

CLASE 3 - Práctica -

- Construir un componente "Nota" que esté compuesto por un título <h2> y un texto <p>. Este componente recibirá dos propiedades: titulo y texto.
- Pasarle a los elementos html de este componente (<h2> y <p>) estilos de css a partir de una constante que pasaremos dentro del componente como propiedad.
- Utilizar el componente Nota dentro del componente App. Pasar propiedades correspondientes como parámetros.
- Validar las propiedades del componente Nota con PropTypes.

Dado el siguiente array:

```
var notas = [  
  {  
    titulo: "Pizza casera",  
    texto: "300 gr de harina, 160 ml de agua, 5 gr de levadura, 1 cucharadita de sal, 20 gr de aceite, 200 gr de salsa de tomate, 100 gr de mozzarella y orégano."  
  },  
  {  
    titulo: "Comprar en el super",  
    texto: "Vino, picadas y servilletas "  
  },  
  {  
    titulo: "Pelis para ver el finde",  
    texto: "Coco, Pantera negra, The post, Downsizing"  
  }  
]
```

- Construir un componente "ListadoDeNotas" que contenga una lista de componentes tipo Nota que utilice la información del array dado. Esta información debe recibirlo como propiedad.
- Utilizar el componente ListadoDeNotas dentro del componente App.
- Implementar Proptypes en el componente ListadoDeNotas.
Pista: utilizar arrayOf() y shape().

Ejercicio integrador

1. Modificar los componentes para que puedan recibir todo lo que necesiten por parámetro. Agregar PropTypes a todos los componentes hechos hasta el momento.
2. Convertir los botones generados en la ejercicio integrador de la clase 1 en un solo componente "Button.js" que reciba por propiedad:
 - title (texto): el texto a mostrar (obligatorio).
 - icon (nodo): El ícono (opcional con default null).
 - type (enum): el tipo de botón. Si es "primary" le pone la clase "btn-primary" y si es "secondary" le pone la clase "btn-secondary" (opcional con default "primary").
 - active (bool): le pone la clase "active" (opcional default false).
3. Una vez que tenemos el Componente Button.js, reemplazarlo en los lugares donde utilizabamos los otros botones.
4. Modificar GridItem.js para que reciba todos los datos por props. Generar los PropTypes correspondientes a todas las propiedades del componente.
5. Modificar ListItem.js para que reciba todos los datos por props. Generar los PropTypes correspondientes a todas las propiedades del componente.
6. Modificar **ItemsSection** para que reciba:
 - items: películas a mostrar. Esta propiedad recibe un array y es requerida. Utilizar array con shape.
 - type: modo de visualizar esa sección. Esta propiedad recibe un string (enum). La opción "grid" renderiza las películas usando GridItem.js . La opción "list" renderiza las películas utilizando ListItem.js. Utilizar oneOf.

