**GUIA CDO**

## **CYGWIN**

Abrir el programa de ***“cygwin”***



Ilustración 1 cygwin

Como se puede apreciar en la ilustracion1 esa es la pantalla principal del programa, maneja un entorno a una terminal tipo CMD. El programa trabaja desde la ubicación donde se encuentra instalado en este caso se encuentra en la unidad C.

Para empezar a trabajar en un nuevo proyecto de nuestro interés nos moveremos de esa ruta, con los siguientes comandos.

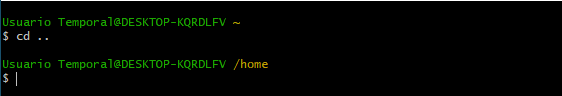


Ilustración 2 comandos

Una vez que en la parte superior aparezca ***/home*** significa que estamos en la carpeta principal, podemos trabajar directamente ahí o en un caso de que nos guste trabajar ordenadamente generamos una carpeta por proyecto.

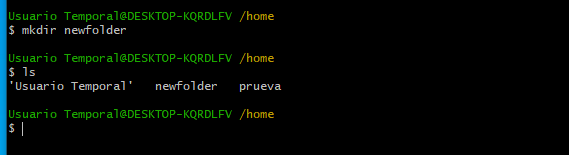


Ilustración 3 comandos

Estos comandos son básicos en el manejo del sistema operativo Linux. Una explicación rápida ***“mkdir”*** creara una carpeta en la ruta que tiene arriba, con el nombre de nuestro gusto y el comando ***“ls”*** es para tener una vista panorámica de todos los archivos existentes en esa ruta. Como se puede apreciar nuestra nueva carpeta.

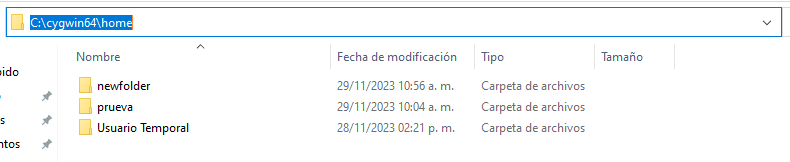


Ilustración 4 Disco local

En la ilustración 4 es un ejempló de cómo podemos visualizar donde estamos trabajando.

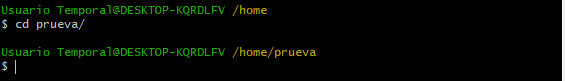


Ilustración 5 comando acceder a carpeta

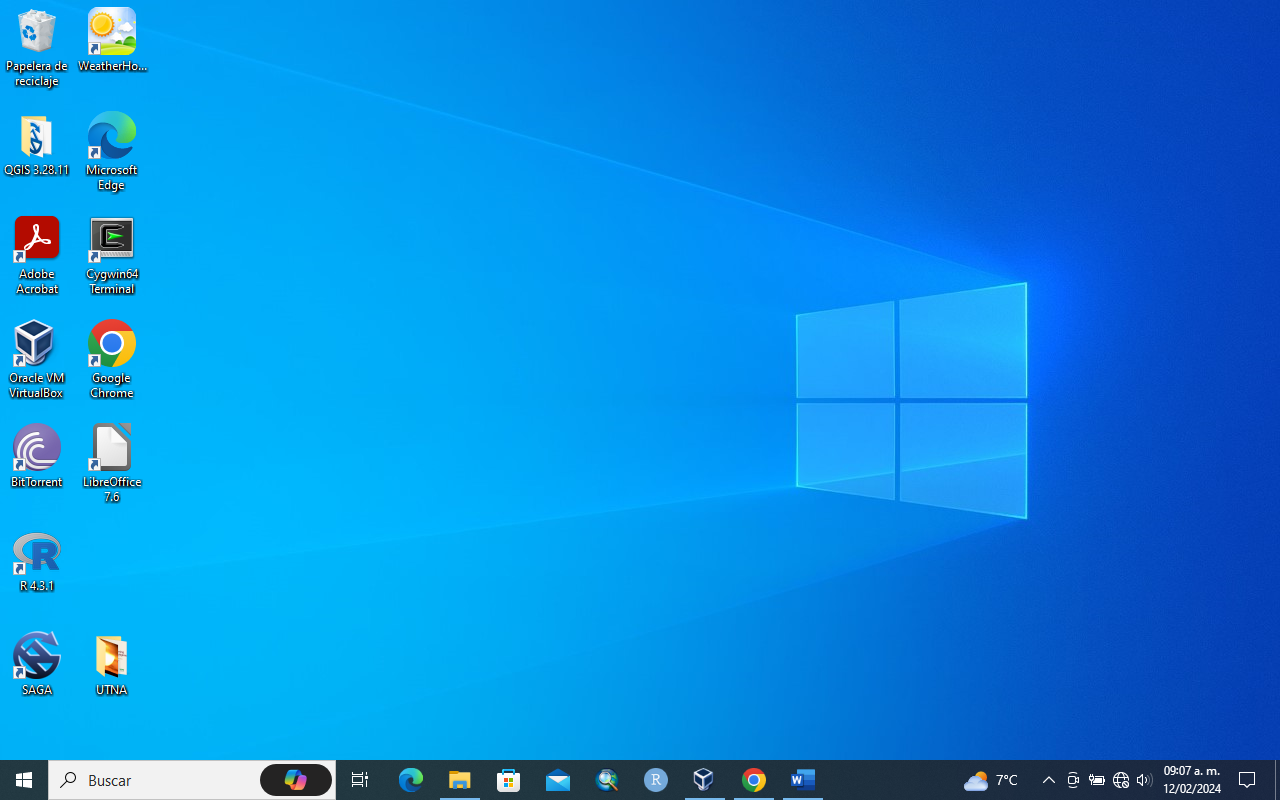
Para acceder a la carpeta que trabajaremos se pone el siguiente comando como el de la ilustración 5



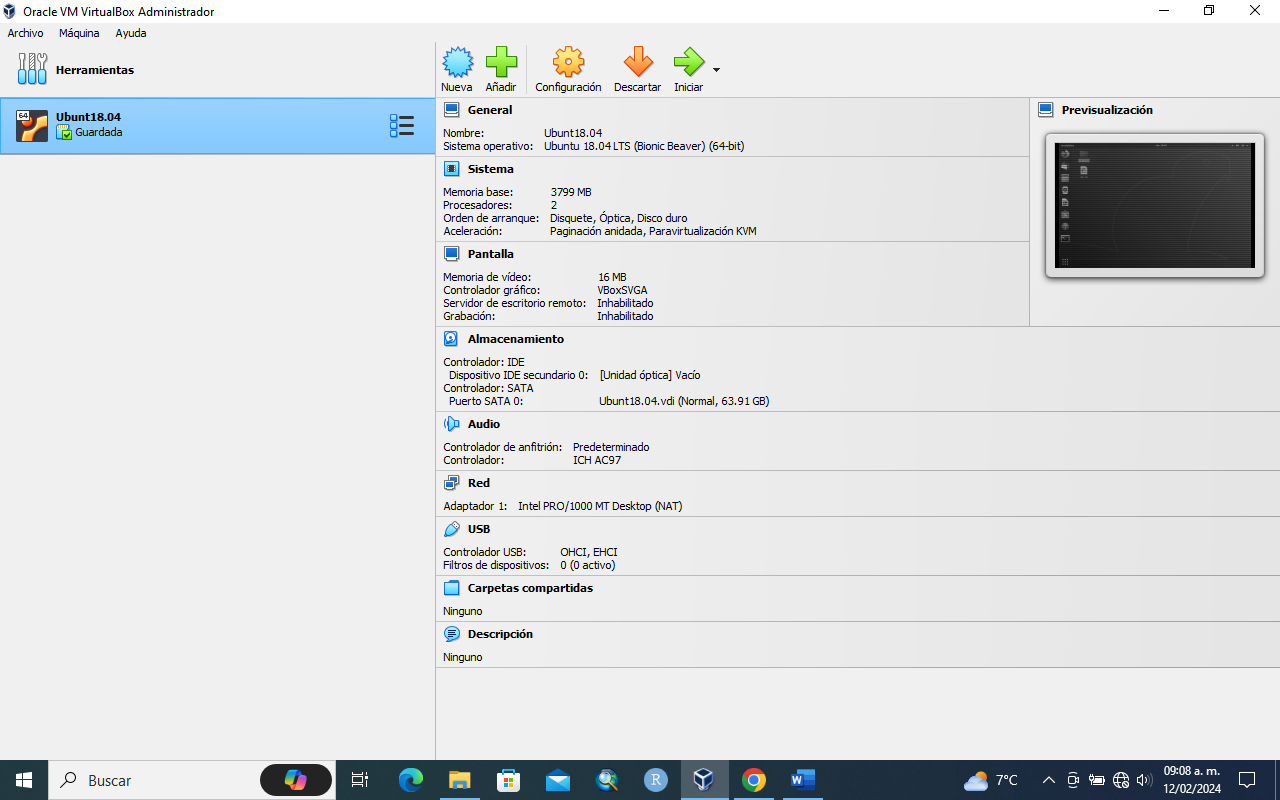
La estructura para trabajar en la terminal es la siguiente como en la ilustración 6. Esta consta de siempre iniciar con ***“cdo”*** después esta información depende a lo que ejecutemos o deseemos realizar, esta información se puede ayudar gracias otro documento ***cdo\_manual*** en este detalla más la información de sus correctas formas de trabajar y por último el archivo con el que trabajaremos.

## **UBUNTU**

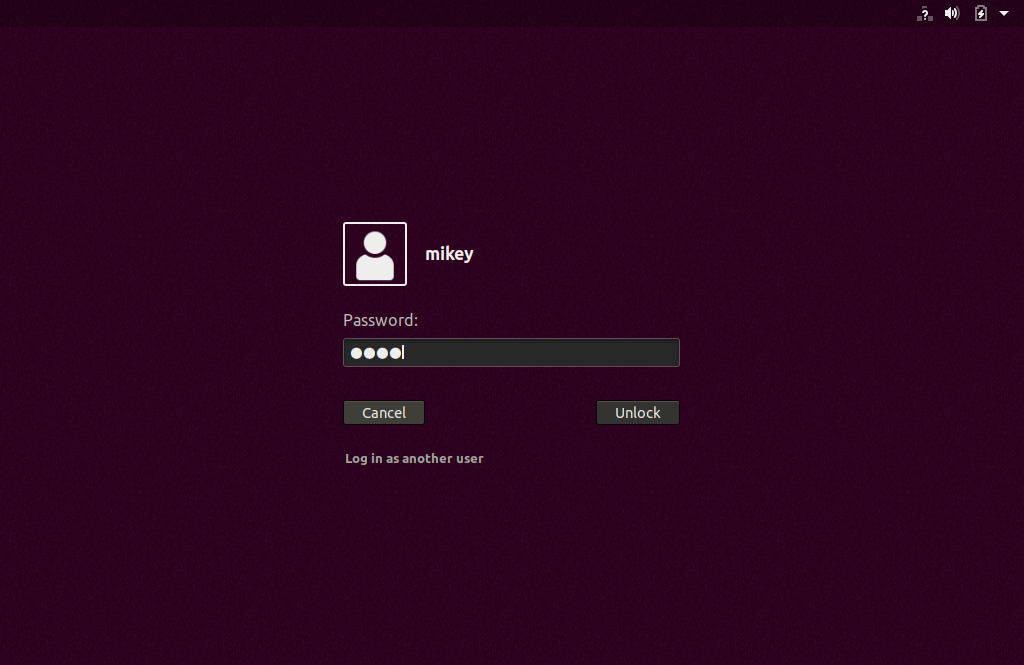
Iniciamos ejecutando Virtual Box, el programa debe de estar instalado previamente y con su ISO de software que trabajaremos “en este cado trabajamos una versión de Ubuntu 18.04 Lts”.



