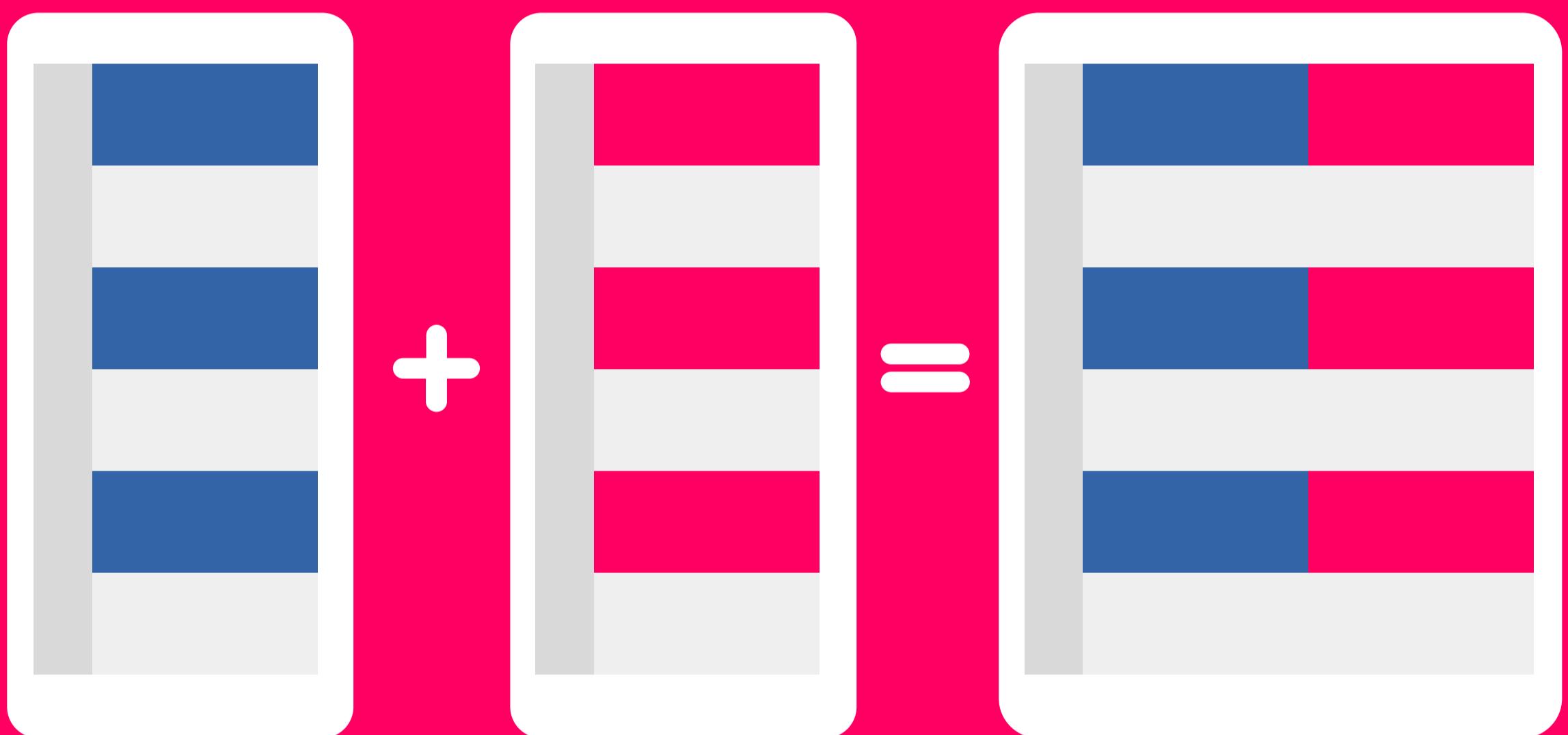




El futuro digital
es de todos

MinTIC



Manejo de Dataframes



Universidad de Caldas

Hola:

En este texto continuaremos hablando de *Dataframes*.

