

Universidad Tecnológica Nacional

Facultad Regional Rosario

Tecnicatura Universitaria de la Programación



Laboratorio de Computación III

Unidad P1.2

Lógica de componentes

Práctica principal

Crear un componente llamado Beers y dentro de él colocar este arreglo de objetos:
(simulando una conexión a una api donde obtendremos estos datos)

```
const beers = [  
  {  
    id: 1,  
    beerName: "American",  
    beerStyle: "IPA",  
    price: 3,  
    available: true,  
  },  
  {  
    id: 2,  
    beerName: "Argenta",  
    beerStyle: "IPA",  
    price: 4,  
    available: false,  
  },  
  {  
    id: 3,  
    beerName: "Irish",  
    beerStyle: "Red",  
    price: 4,  
    available: true,  
  },  
  {  
    id: 4,  
    beerName: "Scotish",  
    beerStyle: "Red",  
    price: 3,  
    available: true,  
  },  
  {  
    id: 5,  
    beerName: "DeEssoCiTratta",  
    beerStyle: "APA",  
    price: 3,  
    available: true,  
  },  
  {  
    id: 6,
```

```

    beerName: "Santa APA",
    beerStyle: "APA",
    price: 3,
    available: true,
  },
  {
    id: 7,
    beerName: "German",
    beerStyle: "Pilsen",
    price: 1,
    available: true,
  },
  {
    id: 8,
    beerName: "London Porter",
    beerStyle: "Porter",
    price: 2,
    available: false,
  },
  {
    id: 9,
    beerName: "Scotish ALE",
    beerStyle: "Red",
    price: 5,
    available: false,
  },
];

```

1 - Crear un componente donde muestre por pantalla una lista de todas las cervezas. Además, se debe calcular el precio en pesos argentinos antes de mostrarlo, ya que el precio que tenemos en el listado está expresado en USD. (Consejo: utilizar alguna array function que nos brinda js)

2 - Crear un componente donde se muestren en pantalla solo las cervezas disponibles. (Mirando la propiedad *available*)

3 - Crear otro componente donde internamente haga un conteo y nos muestre por pantalla la cantidad de cervezas de estilo Red e IPA que tenemos en la lista. (Prestar atención en la manera de pasar el arreglo por props)

4 - Crear un componente que muestre por pantalla solamente los estilos de cerveza que hay en esta lista de cervezas.

Práctica adicional

1. En el componente *App.js* poseo el siguiente arreglo:

```
const products = ["table", "couch", "chair1", "chair2"];
```

Escribir el componente *Products.js* que reciba por *props* ese arreglo y devuelva un componente *Product* por cada uno de los valores del arreglo. A su vez, *Product* manda por la prop *name* el valor de la posición del arreglo (El primer producto recibirá "table" en *name*, el segundo "couch" y así sucesivamente)

Pista: la etiqueta del componente *Product* se vería así:

```
<Product name={/* aca el valor */} /> .
```

2. En el componente *App.js* poseo el siguiente arreglo:

```
const names = ["Parker", "Simmons", "Lewis", "Poe"];
```

Escribir el componente *Persons.js* que reciba por *props* ese arreglo, filtre aquellos nombres que comiencen con la letra **P** y devuelva un componente *Person* por cada una de las personas que cumplen esa condición. A su vez, *Person* recibirá por *props* el valor del nombre.

Pista: la etiqueta del componente *Person* se vería así:

```
<Person name={/* aca el valor */} />
```

3. En el componente *App.js* poseo el siguiente arreglo:

```
const persons = [{ name: 'Juan', age: 26 }, { name: 'Gabriel', age: 27 }, { name: 'Valentina', age: 22 }, { name: 'Paula', age: 25 }, { name: 'Andrés', age: 20 }];
```

Escribir el componente *Family.js* que reciba ese arreglo **previamente ordenado de mayor a menor según su edad** y luego devuelva un componente *FamilyMember.js* que posea una etiqueta *h3* que muestre en pantalla el nombre de la persona y luego una etiqueta *p* con su edad.

Se debe utilizar funciones *map* y *sort* para la resolución de estos ejercicios.

Fecha	Versionado actual	Autor	Observaciones
28/08/2023	1.0.0	Gabriel Golzman / Esteban Ferreyra	Primera versión