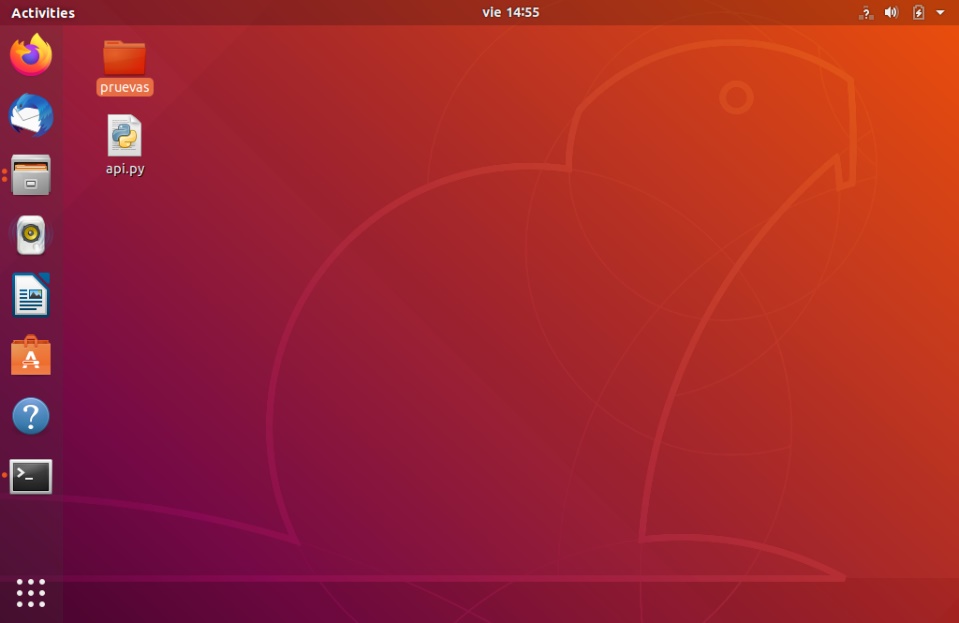
Una vez iniciado el programa, mostrara una vista como en la siguiente figura lo muestra. Para iniciar nuestro sistema en la maquina virtual pulamos doble clic en la parte izquierda o en otro caso, en la parte derecha en la flecha que dice “Iniciar”



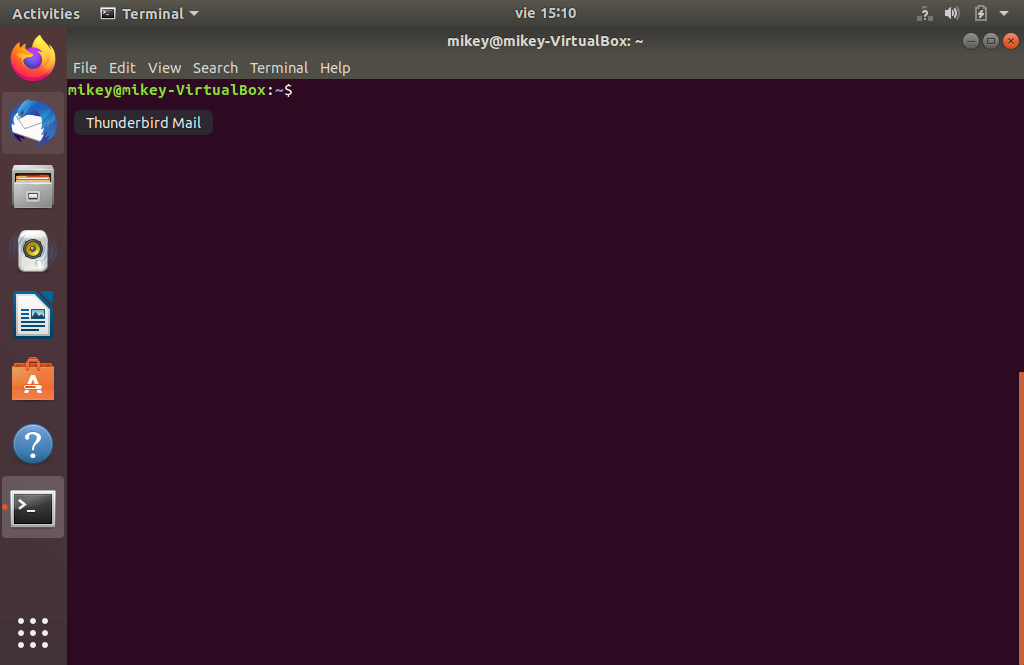
Una vez iniciado la máquina virtual, se ejecutará el sistema operativo previamente instalado caso contario de no haberlo hecho realizar la instalación correspondiendo del sistema operativo a utilizar. Continuando mostrara la vista de login del sistema operativo, escribimos la contraseña…



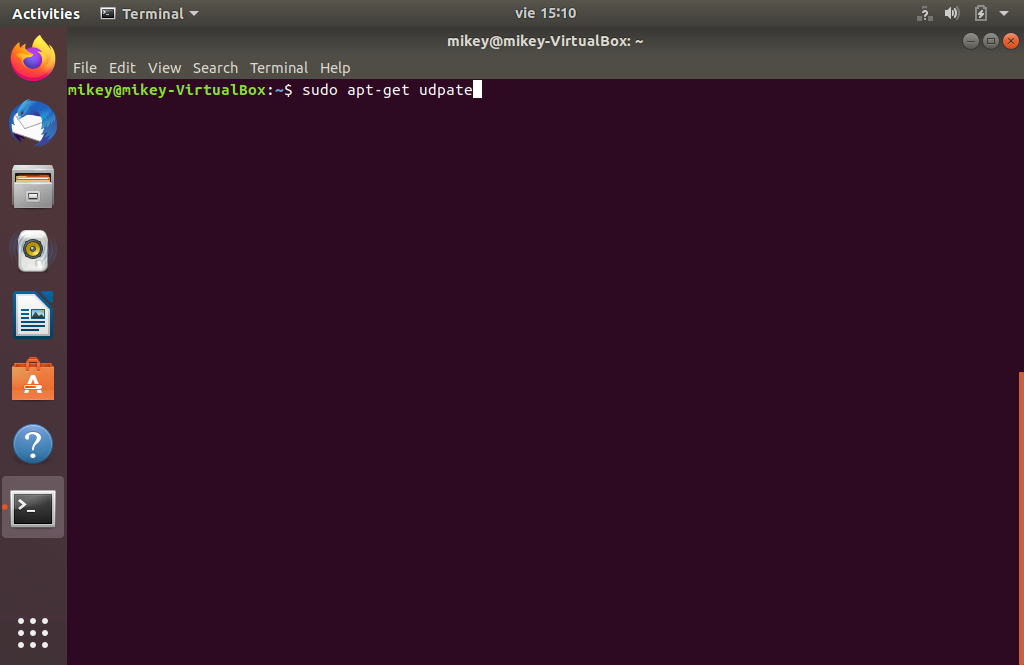
Una vez terminado la espera del inicio de sesión, mostrara la vista principal del sistema operativo, en este caso trabajamos con Ubuntu 18.04 Lts… nota: la configuración y todo eso también se mira en la parte de la instalación, en este caso solo abarcaremos la instalación de programas que usaremos nosotros las cuales es Netcdf CDO & Ncview



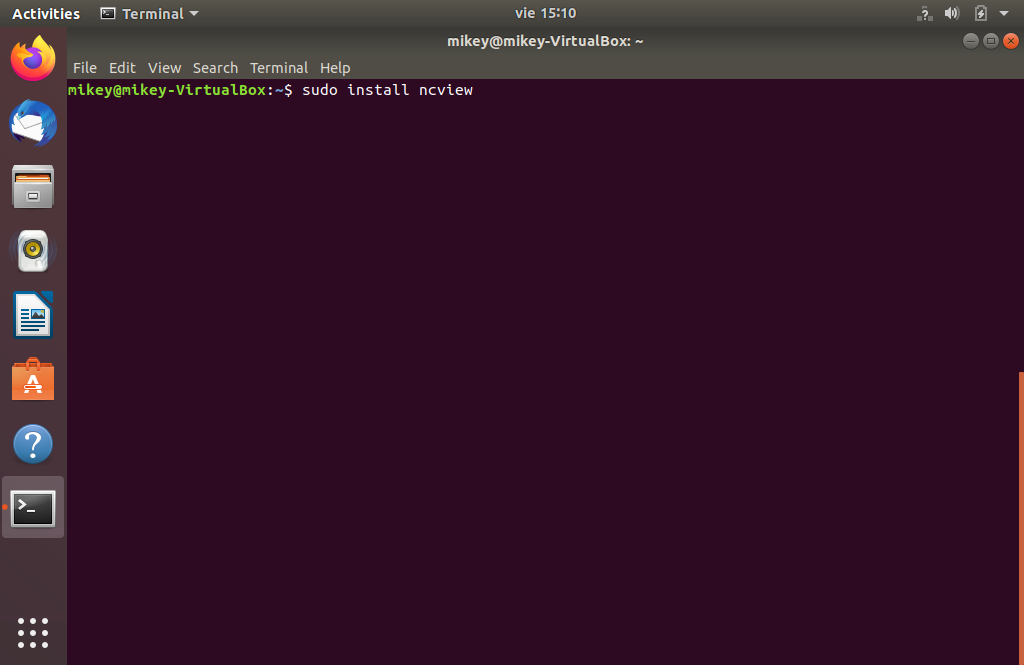
Abrimos la terminal de Ubuntu



Una vez aquí se escribe el síguete comando **sudo apt-get update,** este comando lo que nos ayudará será en actualizar todos los archivos y ver que este correctamente**.**

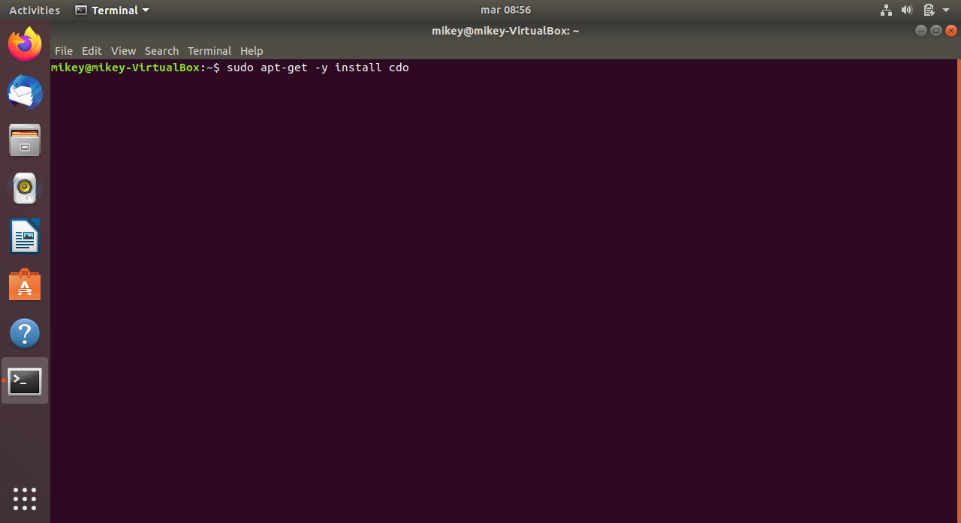


Una vez terminado eso proseguimos con el siguiente comando para instalar ncview. El comando es el siguiente: **sudo install ncview**. Esperamos a que termina de instalar todo lo necesario.

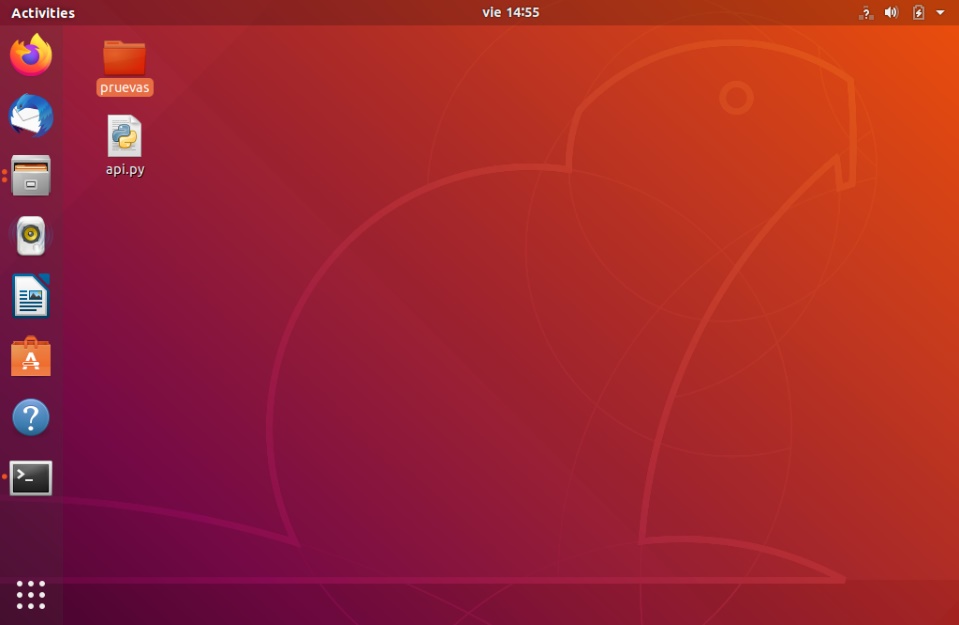


Finalizando la instalación de ncview, volvemos a ingresar el comando de “sudo apt-get update”.

