

Desafío resuelto - Adicto a redes

Se tiene un arreglo con la cantidad de minutos usados en redes sociales de distintos usuarios. Se pide crear el programa `adictos_a_redes.rb` con un método llamado `scan_addicts` que reciba un arreglo con los minutos de uso y como resultado entregue un nuevo arreglo cambiando todas las medidas inferiores a 90 minutos como 'bien' y todas las mayores o iguales a 90 como 'mal'.

```
scan_addicts([120, 50, 600, 30, 90, 10, 200, 0, 500])  
# => ["mal", "bien", "mal", "bien", "bien", "bien", "mal", "bien", "mal"]
```

Solución al ejercicio de arreglo

Se tiene un arreglo con la cantidad de minutos usados en redes sociales de distintos usuarios. Se pide crear un método llamado `scan_addicts` que reciba un arreglo con los minutos de uso y como resultado entregue un nuevo arreglo cambiando todas las medidas inferiores a 90 minutos como 'bien' y todas las mayores o iguales a 90 como 'mal'.

```
scan_addicts([120, 50, 600, 30, 90, 10, 200, 0, 500])  
# => ["mal", "bien", "mal", "bien", "bien", "bien", "mal", "bien", "mal"]
```

Al resolver problemas de este tipo hay dos acercamientos igualmente válidos, podemos partir por la iteración del arreglo o podemos partir por el método. Hacerlo desde el método es mejor porque una vez que tengamos listo el ejercicio no tenemos que reescribirlo.

Creando el método

Para crear el método tenemos que saber lo que recibe y lo que devuelve. En este ejercicio recibe un arreglo y devuelve un arreglo nuevo.

```
def scan_addicts(array)
  results = []
  results # Lo que devuelve.
end
```

Iterando sobre el arreglo

```
def scan_addicts(array)
  results = []
  n = array.count
  n.times do |i|
    # Aquí es donde va la lógica de la iteración
  end
  results
end
```

Agregamos la lógica de la transformación, si es mayor que 90 guardamos 'mal' en el arreglo nuevo y si es menor o igual que 90, guardamos 'bien'. Para guardar ocuparemos el método `.push`

```
def scan_addicts(array)
  results = []
  n = array.count
  n.times do |i|
    if array[i] > 90
      results.push 'mal'
    else
      results.push 'bien'
    end
  end
  results # cuidado con el retorno, times retorna la cuenta
end
```