WORKSHOP

-

Lab 3

Creación de un servicio ATP con Developer Cloud Service

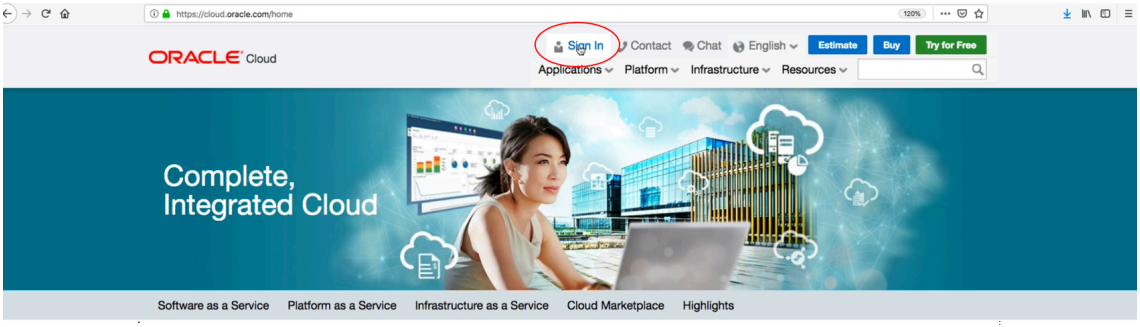
Objetivo del taller

Crear una instancia de Base de Datos ATP (Autonomous Transaction Database) para utilizar como alternativa a lo visto antes con Kubernetes y PDB Oracle Database Multitenant.
Se verán y explicarán las ventajas de este servicio ATP, que resulta ser más escalable, seguro y ejecuta más rápido que PDB sobre Virtual Machine o Baremetal. Podrá experimentar cómo un servicio autónomo facilita last areas de desarrollo de aplicaciones Cloud Nativas y Microservicios empresariales.