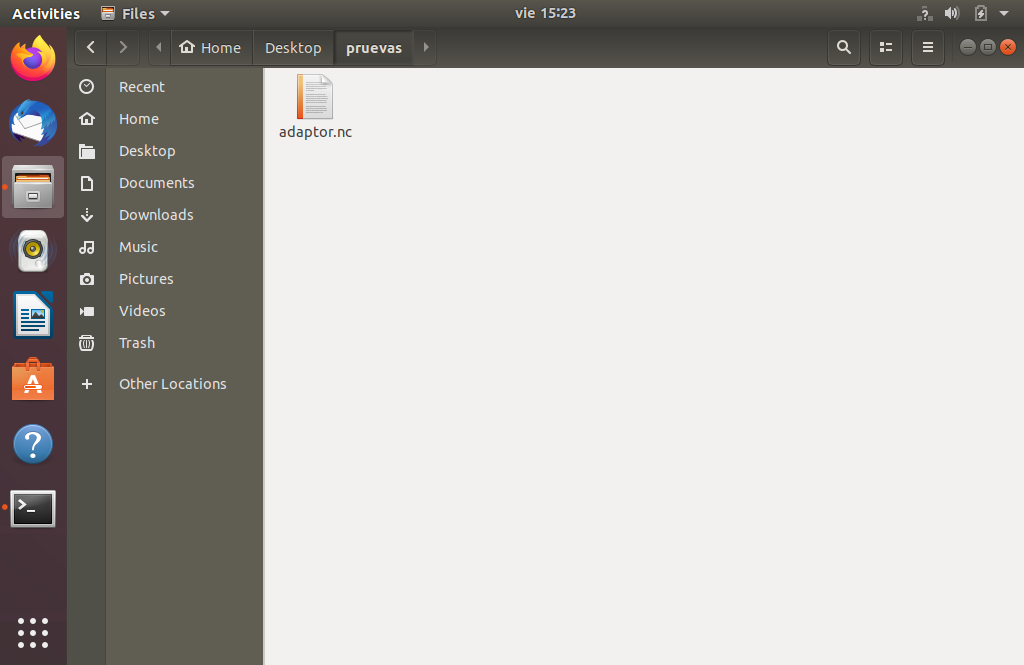
Terminado ese proceso continuamos con la instalación de netCDF CDO, lo haremos usando el siguiente comando: **Sudo apt-get -y install cdo**



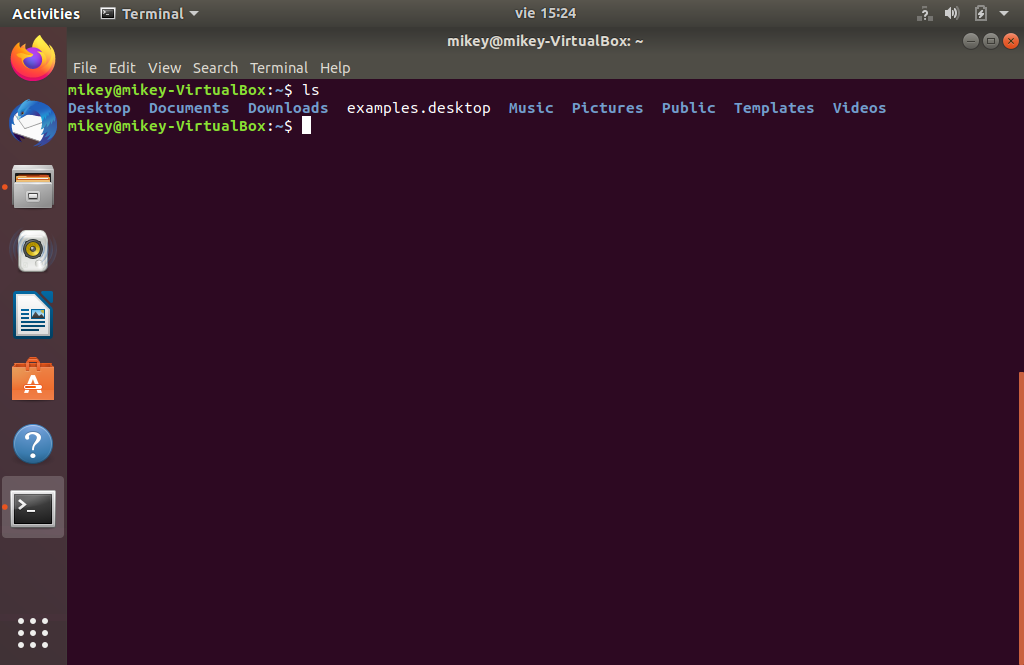
Una vez finalizada las instalaciones de estas librerías, lo siguiente es ver en qué parte del equipo están los archivos que vamos a trabajar, en este caso trabajaremos con la carpeta de “**purevas**” y esta se encentra en el escritorio.



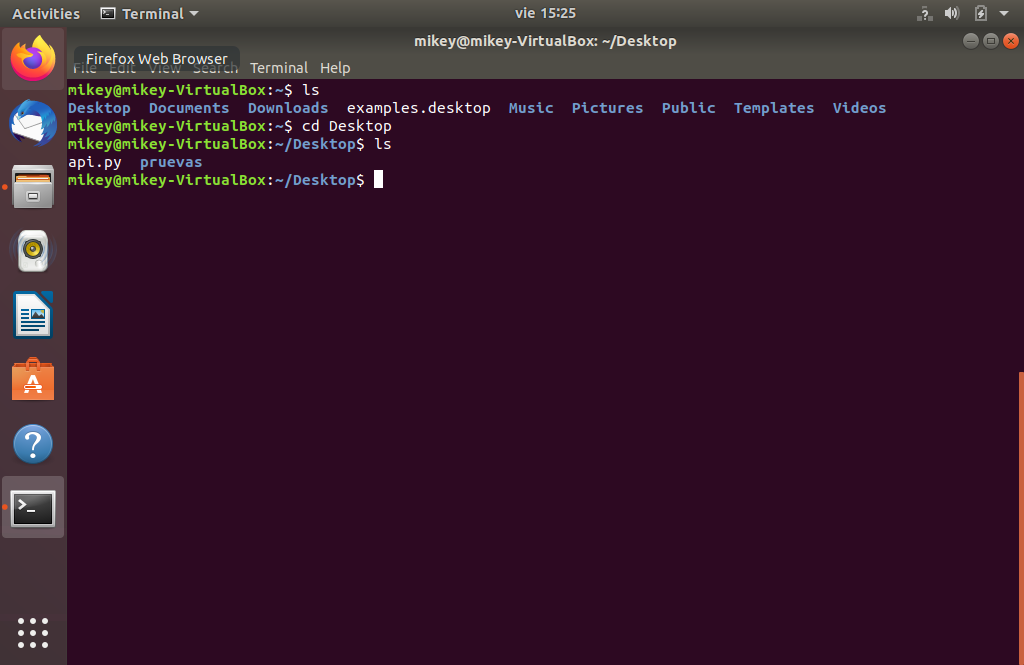
Abrimos la carpeta y visualizamos su interior, vemos que se encuentra un archivo **.nc**, es importante saber que los archivos que trabajaremos deben tener terminación .nc



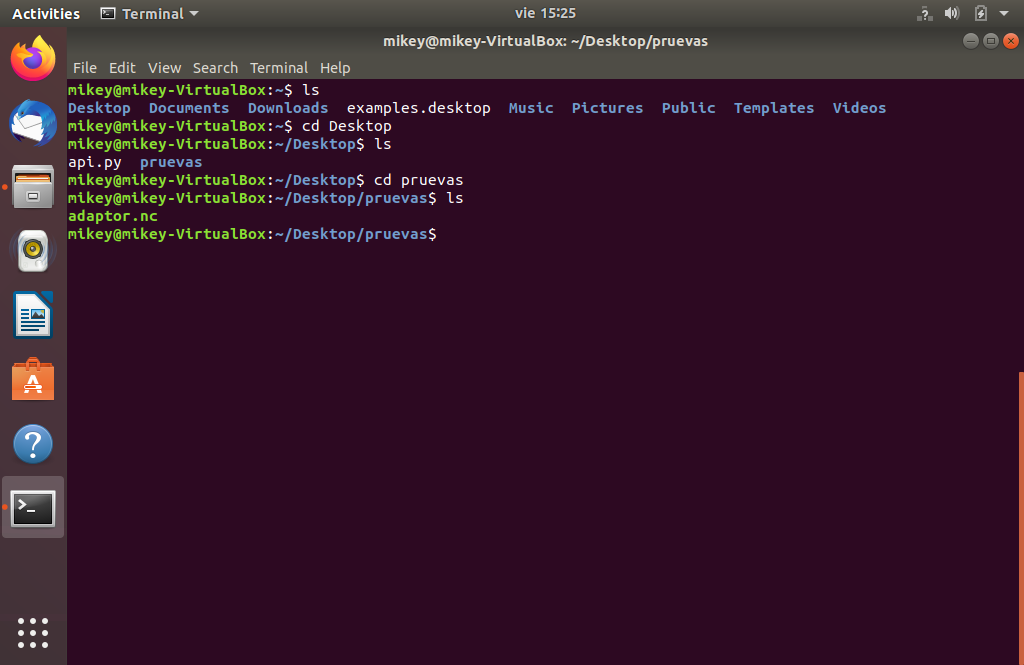
Volviendo con la terminal, ejecutamos comando básico de Ubuntu. “Ls” para preguntar que información se encuentra en la ubicación que estamos. Como nosotros ocupas ir a escritorio seria ir a Desktop



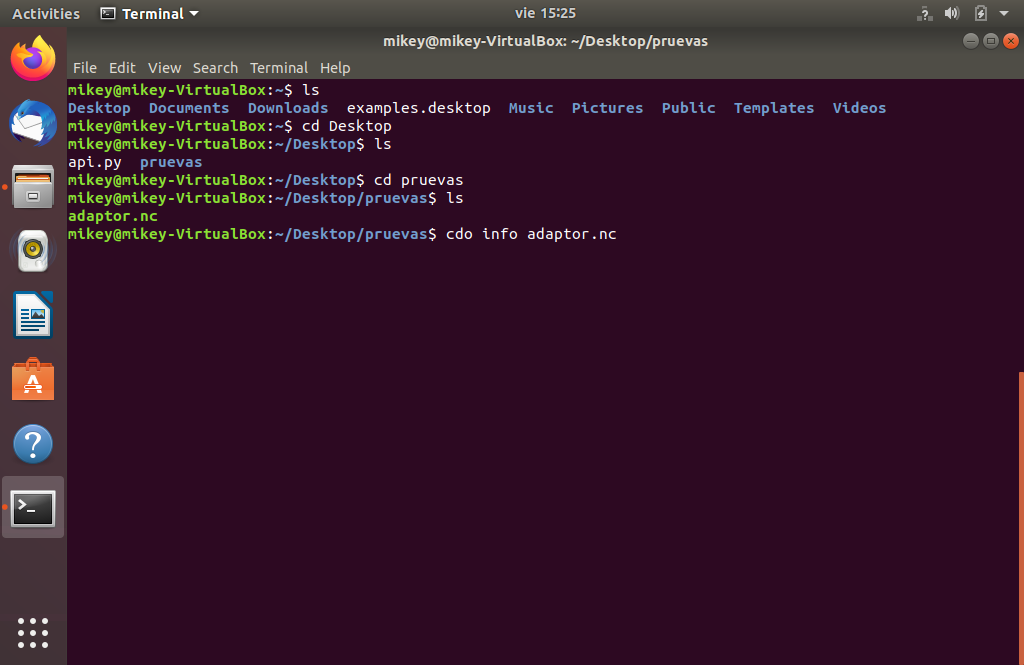
Escribimos el siguiente comando “cd y el nombre del archivo o carpeta” en nuestro caso cd Desktop. Nota lo tenemos que escribir tal cual como esta. Después preguntamos que información se encuentra con “Ls”



