



Helvética N ...



Paso 8 de 8



IBM Developer
SKILLS NETWORK

Laboratorio práctico: programación de trabajos con crontab

Tiempo estimado necesario: **20** minutos

Objetivos

Después de completar esta práctica de laboratorio, podrá:

- Enumere los trabajos cron existentes.
- Agregue un trabajo cron.
- Eliminar trabajos cron.

Acerca de Skills Network Cloud IDE

Skills Network Cloud IDE (basado en Theia y Docker) proporciona un entorno para laboratorios prácticos para laboratorios relacionados con cursos y proyectos. Theia es un IDE (entorno de desarrollo integrado) de código abierto, que se puede ejecutar en el escritorio o en la nube. Para completar este laboratorio, usaremos el IDE de la nube basado en Theia ejecutándose en un contenedor Docker.

Aviso importante sobre este entorno de laboratorio

Tenga en cuenta que las sesiones para este entorno de laboratorio no se conservan. Cada vez que se conecta a este laboratorio, se crea un nuevo entorno para usted. Todos los datos que haya guardado en la sesión anterior se perderán. Planifique completar estas prácticas de laboratorio en una sola sesión para evitar perder sus datos.

Ejercicio 1: comprender la sintaxis del archivo crontab

Cron es un demonio del sistema que se utiliza para ejecutar las tareas deseadas en segundo plano en momentos designados.

Un archivo crontab es un archivo de texto simple que contiene una lista de comandos destinados a ejecutarse en momentos específicos. Se edita con el comando **crontab**.

Cada línea en el archivo crontab tiene cinco campos de fecha y hora, seguidos de un comando, seguido de un carácter de nueva línea ('\n'). Los campos están separados por espacios.

Los cinco campos de fecha y hora no pueden contener espacios. Los cinco campos de fecha y hora y sus valores permitidos son los siguientes:

minuto (0-59)

hora (0-23, 0 = medianoche)

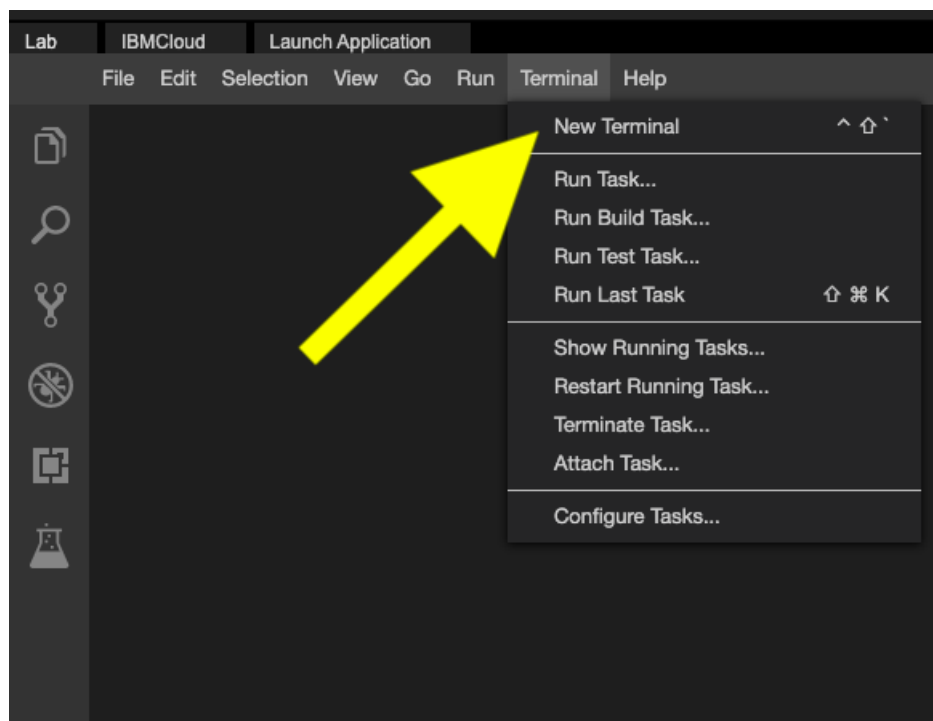
día (1-31)

mes (1-12)