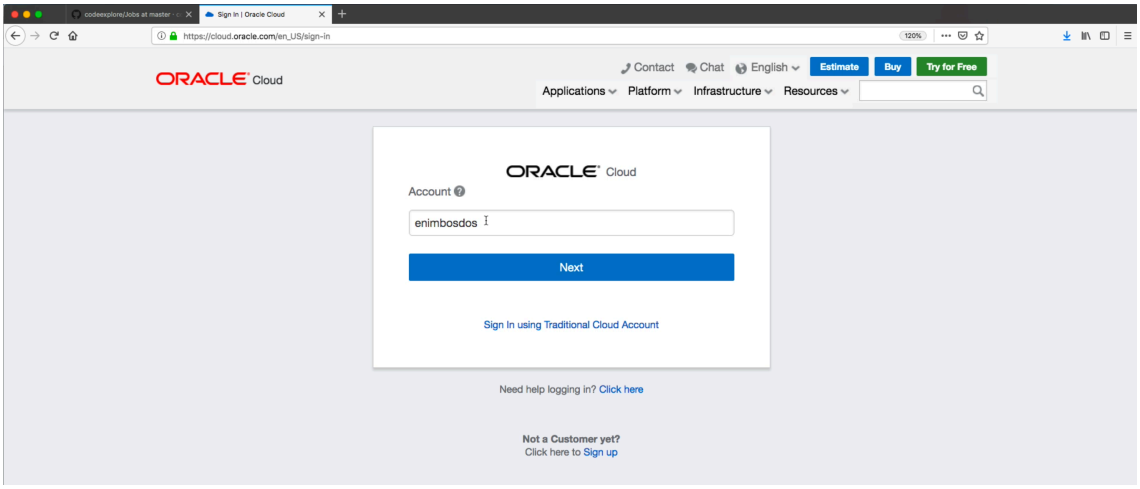
Instrucciones

Preparación del entorno

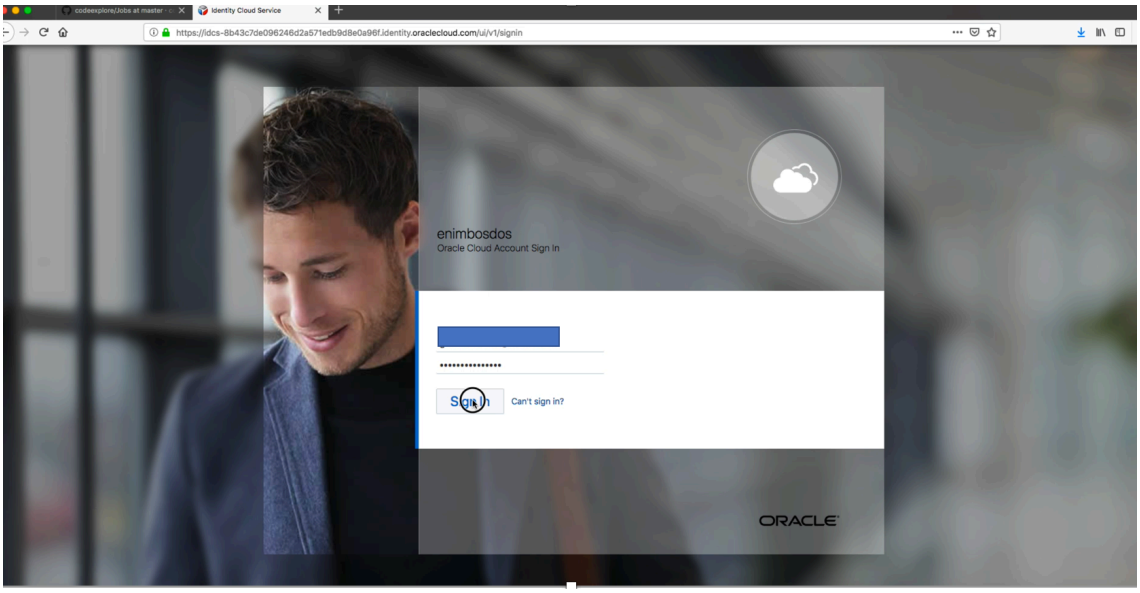
Acceder al entorno cloud del taller



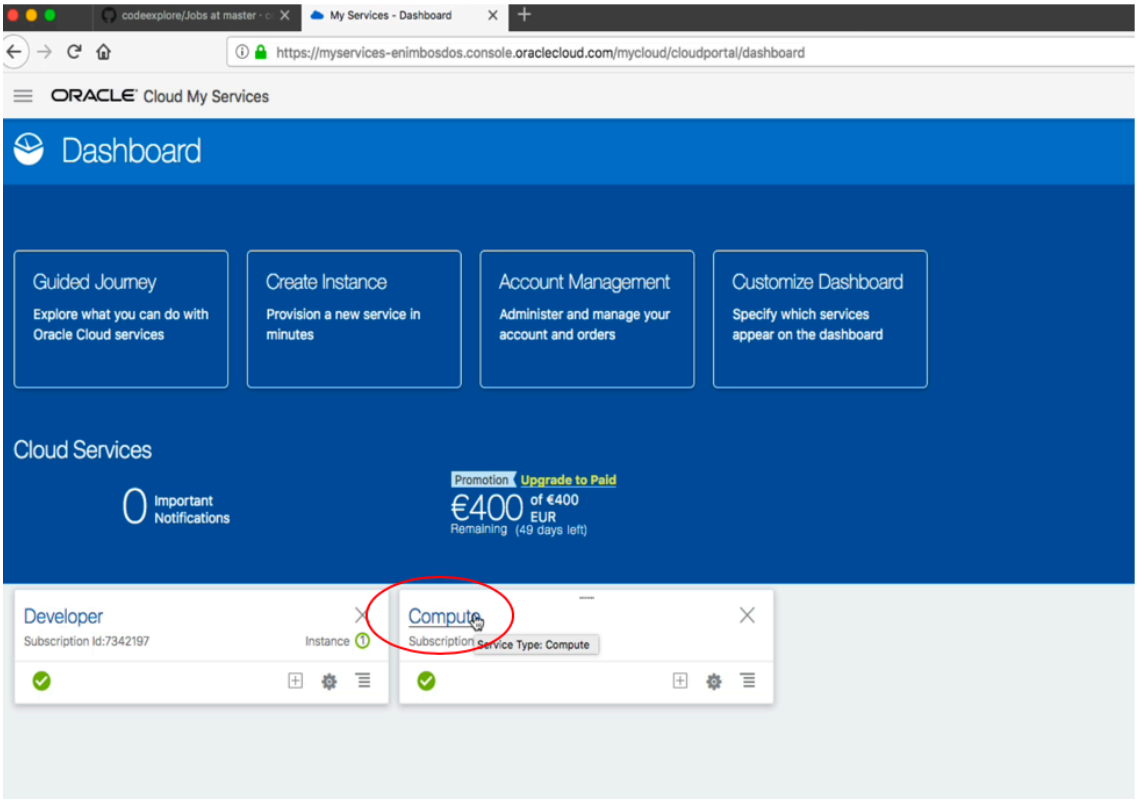
Seleccionar la cuenta proporcionada en las credenciales