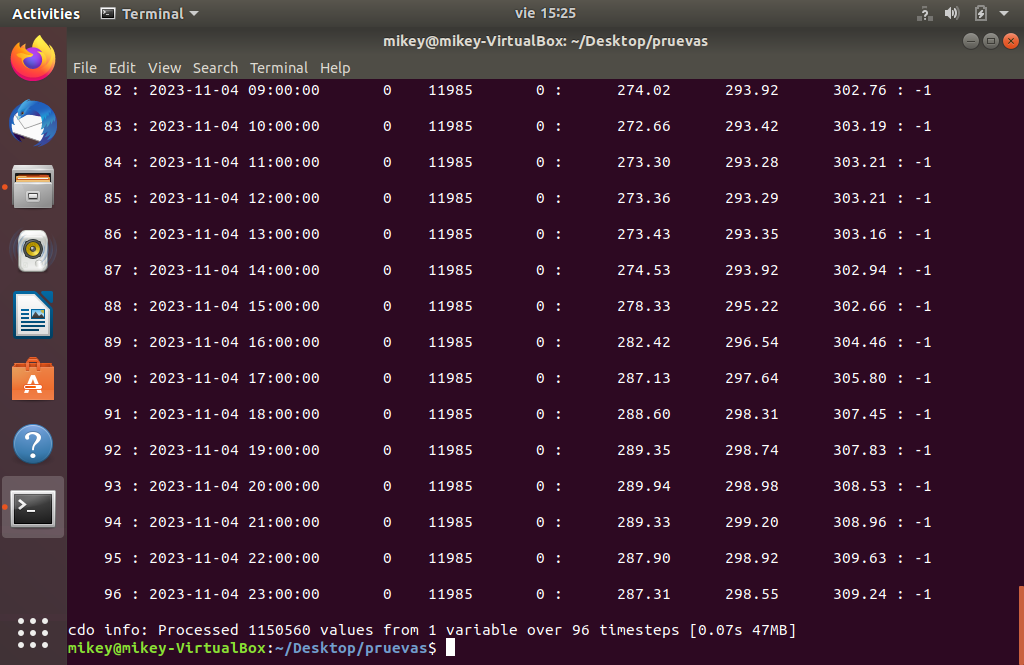
Accedemos a la carpeta prueba con el comando “cd” y revisamos la información/archivos existentes. Vemos que aparece “adaptor.nc”, que anterior mente ya lo habíamos ubicado, pero en un modo internas y no en un entorno de comandos.



Para trabajar con este archivo usaremos comando de NetCDF CDO, aquí explicaremos unos que otros pero para mayor interés se encuentra un archivo de PDF donde trae todo su contenido y todos los comandos existentes mas sus explicaciones de su correcto uso. Dicho esto, continúanos ingresando lo siguiente: **cdo info adaptor.nc**

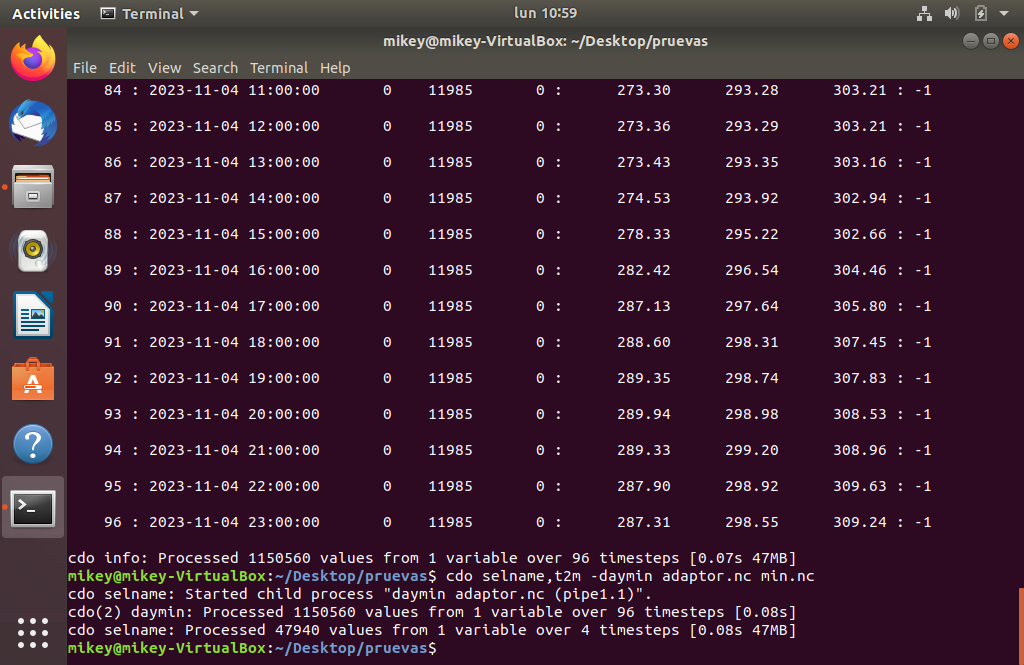


Este comando sirve para mostrar toda la información que contiene nuestro archivo .nc la información al muestra en un entorno de comando como se visualiza a continuación.

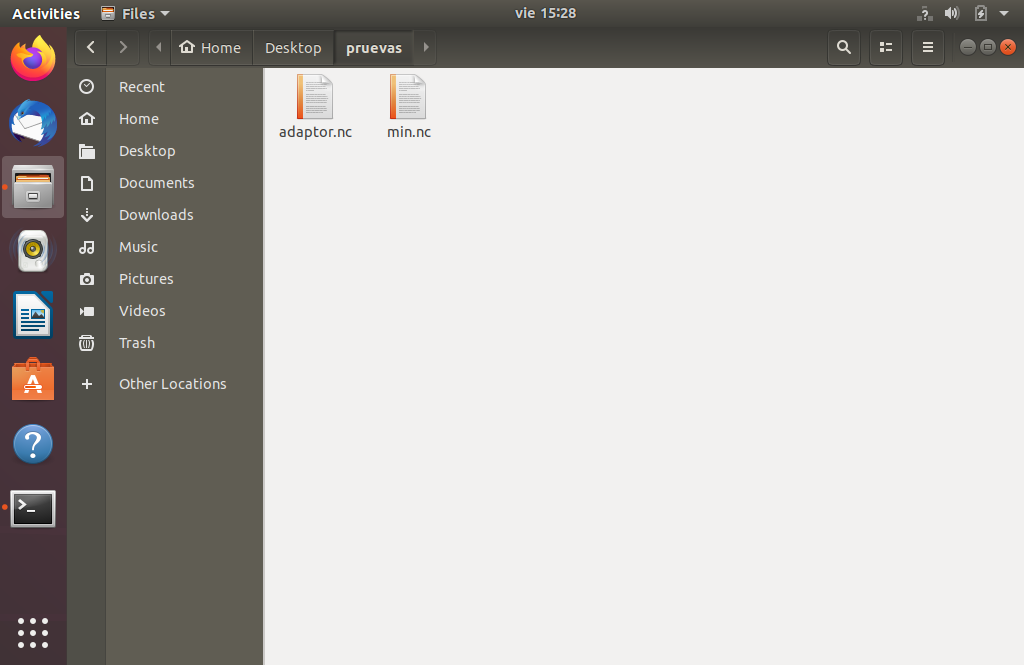


Ingresamos lo siguiente: **cdo selname,t2m -daymin adaptor.nc min.nc**

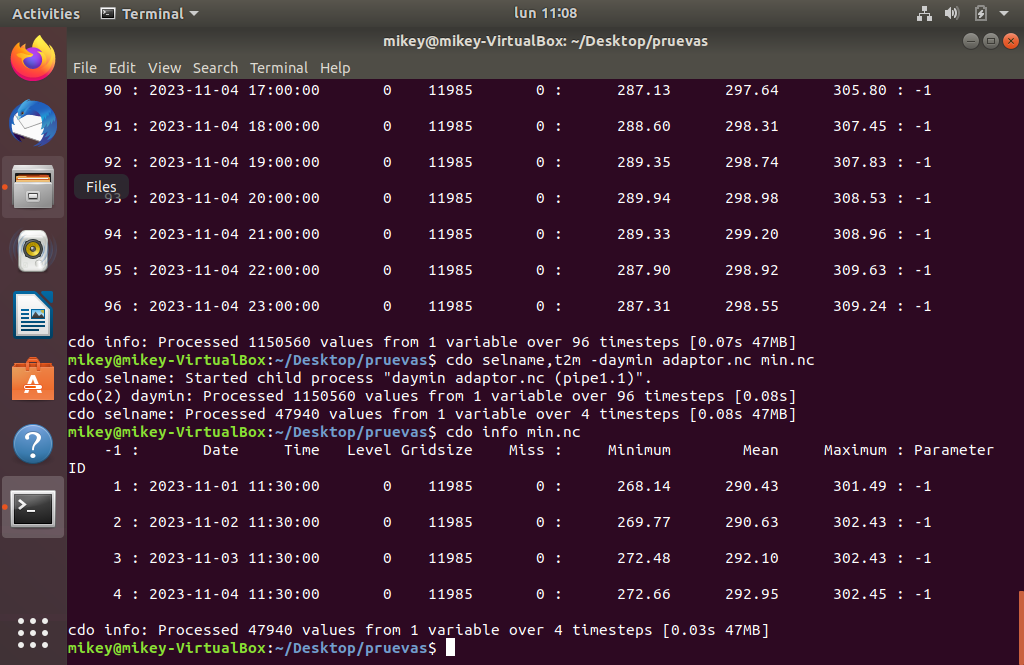
Este comando es para que nos extraiga el dato mínimo de temperatura la cual su variable es (t2m) y le decimos que nos guarde esta información en un archivo llamado min.nc con el mismo formato.



Una vez terminado el proceso nos dirigimos a la carpeta en modo interfaz y visualizamos que se creó nuestro archivo.



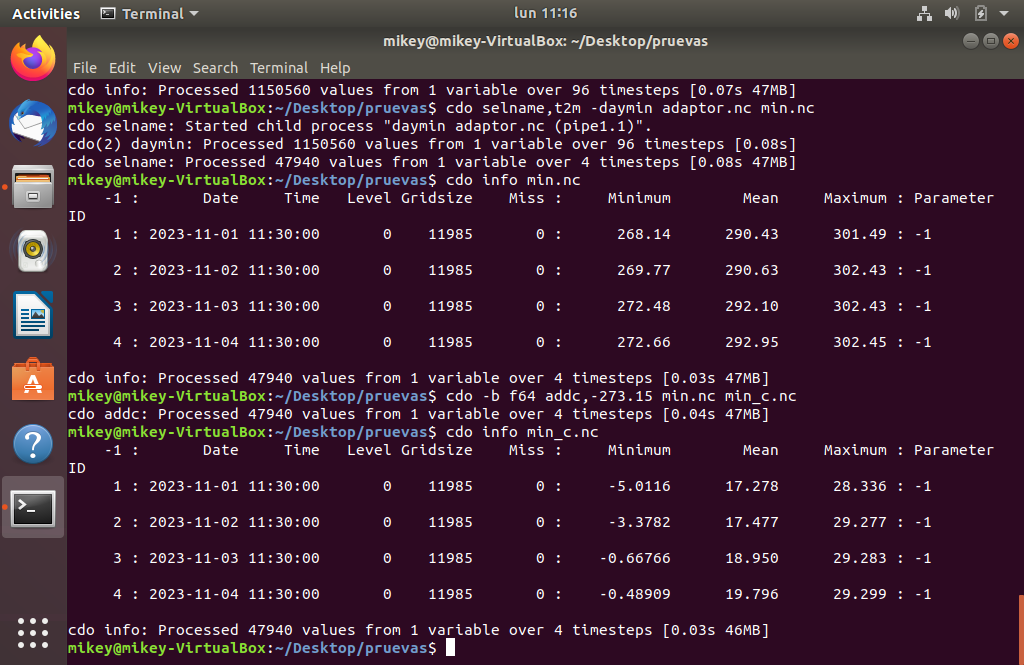
Escribimos el comando para visualizar la información del archivo min, “cdo info min.nc” y podemos apreciar que la información se redujo a solo 4 datos a comparación de los 96 que anterior mente salían. Esto podrá variar dependiendo a que entorno estemos trabajando aquí únicamente se están trabajando 4 días por 24 horas del día por eso nos da un total de 96 datos en el archivo.