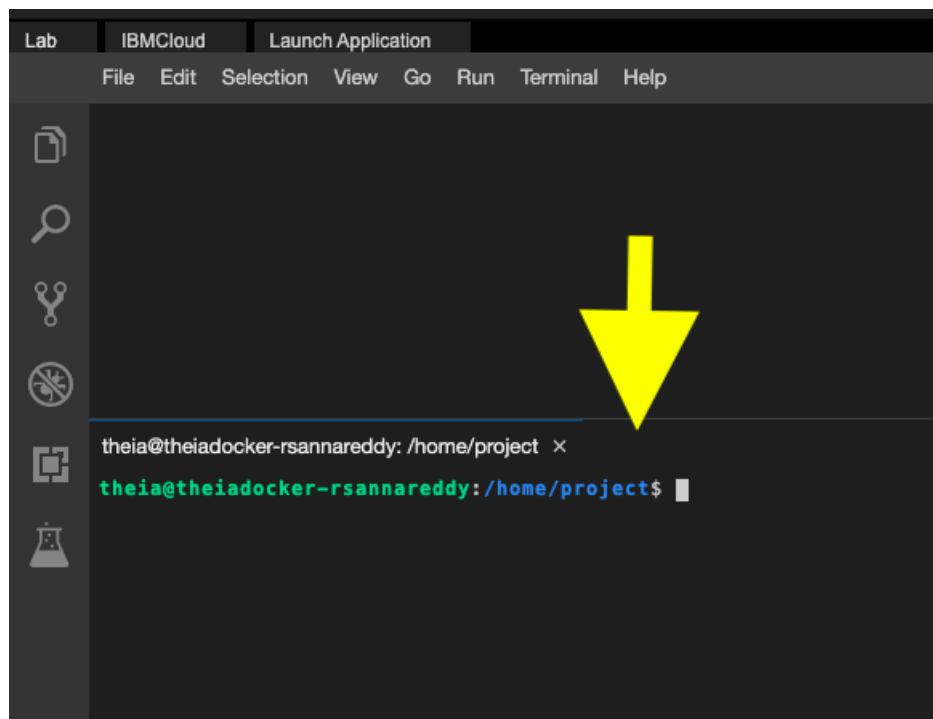
día de la semana (0-6, 0 = domingo)

Ejercicio 2: enumerar trabajos cron

Abra una nueva terminal, haciendo clic en la barra de menú y seleccionando **Terminal** -> **Nueva terminal**, como se muestra en la imagen de abajo.



Esto abrirá una nueva terminal en la parte inferior de la pantalla como se muestra en la imagen de abajo.



Ejecute los siguientes comandos en la terminal recién abierta.

La opción `-l` del comando `crontab` imprime el `crontab` actual.

```
crontab -l
```

Puede recibir un mensaje no `crontab` for `theia` si su `crontab` está vacío.

Ejercicio 3: agregar un trabajo en el archivo `crontab`

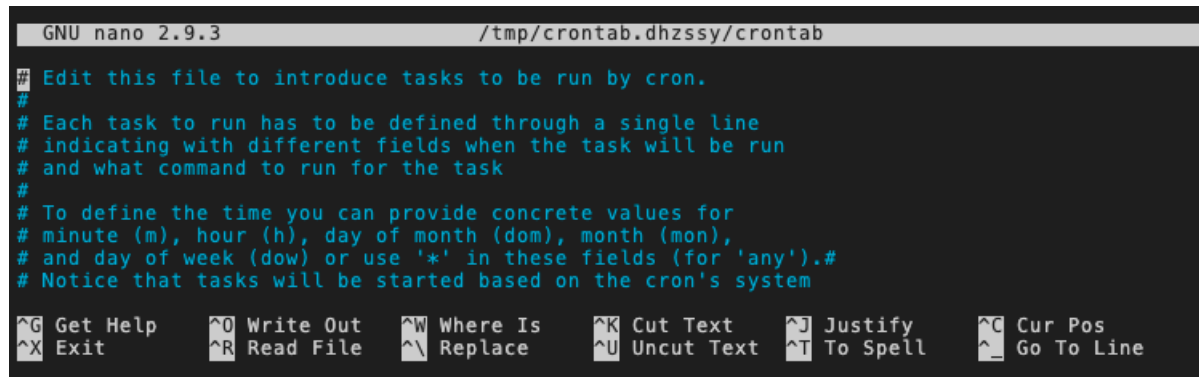
3.1. Agregue un trabajo a `crontab`.

Para agregar un trabajo cron, ejecute el siguiente comando

```
crontab -e
```

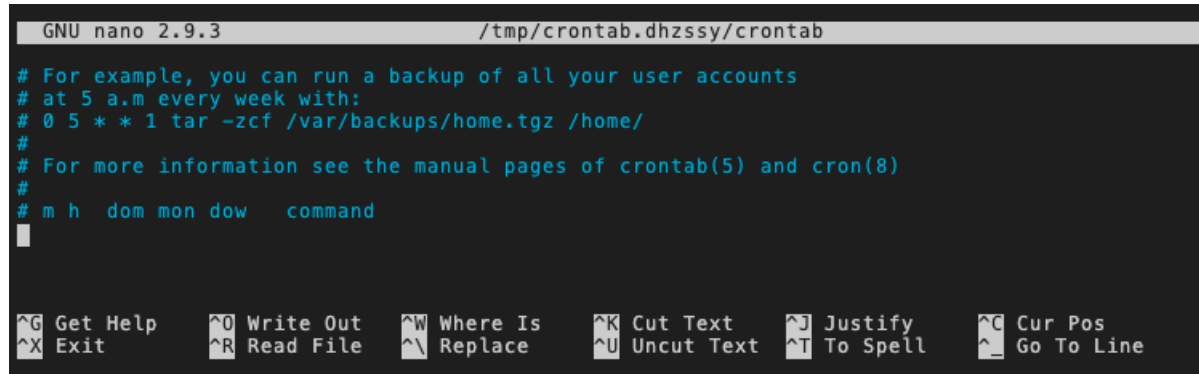
Esto creará un nuevo archivo `crontab` para usted (si aún no tiene uno) y ahora está listo para agregar un nuevo trabajo cron.