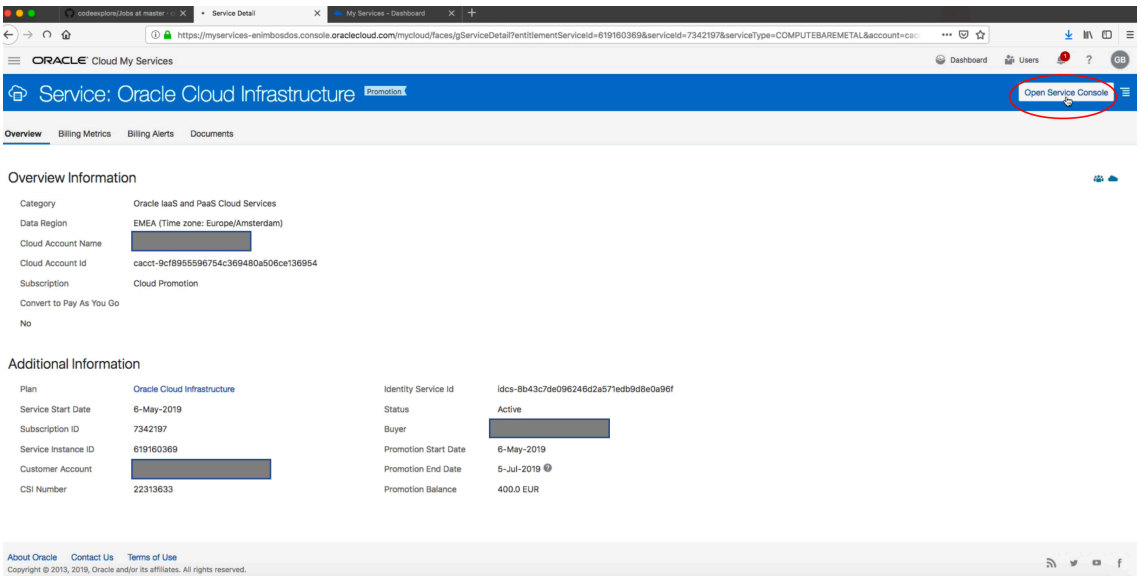
Acceder con las credenciales proporcionadas



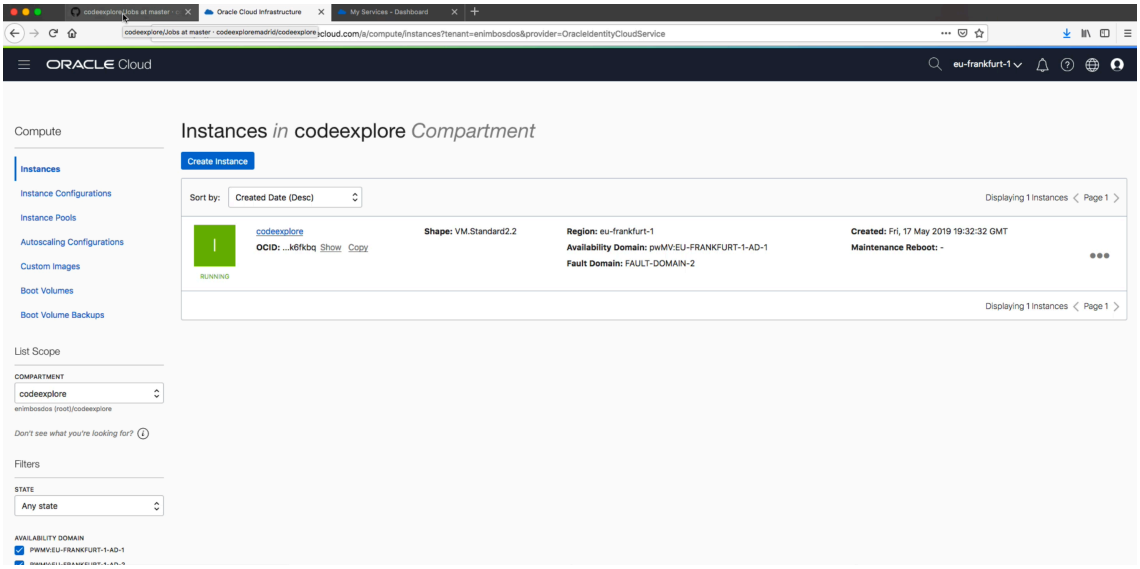
En el entorno deben aparecer dos servicios ya provisionados: Developer y Compute. Hacer click en el servicio Compute.