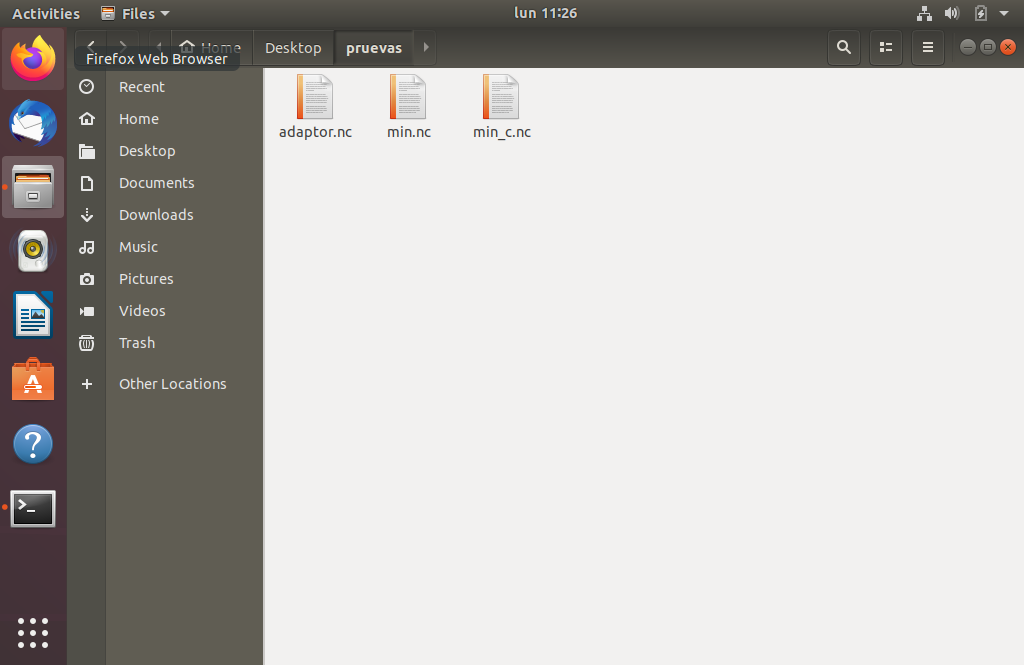
Para lo que estamos trabajando con grados C° y el archivo desde un inicio maneja datos de grados F° así que introducimos el siguiente comando: **cdo -b f64 addc,-273.15 min.nc min\_c.nc**

Lo que hace este comando toma el archivo anterior de datos min.nc nosotros decimos que tome ese archivo eso puede cambiar dependiendo el modo de trabajo de cada quine, volviendo le decimos que la conversión de esos datos nos la guarde en un nuevo archivo llamado min\_c.nc

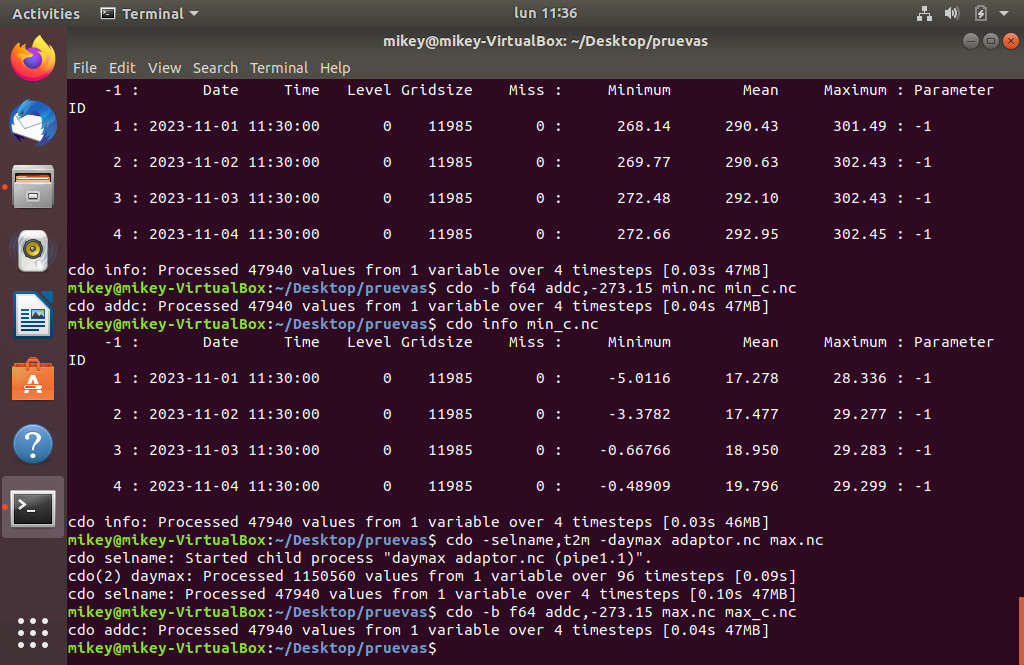
Ejecutamos el comando de info para visualizar el cambio y podemos apreciar la diferencia entre los datos.



Visualizamos que exista en archivo nuevo en la carpeta de pruevas y continuamos



Hacemos lo mismo, pero es esta ocasión será para obtener el dato máximo…



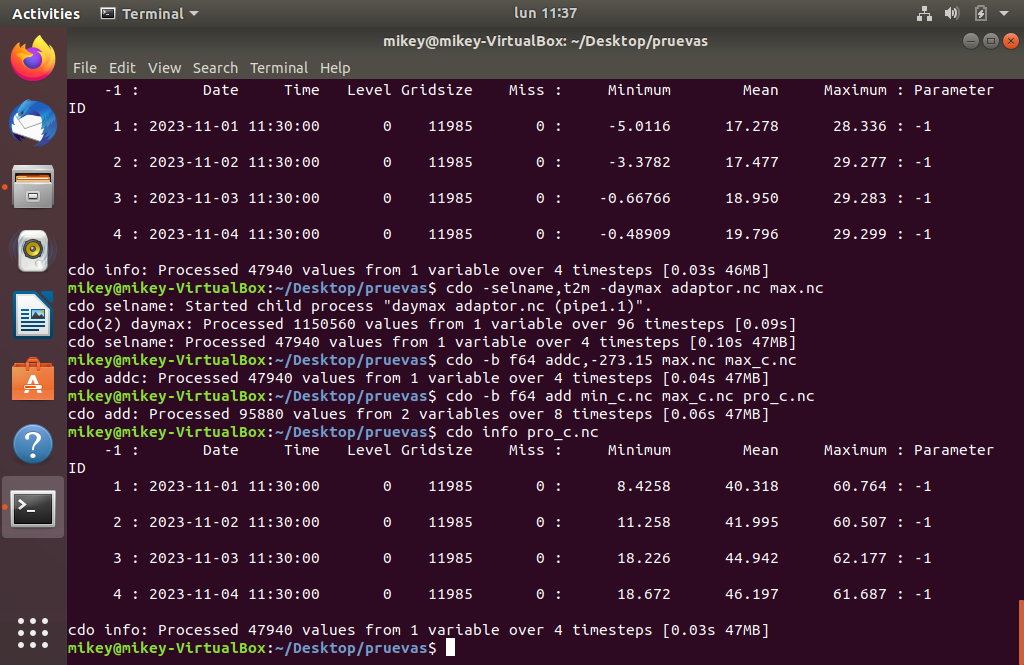
Revisamos que estén los archivos generados…



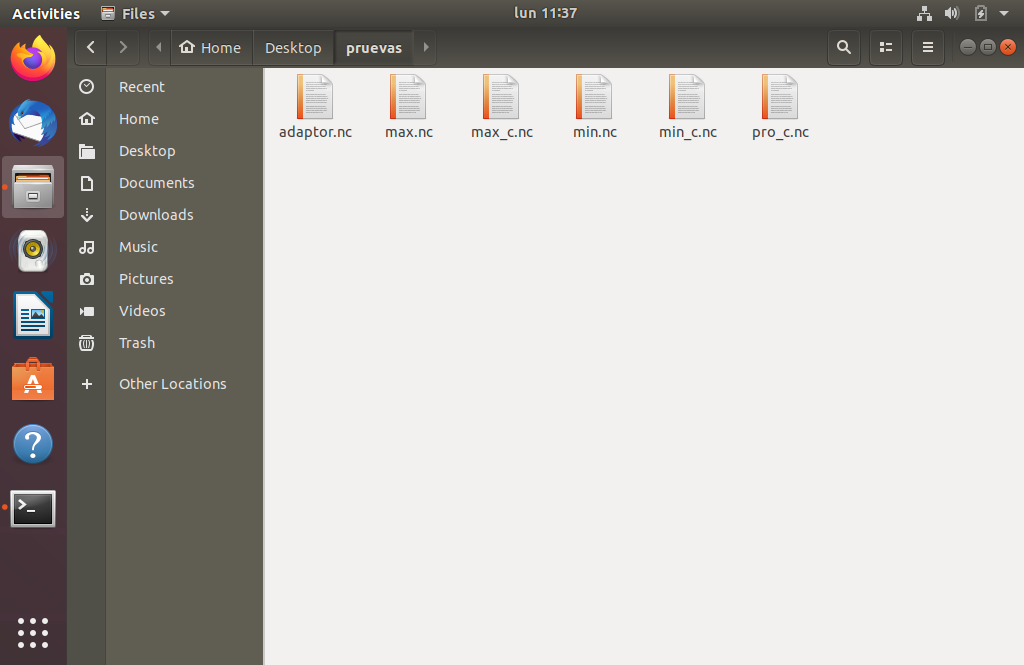
Sacamos el promedio también ese lo necesitaremos para obtener ingresamos el siguiente comando:

**cdo -b f64 add min\_c.nc max\_c.nc pro\_c.nc**

Usamos los archivos anteriores que están en grados C° y al momento de hacer el promedio guarde esa información en un archivo llamado pro\_c.nc

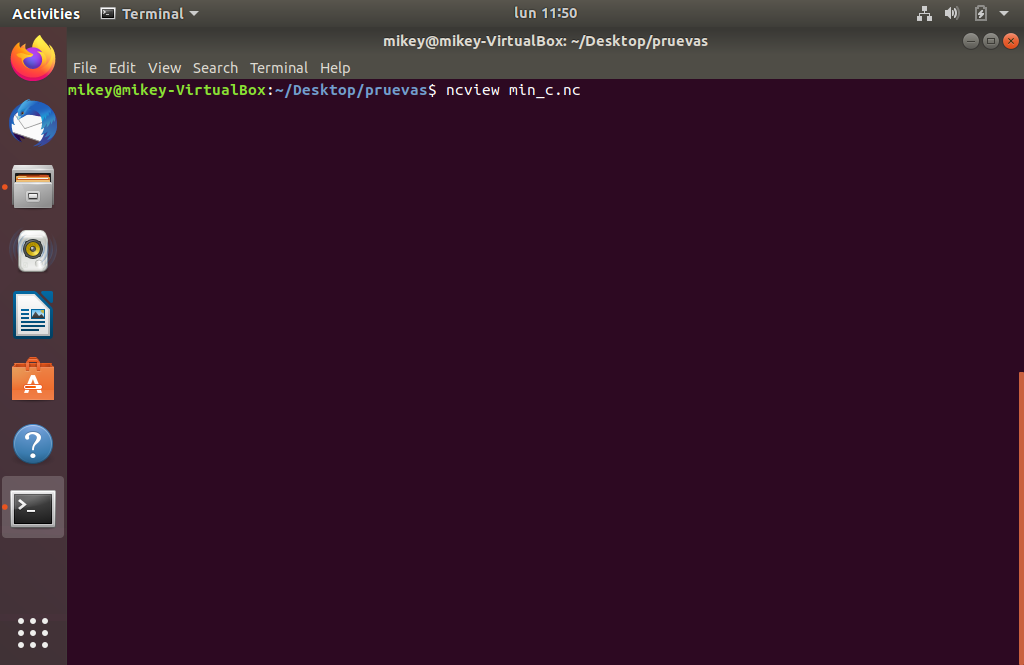


Revisamos que este el archivo correspondiéndote….

