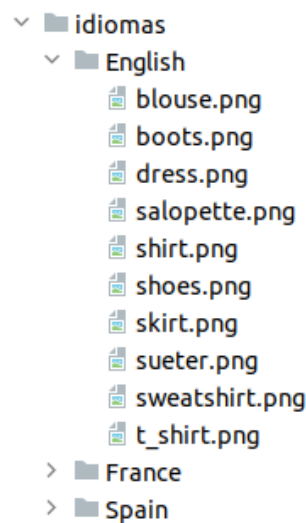


APELLIDOS Y NOMBRE:

Siguiendo la línea del ejercicio explicado en clase, cuyas imágenes puedes ver aquí, junto con la estructura del programa que **debes de seguir** (hay que seguir esta estructura, no se corregirá otro planteamiento).

git clone https://github.com/MAlejandroR/examen_imagenes.git

En este ejercicio tenemos *una serie de directorios con idiomas, categorías* y dentro de cada uno de esos *idiomas unas imágenes*, donde el nombre de la imagen es el nombre de su representación en el idioma establecido (todas las imágenes tienen extensión png):



Las imágenes las tenemos en una estructura de array asociativo, que es un atributo estático y público de la **clase Vocabulario** (ver estructuras de clases en el anexo)

Se pide que implementes la utilidad **realizar examen** con los requisitos que se especifican a continuación: Lo primero mostraremos un menú de selección, donde el usuario aportará su nombre y seleccionará las opciones

Datos del examen

Nombre y apellidos

English Cuerpo

Al presionar hacer el examen nos aparecerán 5 imágenes aleatorias del idioma seleccionado en el select.

Las imágenes nos aparecerán de 1 en una, con cajas de texto vacías para escribir cada una de las letras de esa imagen.

APELLIDOS Y NOMBRE:

Habr  tantas cajas de texto, como letras tenga la imagen, como podemos ver en la siguiente ilustraci n, donde tambi n se aprecia que nos ir  mostrando cada n mero de pregunta

Pregunta n mero 1



[Evaluar](#)

Cuando hayamos terminado las 5 im genes, nos mostrar  el resultado del examen mostrando un informe de cada preguntas.

Vemos en la imagen siguiente un ejemplo de c mo puede quedar

Resultados del examen

Pregunta n mero 1:



Pregunta: **blusa**
Respuesta: **camis**
Letra acertada: 0
FALLADO

Pregunta n mero 2:



Pregunta: **falda**
Respuesta: **falda**
Letras acertadas: 5
ACERTADO

APELLIDOS Y NOMBRE:

En el informe final, aparecerá si hemos aprobado o suspendido y un listado de cada una de las preguntas realizadas con información de:

- La imagen
- La respuesta
- Letras acertadas
- La valoración final: APROBADO o SUSPENSO.
 - Habremos aprobado si hemos acertado:
 - 3 palabras o mas de forma completa
 - 2 palabras de forma completa y de las otras 3, al menos una de ellas con **más de la mitad de las letras**

Requisitos de implementación:

Debes de hacer un listado de los requisitos antes de empezar, p.e.:

RF1: Generar un array con 5 imágenes diferentes

....

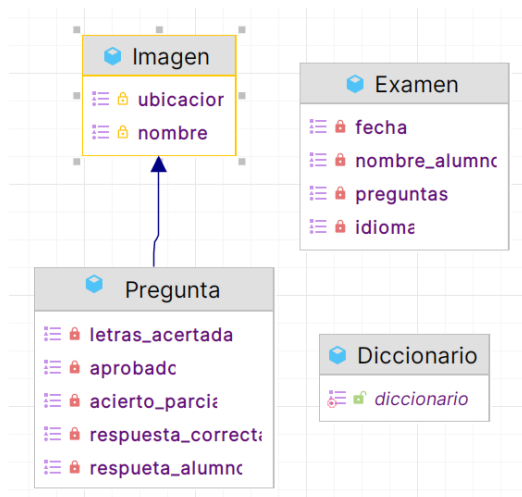
En el código has de indicar las instrucciones que implementan dicho requisito:

Se han de evitar estructuras complejas, haciendo un código legible

Se tiene que seguir la estructura planteada en el git

Anexo

Estructura de clases



Puedes (se valorará), usar las siguientes funciones

- Usa **array_reduce(array, callback)** para reducir un array a un solo valor mediante una función de callback personalizada.
- Aplica **shuffle(array)** para mezclar aleatoriamente los elementos de un array.
- Utiliza **array_splice(\$array, int)** para eliminar o reemplazar elementos desde una posición específica del array.
- Serializa datos con **serialize** para convertirlos en una cadena segura para almacenamiento o transmisión. (no se debe usar `json_decode/json_encode`)
- Codifica cadenas con **base64_encode** y decodificalas con **base64_decode** para manipular datos en un formulario.