

## El módulo lunar.

Agmunsen, nuestro astronauta particular, debe de tomar minerales de la superficie lunar, debe de ir introduciendo los datos de los minerales que vaya encontrando informando sobre varias características de las mismas.

Para cada mineral, Agmun, quiere tomar los valores de:

- Id : Para identificar los minerales.
- Nombre. El identificador debe de tener dos letras, cuatro número y dos letras.  
LLDDDDLL
- Grupo/Origen.
  - ígneas,
  - metamórficas
  - sedimentarias
- Dureza: [Se mide con la Escala de Mohs, que asigna un valor de 1 a 10 según la facilidad con la que un mineral es rayado por otro o por una punta de acero](#)
- Tamaño de grano. El tamaño del grano se clasifica como:
  - Grano muy grueso: > 30 mm
  - Grano grueso: 5 a 30 mm
  - Grano medio: 2 a 5 mm
  - [Grano fino: < 2 mm](#)
- Clasificación.
  - Rocas de construcción: Son aquellas rocas que se usan para la construcción de diferentes estructuras.
  - Rocas ornamentales: Son todas aquellas rocas de uso decorativo, integrado o no en edificio o estructuras superiores.
  - Rocas de uso en utensilios para el hombre.
  - Piedras machacadas: Áridos, ripios, agregados, etc. que son muy utilizados en la construcción como material de relleno.
- Tamaño de cristales. De 0 a 10.
- Temperatura de formación. Desde -100 a 100 grados kelvin
- Estructura. Texto Libre
- Forma de los granos. Texto Libre.
- Textura.
  - Vitrea.
  - Afanítica.
  - Fanerítica.

Agmunsen quiere no depender de un único sistema para introducir los datos, por lo que usando las interfaces va a crear dos formas de introducir los valores.

Una de las formas tendrá labels y textos, la otra forma tendrá placeholder. A la primera la llamaremos **extendida**, a la segunda **reducida**.

Etiqueta: campo Texto	Campo Texto (con placeholder)
-----------------------	-------------------------------

Nuestro control en la tierra, nos irá cambiando cual es el criterio para poder determinar si una roca es útil para nuestros estudios. Nuestros geólogos jefes quieren usar tres criterios para determinar si una roca es válida.

<b>Criterio Igneas.</b>	<b>Criterio Metamórficas</b>	<b>Criterio Sedimentaria.</b>
Roca de grupo ígneas, Grano muy grueso	Rocas de grupo Metamórfica. Grano medio o fino. Textura Vitrea.	Rocas de grupo sedimentaria. Textura: Fanerítica.

Dependiendo de validador, agmunsen verá una carita feliz en el caso de que la roca cumpla los criterios de validación. En caso contraria recibirá un cara enfadada.



Esto en ambos sistema de entrada de datos.

Además la agencia espacial europea, indica que la NASA nos ha pedido que le mostremos la información de la roca, una vez que sea válida, pero en dos formatos de salida.

<b>Formato Americano</b>	<b>Formato Europeo</b>
Textos en inglés y medias en grados Faren	Textos en español y medidas en grados Celsius

Vamos a continuar con el ejercicio.

Vamos a crear una interface que se denomine IPilotable que haga cumplir las siguientes preguntas.

- DameIdentificador()
  - DameNombreCompleto()
  - DameEdad()
- 
- Vamos a crear una nueva clase Astronauta. Que implemente la interface.
    - identificador
    - nombre completo
    - edad
- 
- Crear una clase denominada Misión, que contendrá un piloto y una misión (la misión es el validador), además de una forma de entrada de datos y otra de salida. Pongamos que son las interface Entrada y la interface Salida, tendrá un método que se denominará Analiza (al que se le tiene que alimentar con una roca) y caso de ser válida la mostrará con el mostrador.

Crear un objeto MisionIgne, que recibirá al astronauta Agmunsen y la misión (Validador Igneas), mandarle una roca que cumpla la misión y otra que no cumpla la misión.

<pre>Interface IPilotable {     dameId(): string;     dameNombre(): string;     dameEdad(): number; }  Class Astronauta implements IPilotable {     Identificador :string;     NombreCompleto: string;     Edad: number; }</pre>	<pre>Class Misión {     IPilotable :Astronauta;     IValidable : Misión;     IEntrada: Entrada;     ISalida: Salida;      Constructor (IPilotable :Astronauta, Misión: IValidable,     IEntrada: Entrada, ISalida :Salida)     {         This.Piloto = Piloto;         This.Mision = Misión;         This.Entrada = Entrada;         This.Salida = Salida;     }      Analiza (MiRoca :Roca) : boolean     {         Let Roca = Entrada.dameRoca();         If (Misión.isValid(MiRoca)         {             Salida.Muestra();         }     } }</pre>
<pre>Let Piloto :IPilotable = new Astronauta("Agmin","Agmunsen Perez",45); Let Mision :IValidable = new ValidadorIgne(); MisionIgne :Misión = new Misión(Piloto,Misión); Console.writelog(MisionIgne.Analiza()); MisionIgne.Mision = new ValidadorMetamorfico(); Console.writelog(MisionIgne.Analiza());</pre>	