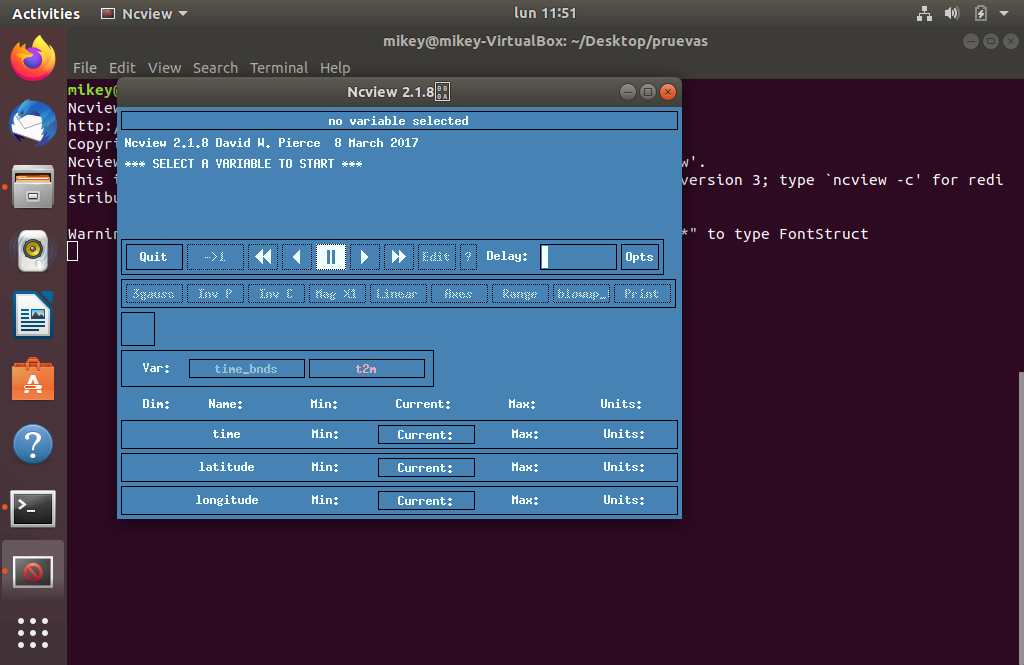


Para visualizar esta información aremos uso de ncview. Esta librería los ayudara a ver de maneta interfaz la información. Para ello vamos a la terminal y escribimos lo siguiente:

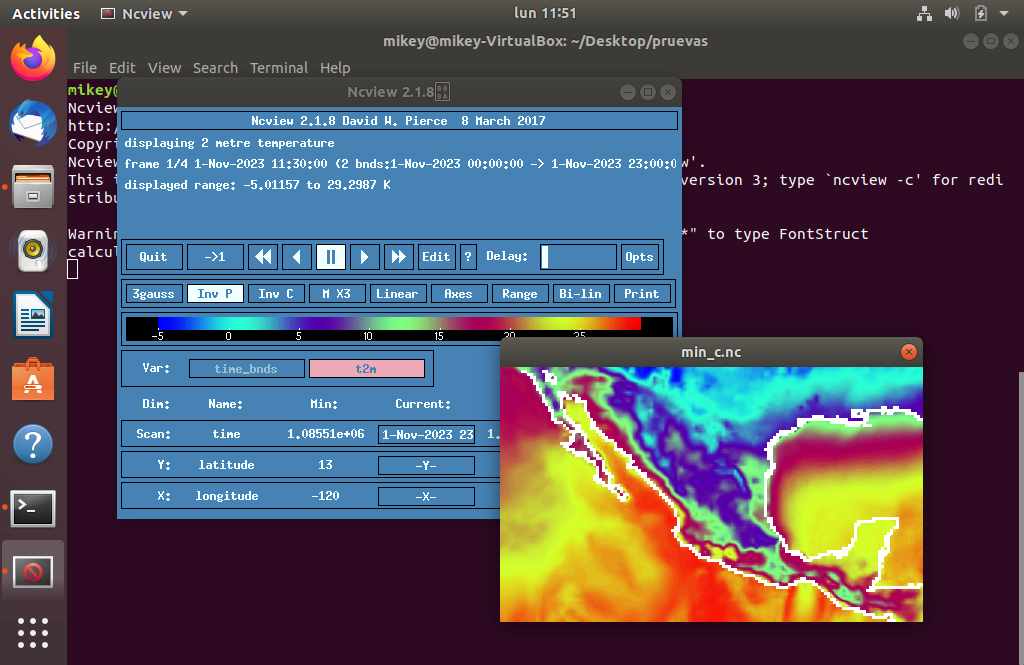
**ncview min\_c.nc**



damos entre y aprecesa un dashboard, nosotros ubicaremos donde dice “var” y seleccionamos la variable de “t2m” ahí puede cambiar dependiendo a todas las variables existentes en un archivó en este caso solo contamos con 1 variables.



Seleccionado la variable aparecerá un mapa mostrando la información de ese día en el dashboard



**Manejo de Unidades Calor (UC)**

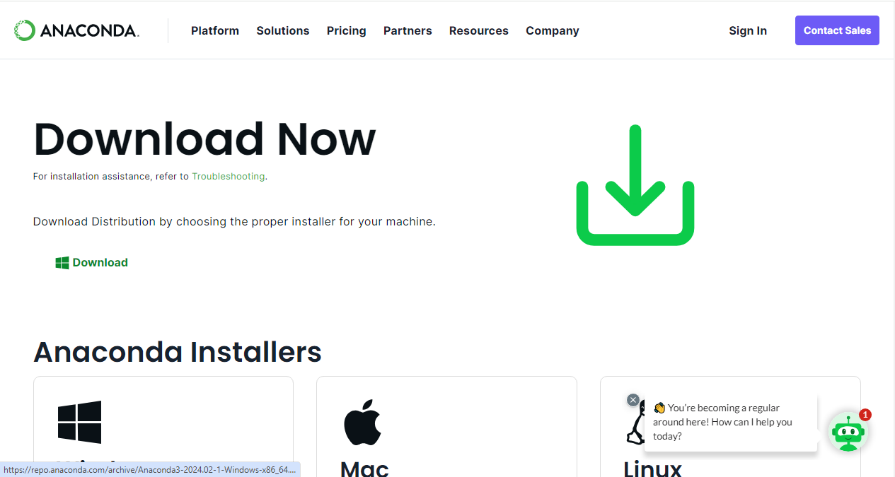
En esta parte a trabajar, usaremos Python, Spyder y varias librerías conforme avancemos iremos haciendo mención de estas librerías. Aquí dependerá de en qué sistema operativo trabajaras, en este caso solo manejamos dos sistemas operativos Windows y Linux (Ubuntu 22.04).

**Windows**

para trabajar en Windows necesitaremos la ayuda de un programa llamado “anaconda”, este programa trabaja con código de Python, trae varias extensiones y facilita su uso en sistemas operativos de Windows… al igual que la descarga de módulos es más intuitiva mediante un dashbord.

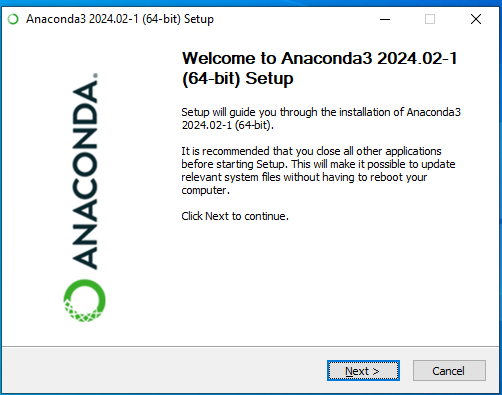
Para descargar el programa ingresaremos la siguiente ruta: [**https://www.anaconda.com/download/success**](https://www.anaconda.com/download/success)

para descargar el programa solicitara que nos registremos, pero en la parte inferior esta solo proceder y descargar sin necesidad de registrarse.



Una vez descargado el programa anaconda, procederemos hacer la instalación de este mismo…

pulsamos el botón de “Next” para seguir avanzando.



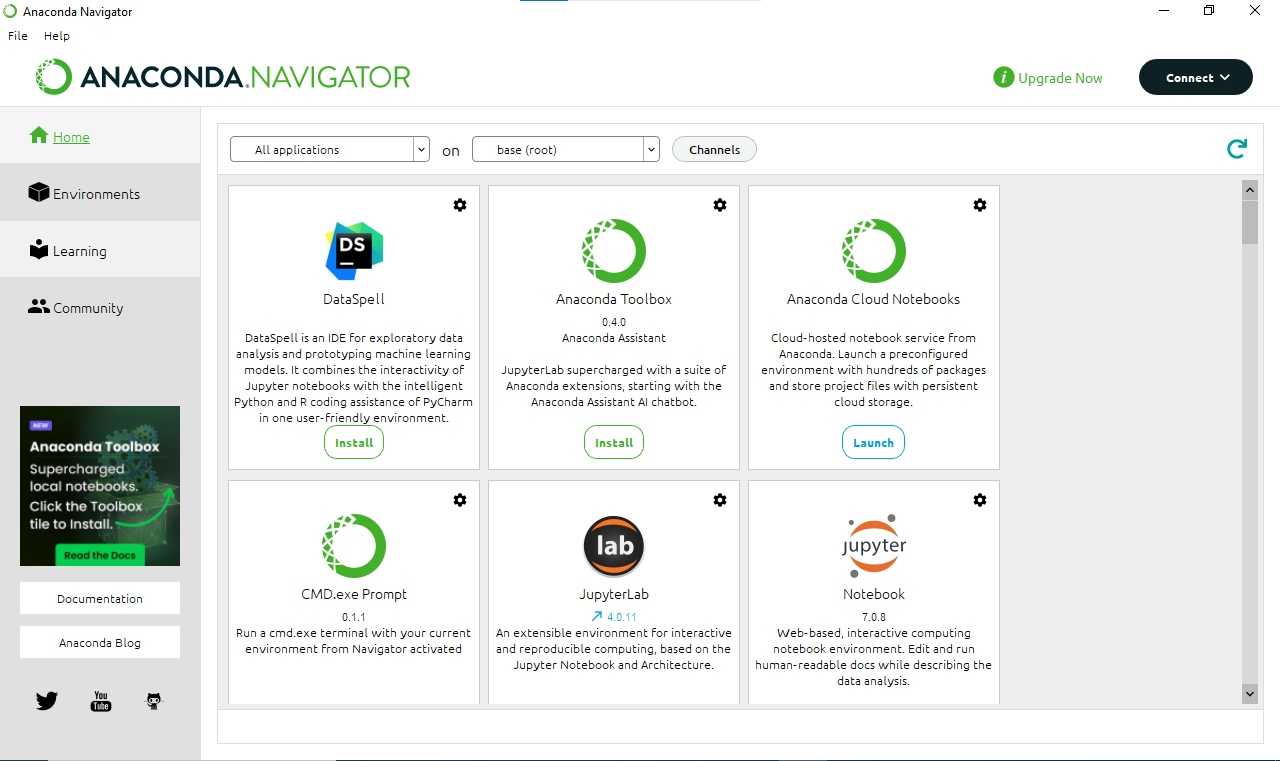
Aceptamos los términos y condiciones, en las siguientes ventanas les damos “Next/Continuar”.



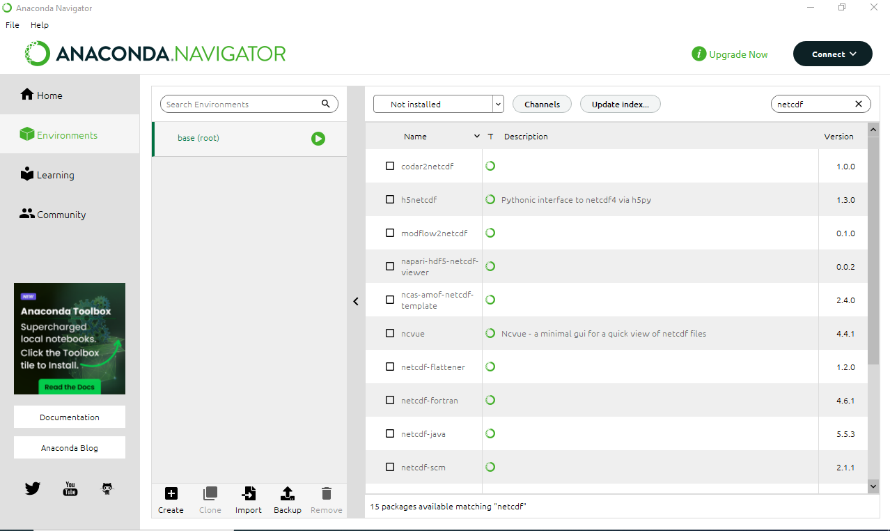
Finalizando eso, empezara la instalación del programa.