Su archivo `crontab` se abrirá en un editor como se muestra en la imagen a continuación.



```
GNU nano 2.9.3 /tmp/crontab.dhzssy/crontab
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').#
# Notice that tasks will be started based on the cron's system
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h  dom mon dow   command
#
```

Desplácese hacia abajo hasta el final del archivo con las teclas de flecha.



```
GNU nano 2.9.3 /tmp/crontab.dhzssy/crontab
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h  dom mon dow   command
#
```

Agregue la siguiente línea al final del archivo `crontab`.

```
0 21 * * * echo "Welcome to cron" >> /tmp/echo.txt
```

```

GNU nano 2.9.3 /tmp/crontab.sqvoQ6/crontab

# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command
0 21 * * * echo "Welcome to cron" >> /tmp/echo.txt

^G Get Help      ^O Write Out    ^W Where Is     ^K Cut Text     ^J Justify     ^C Cur Pos
^X Exit          ^R Read File    ^\ Replace      ^U Uncut Text   ^T To Spell    ^_ Go To Line

```

El trabajo anterior especifica que el comando echo debe ejecutarse cuando el minuto es 0 y las horas son 21. En efecto, significa que el trabajo se ejecuta a las 9:00 pm todos los días.

La salida del comando debe enviarse a un archivo /tmp/echo.txt.

Presione Control + X para guardar los cambios.

Presione 'Y' para confirmar.

```

GNU nano 2.9.3 /tmp/crontab.sqvoQ6/crontab

# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command
0 21 * * * echo "Welcome to cron" >> /tmp/echo.txt

Save modified buffer? (Answering "No" will DISCARD changes.)
Y Yes
N No      ^C Cancel

```

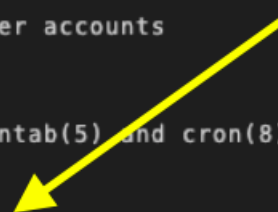
Presione Enter para salir del editor.

Verifique si el trabajo se agregó al crontab ejecutando el siguiente comando.

```
crontab -l
```

Debería ver el trabajo recién agregado en la salida.

```
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command
0 21 * * * echo "Welcome to cron" >> /tmp/echo.txt
theia@theiadocker-rsannareddy:/home/project$
```



3.2. Programe un script de shell .

Creemos un script de shell simple que imprima la hora actual y las estadísticas actuales de uso del disco.

Paso 1: en el menú de la pantalla del laboratorio, use **Archivo-> Nuevo archivo** para crear un nuevo archivo:

Paso 2: ingrese el nombre del archivo como 'diskusage.sh' y haga clic en 'Aceptar'

Paso 3: guarde los siguientes comandos en el script de shell.

```
#!/bin/bash
# print the current date time
date
# print the disk free statistics
df -h
```

Paso 4: Guarde el archivo usando la opción de menú **Archivo-> Guardar** .

Paso 5: Verifique que la secuencia de comandos esté funcionando:

```
chmod u+x diskusage.sh
./diskusage.sh
```

El script debe imprimir la marca de tiempo actual y las estadísticas de uso del disco.

Programemos este script para que se ejecute todos los días a la medianoche a las 12:00 (cuando la hora es 0 en el reloj de 24 horas). Queremos que se agregue la salida de este script /home/project/diskusage.log.

Edite el crontab.

```
crontab -e
```

Agregue la siguiente línea al final del archivo:

```
0 0 * * * /home/project/disksusage.sh >>/home/project/diskusage.log
```

Presione Control + X para guardar los cambios.

Presione 'Y' para confirmar.

Presione Enter para salir del editor.

Verifique si el trabajo se agregó al crontab ejecutando el siguiente comando.

```
crontab -l
```

Debería ver el trabajo recién agregado en la salida.

Ejercicio 4: eliminar el crontab actual

La opción -r hace que se elimine el crontab actual.

Precaución: esto elimina todos sus trabajos cron. Tenga mucho cuidado cuando utilice este comando en un servidor de producción.

```
crontab -r
```

Verifique si su crontab está eliminado.

```
crontab -l
```

Ejercicios de práctica

1. Problema:

Cree un trabajo cron que ejecute la tarea `date >> /tmp/everymin.txt` cada minuto.

- Haga clic aquí para obtener una pista
- Haga clic aquí para SOLUCION

Autores

Ramesh Sannareddy

Otros colaboradores

Rav Ahuja

Cambio de registro

Fecha (AAAA-MM-DD)	Versión	Cambiado por	Cambiar Descripción
2021-05-30	0,1	Ramesh Sannareddy	Versión inicial creada del laboratorio

Copyright (c) 2021 IBM Corporation. Reservados todos los derechos.

Anterior