

En shell script, los arrays son estructuras de datos que almacenan una colección de elementos, donde cada elemento es accesible mediante un índice numérico. Aquí tienes una descripción de las operaciones básicas que se pueden realizar con arrays en shell script:

1. Crear un array: Puedes crear un array usando la sintaxis `nombre=("elemento1" "elemento2" "elemento3")`. Por ejemplo:

```
nombres=("Alice" "Bob" "Charlie")
```

2. Acceder a un elemento: Para acceder a un elemento del array, usa la sintaxis `${nombre[indice]}`. Por ejemplo:

```
echo "${nombres[1]}" # Imprime "Bob"
```

3. Longitud del array: Para obtener el número de elementos en un array, usa la sintaxis `${#nombre[@]}`. Por ejemplo:

```
echo "${#nombres[@]}" # Imprime "3"
```

4. Añadir un elemento: Para agregar un elemento al final de un array, usa la sintaxis `nombre+=("elemento")`. Por ejemplo:

```
nombres+=("David")
```

5. Eliminar un elemento: Para eliminar un elemento de un array, usa la sintaxis `unset nombre[indice]`. Por ejemplo:

```
unset nombres[1] # Elimina "Bob"
```

6. Iterar sobre los elementos de un array: Puedes usar un bucle `for` para recorrer los elementos de un array. Por ejemplo:

```
for nombre in "${nombres[@]"; do
  echo "$nombre"
done
```

7. Acceder a un rango de elementos: Puedes acceder a un subconjunto de elementos de un array usando la sintaxis `${nombre[@]:inicio:longitud}`. Por ejemplo:

```
echo "${nombres[@]:1:2}" # Imprime "Bob Charlie"
```

8. Encontrar el índice de un elemento: Puedes usar un bucle for para buscar el índice de un elemento específico en un array. Por ejemplo:

```
buscar="Charlie"
indice=-1
for ((i=0; i<${#nombres[@]}; i++)); do
  if [[ "${nombres[i]}" == "$buscar" ]]; then
    indice=$i
    break
  fi
done
```

9. Ordenar un array: Para ordenar un array alfabéticamente, puedes utilizar el comando sort. Por ejemplo:

```
#!/bin/bash
# Lista no ordenada de nombres
nombres=("Alice" "Eve" "Charlie" "David" "Bob")

# Ordenar la lista de nombres alfabéticamente usando el comando 'sort'
IFS=$'\n' nombres_ordenados=$(sort <<<"${nombres[*]}")
unset IFS

# Imprimir la lista de nombres ordenados
echo "Nombres ordenados:"
for nombre in "${nombres_ordenados[@]"; do
  echo "$nombre"
done
```

Este script crea un array no ordenado de nombres, los ordena alfabéticamente utilizando el comando sort, y luego imprime los nombres ordenados en la terminal. El comando sort toma el contenido de la lista de nombres y los ordena línea por línea. La variable IFS se modifica temporalmente para asegurar que sort utilice el salto de línea como separador de elementos en lugar del espacio en blanco.

10. Filtrar elementos de un array: Puedes usar un bucle for y una condición if para filtrar elementos de un array según un criterio dado. Por ejemplo:

```
nombres_filtrados=()
for nombre in "${nombres[@]"; do
  if [[ $nombre != "Bob" ]]; then
    nombres_filtrados+=("$nombre")
  fi
done
```

Estas son algunas de las operaciones básicas que se pueden realizar con arrays en shell script. Puedes combinar estas operaciones para realizar tareas más complejas según tus necesidades.