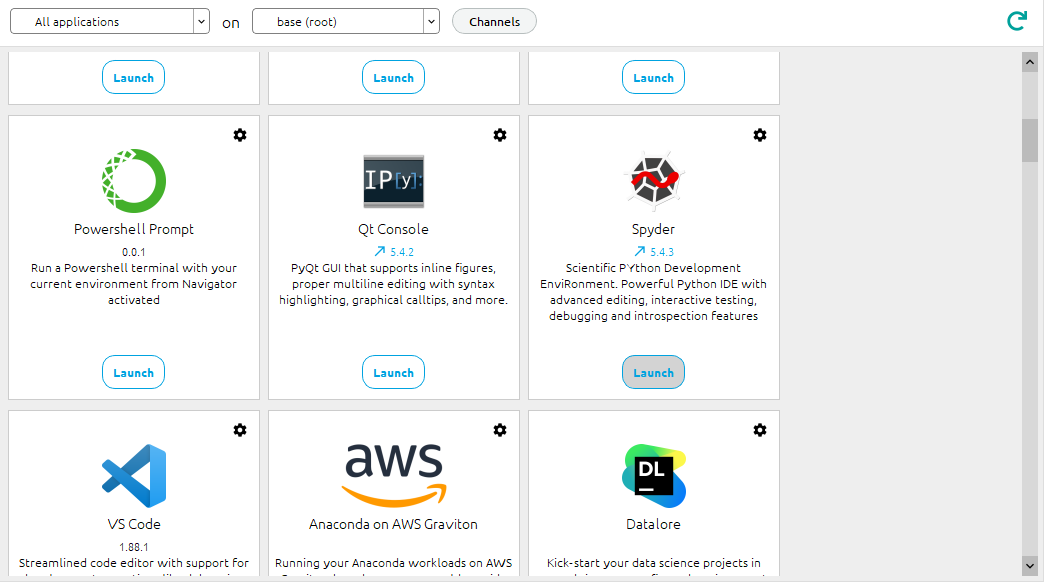
Una vez finalizado pasaremos a abrir el programa desde el buscador de nuestro equipo como “anaconda”, le damos clic.



Nos abrirá una ventana principal con todas las opciones que dispone anaconda, antes de proseguir instalaremos los módulos que ocupáremos el principal es buscar uno llamado “**netcdf**” lo seleccionamos y esperamos a que lo descargue e instale en los módulos.



Después que se instale el módulo, regresaremos a “home” y buscaremos un programa llamado “Spyder” lo ejecutamos y esperamos a que cargue.



Nota de aquí brincaremos a un apartado de **Spyder code**, ya que es lo mismo si se usa Windows o Ubuntu.

**Ubuntu**

Para trabajar en Ubuntu, lo primero es dirigirnos a la terminal, una vez ahí introduciremos los siguientes comandos.

* **cd /tmp**
* **curl https://repo.anaconda.com/archive/Anaconda3-2020.02-Linux-x86\_64.sh --output anaconda.sh**
* **sha256sum anaconda.sh**
* **bash anaconda.sh**
* durante este paso nos preguntara si deseamos instalar el programa le escribimos “yes/y” dependiendo el caso. Una vez finalizado eso nos dirá la ruta dónde estar instalada toda la información de anaconda, si dejamos la ruta predefinida damos “Enter” y continuamos.
* **source ~/.bashrc**

finalizamos todo lo echo con un último comando de Ubuntu “**Sudo apt-get update**” actualizando todas las modificaciones, instalaciones realizadas. Antes de proseguir en la misma terminal escribimos el siguiente comando:

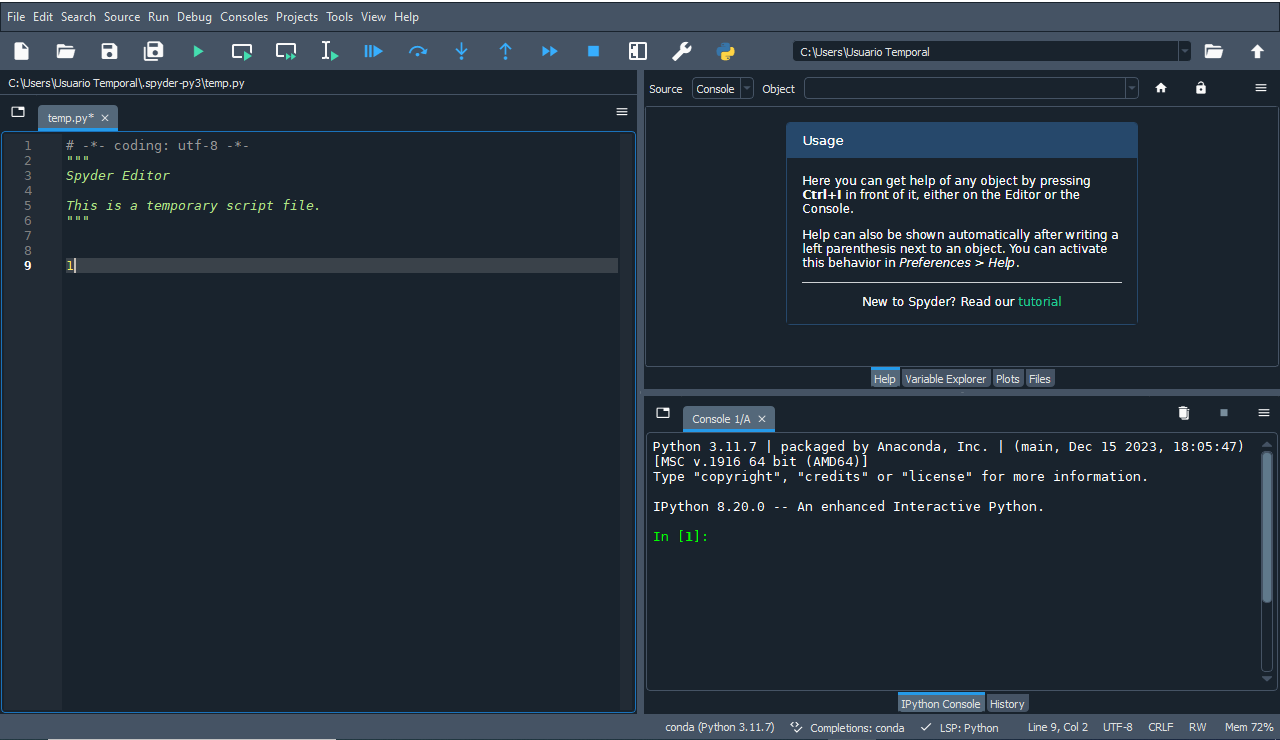
**conda list**

este comando nos mostrara todos los módulos que tiene instalado, una vez finalizado eso escribimos en la terminal “**spyder**” y damos enter. Empezara la ejecución de este programa.



**Spyder Code**

Una vez iniciado spyder nos mostrada la siguiente interfaz, donde en la parte Izquierda será donde se escribirá código de Python, en la parte de la derecha se divide en dos ventanas una superior y una inferior, comenzando con la parte superior en esta mostrará todas las variables y funciones que se tengan en el código, para poder analizar mejor que información lleva cada variable. Y en la parte inferior es la consola donde nos mostrara si el código fue ejecutado correcta mete o existen errores, los cuales también nos marcara en esta misma parte.



Tips:

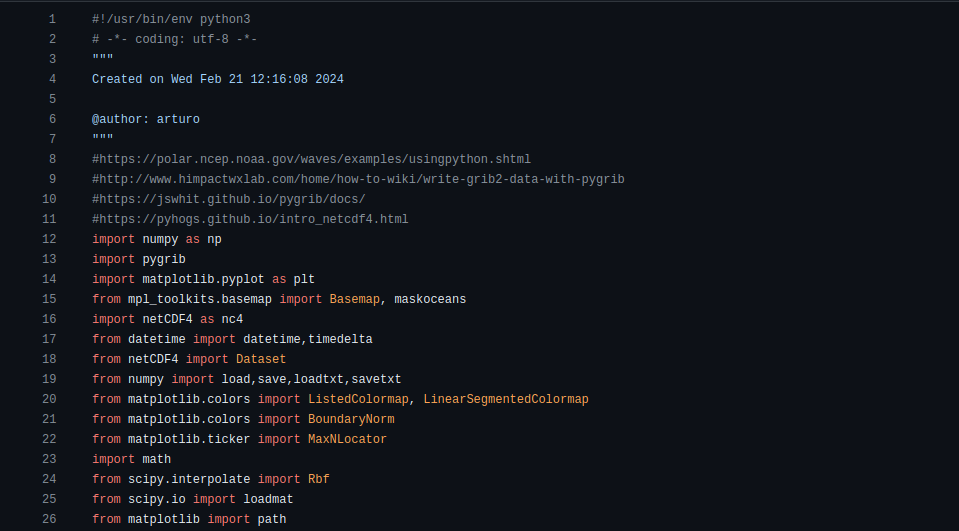
* con F9 – ejecutas rápido el código, solo seleccionas la parte que quieras trabajar

Para descargar el codigo estara en la siguente liga <https://github.com/infinity416/script_UGC/blob/3b7f2e199986db171028171a95316530ca04cee7/Uidades_Grado_Calor.py>

una vez copeado el codigo, abriremos spyder o si ya esta abierto en dado caso de no haberlo cerrado. Pegamos el codigo del lado izquierdo y esperamos a que carge el codigo.

El codigo se era explicando parte por parte para poder ir entendiendiendo su funcionamiento.

Lo primero al inicar cualquier programa, es escribri las librerias a utilizar, en este caso debemos tener instaldas previamente todas estas librerias y mandar a llamar aquellas que pertenecen a la librería. En caso de que no se cuenten instaladas las librerias puedes ir instlando manualmente las librerias, depediendo el caso si se usa un sistema operativo sea Windows o Ubuntu.



**Parte1**

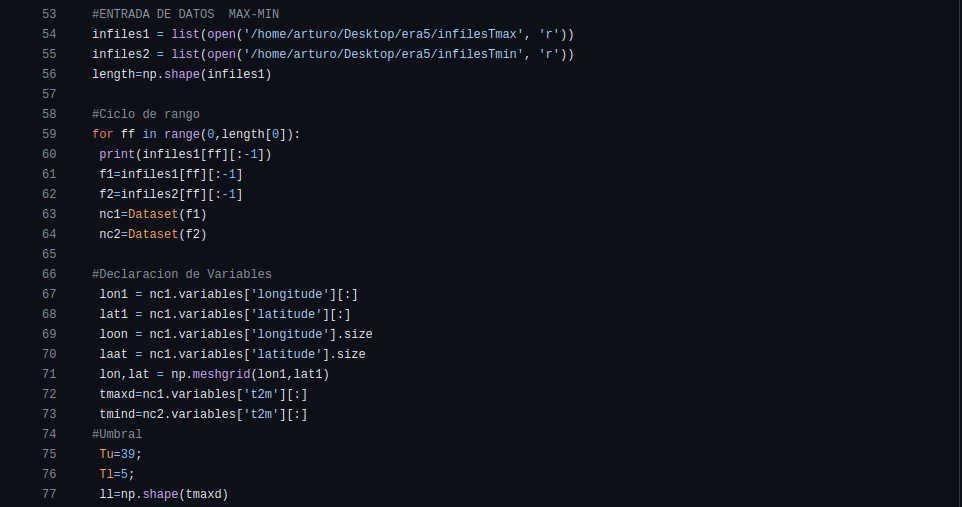
En la parte 2 se encuentra declarado ua tabla de colores en forma de matriz, para esto debemos saber del tema de los colores en codigo de los cuales son los tres colores primarios (Red, Yellow y Blue).

En la parte 3 se declara la entrada de datos, los cuales se van a utilizar, para esto tenemos declaradas dos variables (infiles1 y infiles2) cada una se encarga de traer datos de temperaturas maximas y minimas. Esta misma informacion se ve representada en las variables (nc1 y nc2).

**Parte2**

Parte 4 extraeremos la infrormacion que nececitamos que es la longitud y latitud al igual que la el dato que nececitamos en este caso es temperatura.

Parte 5 se declaran el umbral a utilizar esto depende del uso, esos datos pueden cambiar.

