

jueves, 12 de octubre de 2023

# DevOps Engineer Code Challenge

---

## Challenges

### - Challenge 1

(Explicaciones tanto aquí como en código)

Estas son las herramientas que he utilizado para conseguir los tres objetivos que se plantean:

- Aísla grupos de nodos específicos (**nodeSelector**).
- Asegura que un pod del mismo tipo no se programe en el mismo nodo (**podAntiAffinity**).
- Despliega los pods en diferentes zonas de disponibilidad (**nodeAffinity**).

### - Challenge 2

(Explicaciones tanto aquí como en código)

Planteamiento del ejercicio, hacer dos declaraciones, una se copian los archivos del 'reference' a la 'instance' y otro que instale del reference al cluster AKS.

1. Realizamos estas dos declaraciones en el main.
2. Establecemos las variables, han de estar relacionadas con las del módulo.
3. Ejecutamos los comandos de validación 'terraform validate' y 'terraform fm'
4. Recibimos el siguiente mensaje: 'Success! The configuration is valid'

### - Challenge 3

(Explicaciones tanto aquí como en código)

En este challenge se nos pide que instalemos el chart de helm creado en el challenge 1 usando el módulo del challenge 2, esto va a hacer que tengamos que modificar el código de terraform puesto que los valores para configurar la identificación los obtenemos de terraform mediante entornos.

Dentro de GitHub habra que crear los secretos, para ello hacemos lo siguiente:

1. Vamos a la página del repositorio de GitHub en el que se esta trabajando
2. Configuración
3. Secretos
4. Nuevo secreto del repositorio
5. Agregar secreto

Por último, en cuanto al progreso del GitHub workflow, cuando estamos en la rama main, recoge los secretos de identificación, aplica la configuración de terraform y configura kubectl.