Recordemos que los *Dataframe* son estructuras de datos matriciales que nos permiten manipular información con filas y columnas y tiene ventajas sobre las matrices porque es muy fácil definir los diferentes tipos de datos para las columnas, que se llaman también series y que además tienen una cantidad de funciones predefinidas tanto para el análisis de los datos y para graficarlos.

Este es el nuevo ejemplo para aprender las nuevas funciones que aprenderemos.

```
datos_productos={  
    'category': ['Cleaning', 'Cleaning', 'Entertainment',  
                'Entertainment', 'Tech', 'Tech'],  
    'store': ['Walmart', 'Dia', 'Walmart', 'Fnac', 'Dia', 'Walmart',  
              'price':[11.42, 23.50, 19.99, 15.95, 55.75, 111.55],  
    'testscore': [4, 3, 5, 7, 5, 8]}  
  
dfP = pd.DataFrame(datos_productos)  
dfP
```

Aquí estamos definiendo un nuevo diccionario que se llama *productos* y que tiene cuatro (4) valores: categoría, donde se almacena, precio y resultado.

Si creamos un *Dataframe* quedaría así:

```
datos_productos={  
    'category': ['Cleaning', 'Cleaning', 'Entertainment',  
                'Entertainment', 'Tech', 'Tech'],  
    'store': ['Walmart', 'Dia', 'Walmart', 'Fnac', 'Dia', 'Walmart'],  
    'price':[11.42, 23.50, 19.99, 15.95, 55.75, 111.55],  
    'testscore': [4, 3, 5, 7, 5, 8]}  
  
dfP = pd.DataFrame(datos_productos)  
dfP
```

	category	store	price	testscore
0	Cleaning	Walmart	11.42	4
1	Cleaning	Dia	23.50	3
2	Entertainment	Walmart	19.99	5
3	Entertainment	Fnac	15.95	7
4	Tech	Dia	55.75	5
5	Tech	Walmart	111.55	8

Se parece mucho a una hoja electrónica en donde tenemos las diferentes columnas y las filas que me dan cada producto.

Veamos las funciones que podemos tener allí:

.sort: con esta función se ordena el *Dataframe* con respecto al precio de menor a mayor

```
dfP.sort_values("price")
```

	category	store	price	testscore
0	Cleaning	Walmart	11.42	4
3	Entertainment	Fnac	15.95	7
2	Entertainment	Walmart	19.99	5
1	Cleaning	Dia	23.50	3
4	Tech	Dia	55.75	5
5	Tech	Walmart	111.55	8

Sin embargo, la ubicación de estos elementos cambia al ordenarlo. Esto es algo que enfatizaremos en el laboratorio.

.iloc: con esta función podemos ver una parte del *Dataframe*.

```
dfP.iloc[1:4]
```

	category	store	price	testscore
1	Cleaning	Dia	23.50	3
2	Entertainment	Walmart	19.99	5
3	Entertainment	Fnac	15.95	7

dfP: Podemos solicitar la selección de filas que se encuentran allí almacenadas. Si tenemos miles de productos y necesitamos solo los de entretenimiento, lo podemos hacer con esta función.

```
dfP["category"]=="Entertainment"
```

```
0    False
1    False
2     True
3     True
4    False
5    False
Name: category, dtype: bool
```

Podemos crear un nuevo *Dataframe* donde la condición sea verdadera

```
dfP[dfP["category"]=="Entertainment"]
```

	category	store	price	testscore
2	Entertainment	Walmart	19.99	5
3	Entertainment	Fnac	15.95	7

Puedo hacer funciones más interesantes, por ejemplo ¿Cuáles productos tienen un precio mayor a 20 dólares?

```
dfP[dfP["price"]>=20]
```

	category	store	price	testscore
1	Cleaning	Dia	23.50	3
4	Tech	Dia	55.75	5
5	Tech	Walmart	111.55	8

Todas estas funciones hacen que los *Dataframe* sean de mucha utilidad en el manejo de Python.

En conclusión, vimos algunas funciones del *Dataframe* entre ellas el ordenamiento y el selector para una condición.



Universidad de Caldas