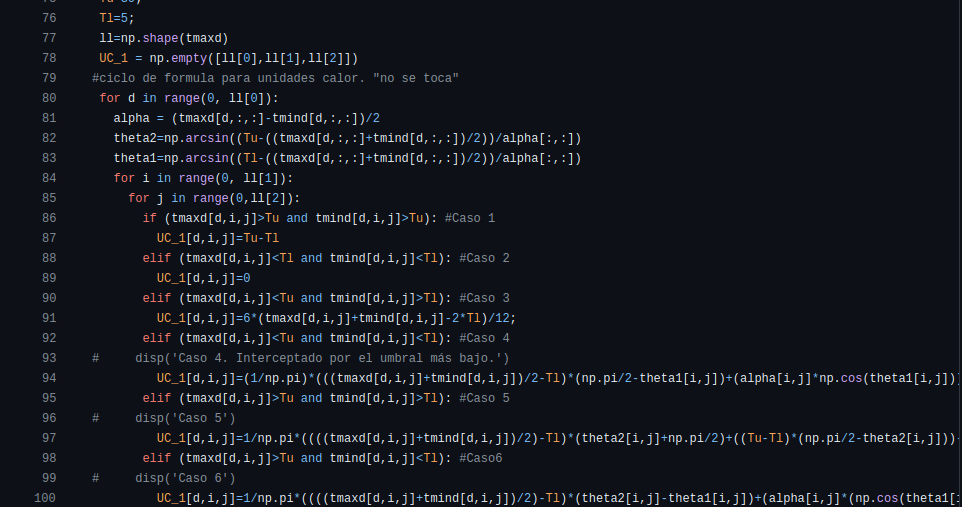


**Parte3**

**Parte4**

**Parte5**

En la parte 6 es un ciclo para preguntar que tipo de dato tenemos y con que formula corresponde, toda estas formulas ya estan establecidas, colo que son adaptadas en codigo para agilizar el calculo de estos mismos datos.

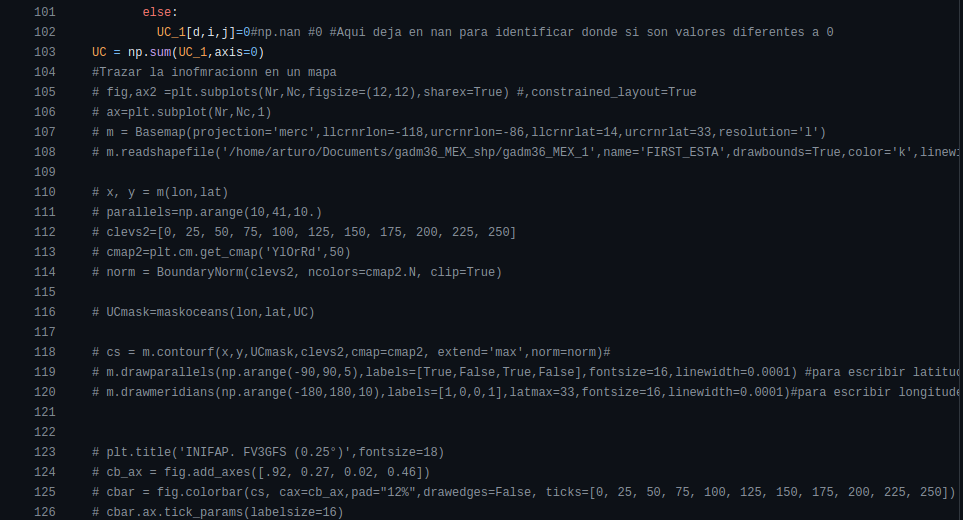


**Parte 6**

Parte 7 toda esa informacion procesada en ese ciclo se ira alojando en una variable temporal asignada como **UC.**

Parte 8 esto es extra, en caso de nececitarlo para ir representando la infromacion obteniada se generara un mapa,para esto ahi que cargar previamente un archvio que es para trazar las diviciones politicas de cada estado.

Parte 9 ahi que definir los datos que anterior mente extragimos (longitud y latitud, al igual que la nueva variable que esta guardando los datos llamada UC).



**Parte 9**

**Parte 7**

**Parte 8**

Para la parte 10 reorganizaremos los datos que enteriormente se estan trabjando, asignadolos en unas nuevas varibales, para los datos que se fueron generando en la parte 6 y 7 asginadolos en la varibale **UC** se le asigna una nueva ahora identificandolo como **G2**, para la longitud (**loon**) se le asigna **E2** y por ultimo latitud (**laat**) se le asigna **C2.**

Por ultimo la parte 11 en esta ultima parte del codigo, es para generar un archivo de excel con todos los datos generados y poder hacer uso de estos mimos. Cabe de mecionar que este es un metodo. Existe otro metodo generando un archivo netcdf4.



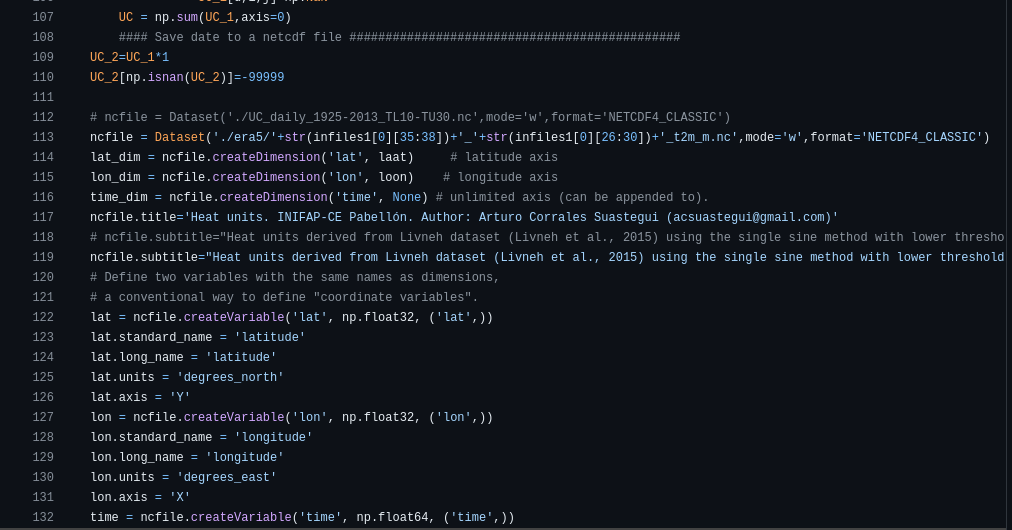
**Parte 10**

**Parte 11**

Toda esa infromacion la encontraran en este codigo https://github.com/infinity416/script\_UGC/blob/3b7f2e199986db171028171a95316530ca04cee7/create\_.NC.py

Esta parte del codigo puede ir despues del parte 7, como se puede apreciar en la parte 12 se genera una nueva varibale **UC2** la cual nos servira para gardar los datos en este archvio de netCDF (**.nc**).

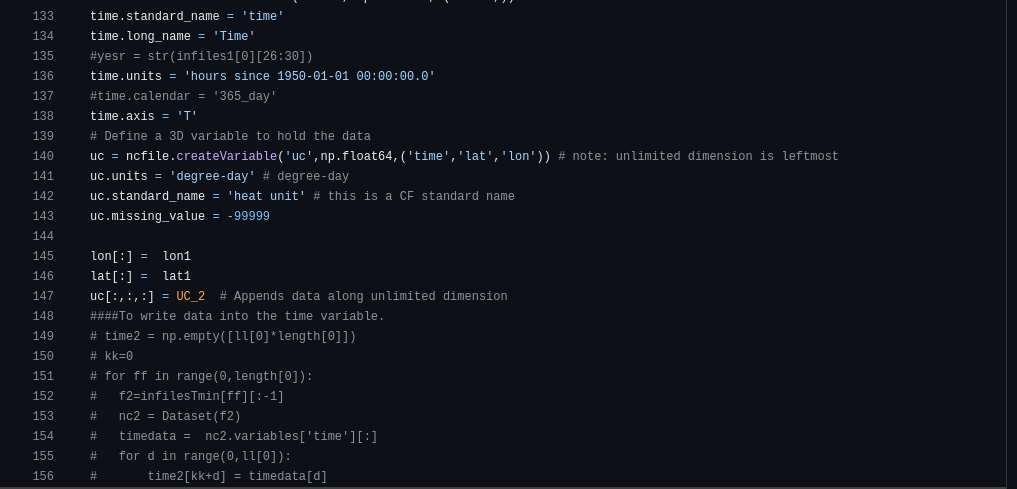
Parte 13 inciamos declarando las vairbles, cada variable representara una informacion, nuevamente una para longitud, latitud y por ultimo una las las UC con la variable UC2. Mas bajo en el codigo detallamos las caracteristicas de cada varibale, tales como su nombre, tipo de dato, una descripcion, en que eje ira el dato, ya que netCDF maneja datos por mallas. Y asi respetimos la informacion para cada una de las varibales.



**Parte 12**

**Parte 13**

Parte 14 resiganmos la s varibales que anterior mente describimos, en este formato que se menciono anterior mente, ay que netCDF maneja datos en forma de malla.



**Parte 14**

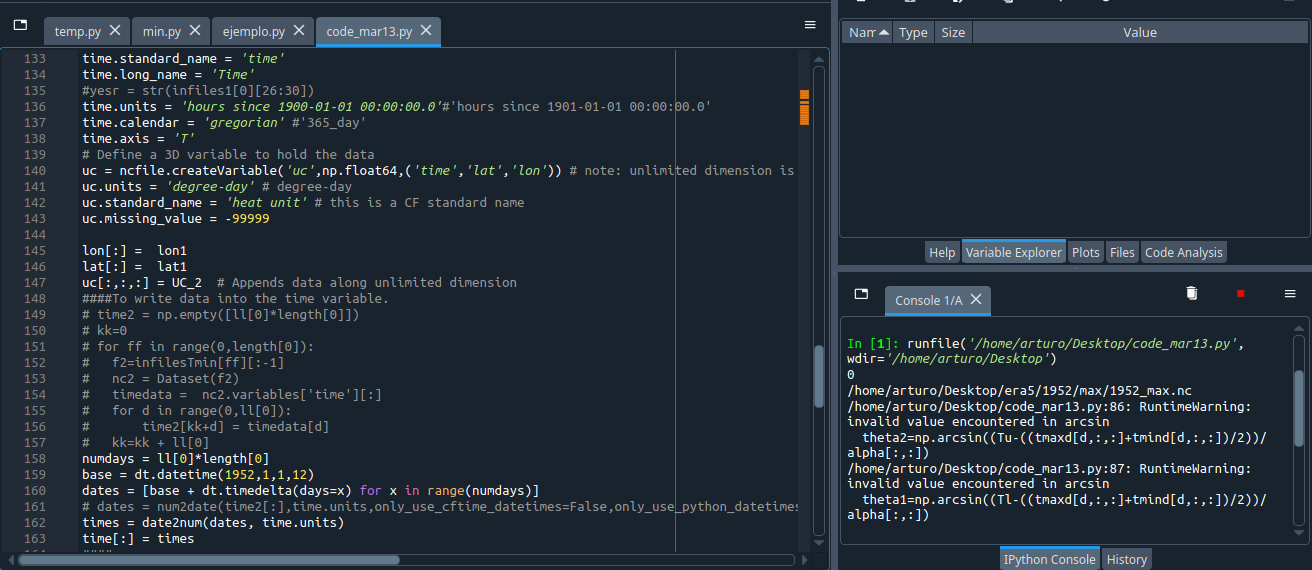
Parte 15 aquí ahi que tener mucho cuidado con lo que es la fecha, ya que esto nos podria generar errores al administra la informacion ya que esto ira guiandose con la descripcion que le dimos al ala variable tiempo anterior mente, en caso de ir cambiando la fecha para los archivos, hacelor en la ***linea 136*** revisando esa parte. Pasarmos a la siguiente parte 16 donde solo seria decirle que muestre el mensaje en la consola de queel archvio fue generado correctamente. Paso que se me olvido mencionar en la misma descripcion viene, una linea de codigo la cual sirve para nombrar el archvio que se va a generar para ser mas exactos en la **linea *117.***



**Parte 15**

**Parte 16**

cuando todo este bien el codigo y no marque errores, cagado bien los archivos de entrada de dato, ejecutamos el codigo mostrandonos algo asi en la parte de la consola en spyder.



Una vez finalizado nos mostrara la siguente informacion en la misma consola, claro que el proceso del codigo fuera el correcto, en caso de salir errores, leer lo mencionado e investigar a que se deba ese error. Tambien podermos observar cuantas variables se procesaron al memento de correr el codigo en la parte superior de la consola.

