

## Desafío - Hashes

- Para realizar este desafío debes haber estudiado previamente todo el material disponible correspondiente a la unidad.
- Una vez terminado el desafío, comprime la carpeta que contiene el desarrollo de los requerimientos solicitados y sube el .zip en el LMS.
- Puntaje total: **10 puntos**
- Desarrollo desafío:
  - El desafío se debe desarrollar de manera Individual.

### Introducciones

A continuación se detallan variados desafíos a desarrollar. Para su correcta evaluación, los programas deben ser almacenados en un archivo comprimido .zip de la siguiente manera:

```
desafios.zip
├── busqueda.rb
├── iter1.rb
├── iter2.rb
├── iter3.rb
└── trimestres.rb
```

Considerar este hash para todos los siguientes desafíos:

```
ventas = {
  Enero: 15000,
  Febrero: 22000,
  Marzo: 12000,
  Abril: 17000,
  Mayo: 81000,
  Junio: 13000,
  Julio: 21000,
  Agosto: 41200,
  Septiembre: 25000,
  Octubre: 21500,
  Noviembre: 91000,
  Diciembre: 21000
}
```

## iter1.rb

Crear el programa `iter1.rb` que itere el hash `ventas` y muestre en pantalla todas las ventas superiores a 45000 (sólo el valor de la venta).

Uso:

```
ruby iter1.rb
```

81000

91000

Se evaluará el output en pantalla.

El hash utilizado para evaluar puede ser distinto y tener más o menos meses.

## iter2.rb

Crear el programa `iter2.rb` que itere el hash `ventas` y muestre en pantalla todos los meses cuyas ventas sean superiores a 45000.

Uso:

```
ruby iter1.rb
```

Mayo

Noviembre



### Tips:

- Respetar el nombre del programa.
- Se evaluará el output en pantalla.
- El hash utilizado para evaluar puede ser distinto y tener más o menos meses.

## iter3.rb

Crear un programa llamado `iter3.rb` que tenga un método llamado `filter` que reciba un hash y devuelva un hash nuevo con los valores superior a un parámetro que será ingresado al momento de llamar al programa.

Uso:

```
filter 45000
```

No hay output, se evaluará el retorno del método



### Tips:

- Respetar el nombre del programa.
- Respetar el nombre del método.
- Se evaluará el llamado al método
- El hash utilizado para evaluar puede ser distinto y tener más o menos meses.

## busqueda.rb

Crear un programa llamado `busqueda.rb` que pueda buscar a que mes pertenece una o más cifras específicas. En caso de no encontrarlo mostrar el mensaje "no encontrado".

Uso:

```
ruby busqueda.rb 15000 31000 27000
```

```
Enero  
no encontrado  
no encontrado
```

Se asumirá que en ningún mes la venta se repite.



### Tips:

- Se evalúa la salida en pantalla por lo que se debe respetar los mensajes.
- Se evaluará con un hash distinto al mostrado.

## trimestres.rb

Crear el programa trimestres.rb donde se pide generar un hash llamado quarters con las ventas de cada trimestre a partir del hash de ventas de los enunciados anteriores. Las claves del hash tienen que ser 'Q1', 'Q2', 'Q3', 'Q4'.

Uso:

```
ruby trimestres.rb
```

```
{"Q1"=>49000, "Q2"=>111000, "Q3"=>87200, "Q4"=>133500}
```



### Tips:

- Los valores ingresados serán distintos.
- Es un ejercicio de arrays o de hashes. ¿Se necesitan los keys?
- Revisar la documentación del método `.each_slice`.
- Necesitamos iterar elementos, o elementos con índices.