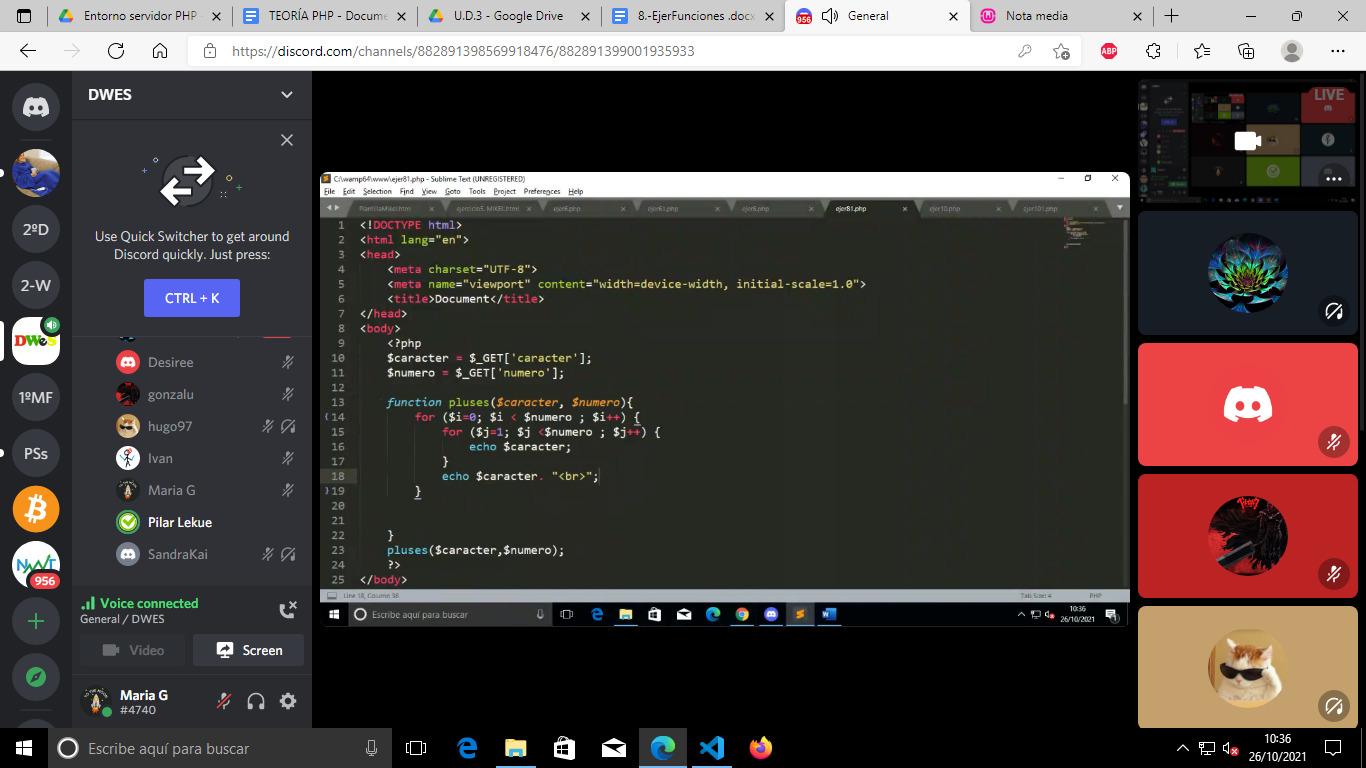
}

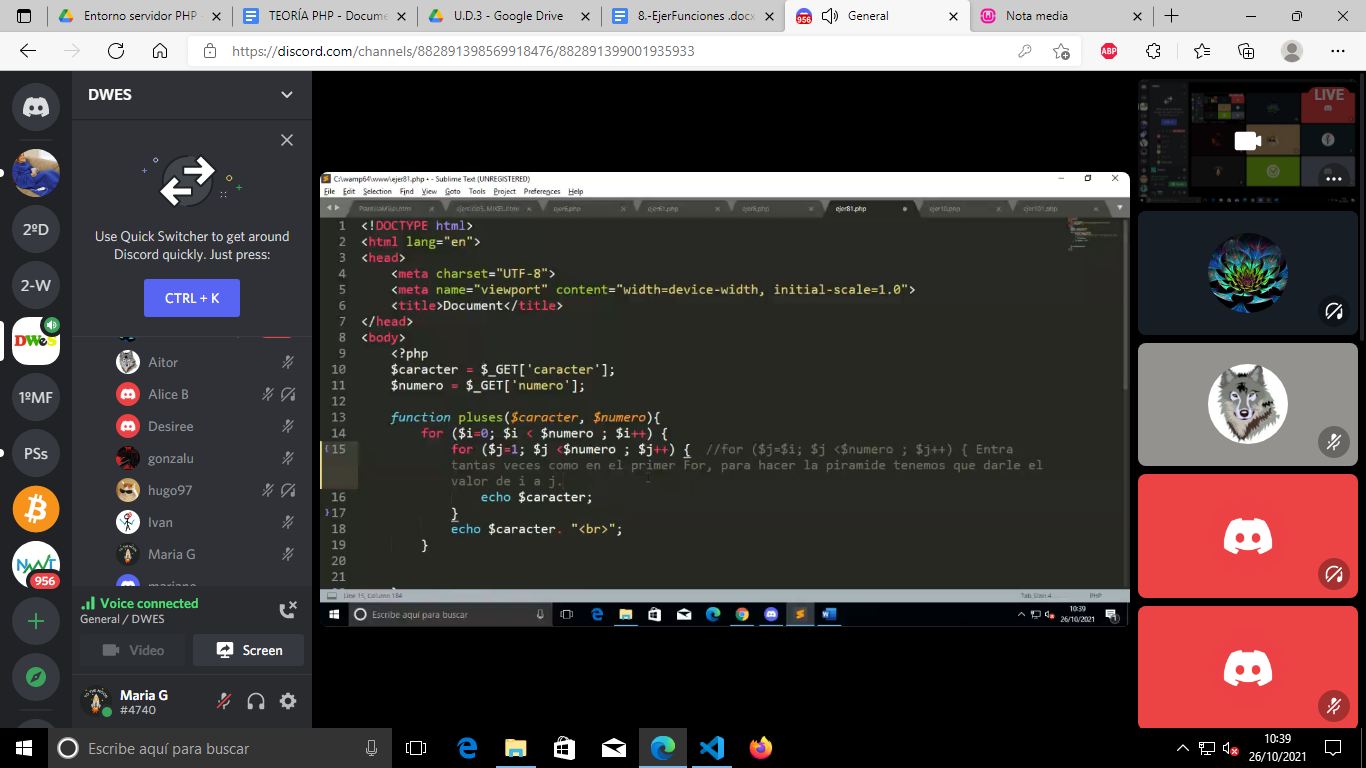
1. Implementar un conversor de moneda: euros – pesetas.
2. Crea una función que recibe 4 números como parámetros de entrada y devuelve un valor, la media de esos números.
3. Crea una función que reciba dos parámetros, un carácter (que puede ser cualquiera) y un número, la función debe mostrar el cuadrado con tantas filas y columnas como indique el número.   
   Por ejemplo: 4 +

++++  
++++  
++++  
++++

1. Crea una función que reciba dos parámetros, un carácter y el número que indicará el número de filas y columnas de la pirámide.

++++  
+++  
++  
+



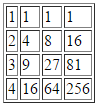


10.-Obtener la siguiente tabla de potencias:

Primera fila: 11,12,13,14

Segunda fila: 21,22,23,24

…



Codificar una función llamada potencia que recibirá 2 parámetros (base y exponente), esta función devolverá el resultado. Dentro de tu función potencia puedes utilizar la función : pow(base, exponente), y si tienes la versión P.H.P 5.6 ó superior , puedes utilizar el operador \*\*, es decir: base\*\*exponente.

