

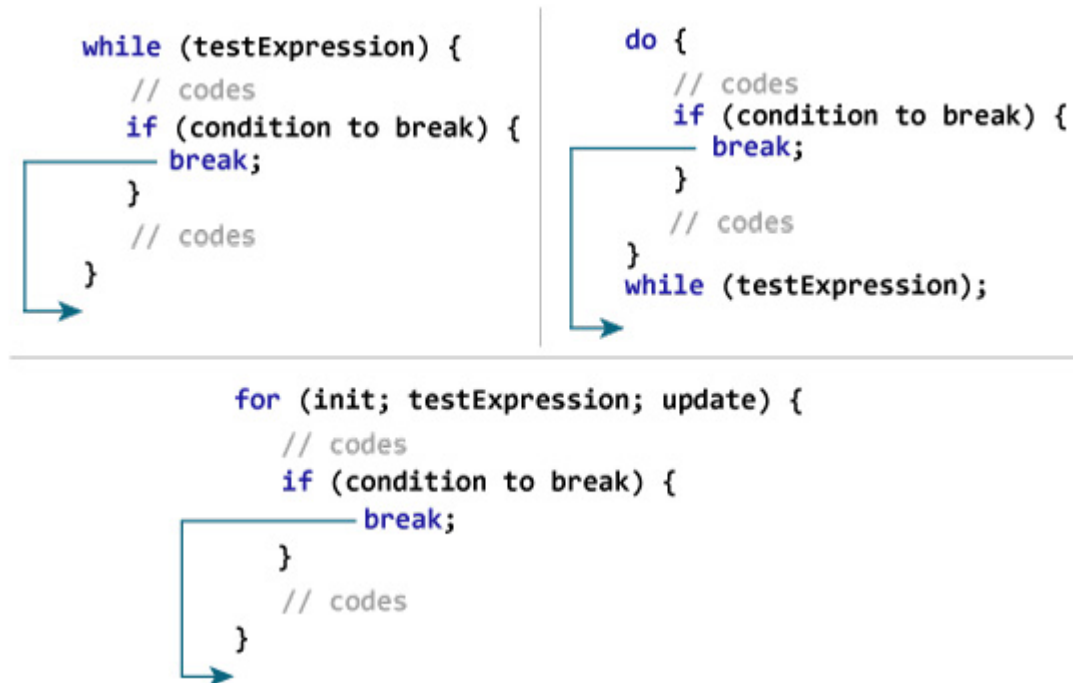
Respuesta a: Ciclo for y foreach

El break, continue fue un tema que me costo entender pero anexo mi informacion en dado caso que se necesiten mas ejemplos para poder entender este tipo de sentencias.

Break

La sentencia "break" termina el bucle inmediatamente, y el control del programa se mueve a la siguiente sentencia después del bucle.

***Normalmente se usa con declaraciones de toma de decisión (If, if else, while, do while, for)**



Ejemplo de un break

//El programa aumentara la variable i hasta que llegue a 5 ya que estamos cortar

```
for (int i = 1; i <= 10; ++i) {
    if (i == 5) {
        break;
    }
    System.out.println(i);
}
```

Output

```
1
2
3
4
```

Lo que me llegó a confundir con esta sentencia era el hecho de en que se podía usar en un caso mas elaborado, el siguiente programa suma todas las variables por el usuario hasta que pone un numero negativo. Podria mejorarse este programa que al insertar un cierto numero, el programa deje de correr y de el resultado.

Ejemplo 2

```
int numero, suma = 0;
Scanner entrada = new Scanner(System.in);
```

```

while (true) {
    System.out.print("Dame un numero: ");
    numero = entrada.nextInt();

    if (numero < 0) {
        break;
    }

    suma += numero;
}
System.out.println("Suma = " + suma);
}

```

Output

```

Dame un numero:1
Dame un numero:2
Dame un numero:3
Dame un numero:4
Dame un numero:-2
Suma = 10

```

Aquí, el bucle while se ejecuta hasta que el usuario introduce un número negativo. Si el usuario introduce un número negativo, rompe la sentencia if y termina el bucle while.

Fuente: [Programiz \(Post en ingles\)](#)

Continue

La sentencia "continue" salta la iteración actual de un bucle.

Cuando se ejecuta la sentencia continue, el control del programa salta al final del bucle.

***Normalmente se usa con declaraciones de toma de decisión (if, if else, while, do while, for)**



Ejemplo de Continue

```

for (int i = 1; i <= 10; ++i) {
    if (i > 4 && i < 9) {
        continue;
    }
    System.out.println(i);
}

```

Output

1
2
3
4
9
10

Cuando el valor de i se convierte en más de 4 y menos de 9, se ejecuta la sentencia continue, lo que omite la ejecución de `System.out.println(i);`

Ejemplo 2

```
for (int i = 1; i < 6; ++i) {  
    System.out.print("Dame un numero: ");  
    numero = entrada.nextInt();  
  
    if (numero <= 0) {  
        continue;  
    }  
  
    suma += numero;  
}  
System.out.println("Suma = " + suma);
```

Output

Dame un numero:1
Dame un numero:0
Dame un numero:-1
Dame un numero:4
Dame un numero:7
Suma = 12

De igual forma, un programa mas complejo seria al igual que en el caso del ejemplo de break, pero ahora estableciendo un maximo de 5 numeros a sumar, si el usuario pone un numero negativo o igual a 0, el programa lo saltara.