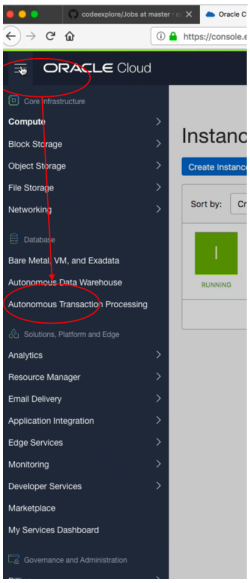
La información del servicio aparecerá. Acceder a la consola del servicio tal como se muestra en la figura.



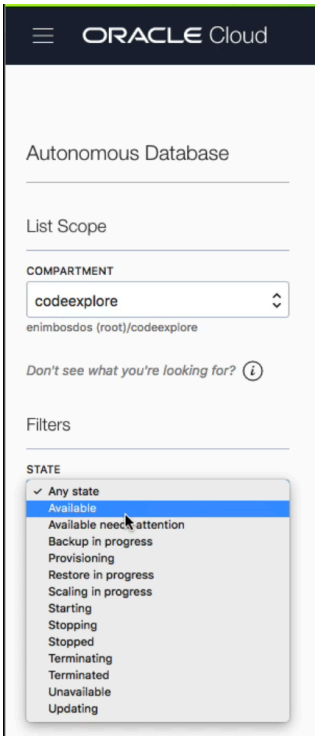
Verificamos que tenemos una virtual machine creada.



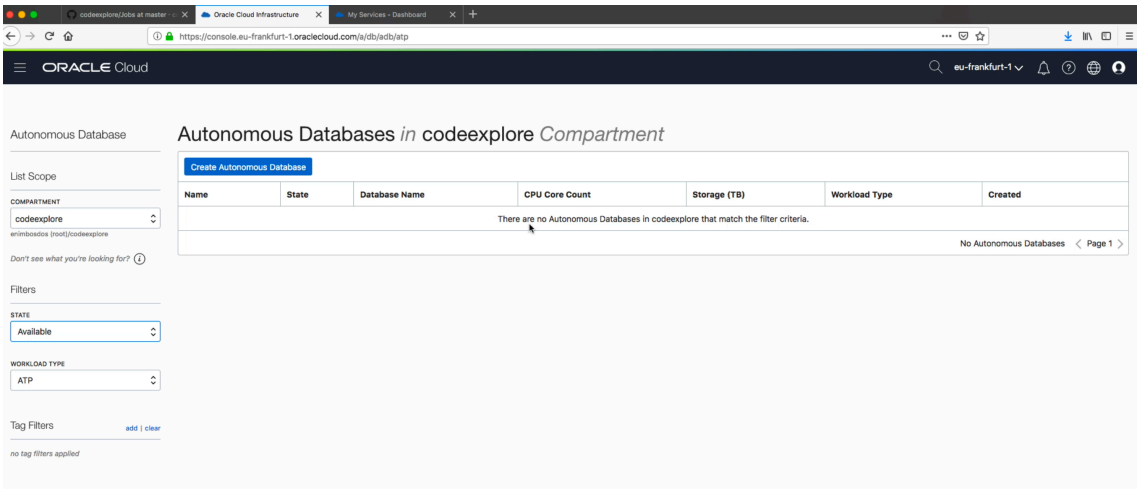
Verificamos que no tenemos ningún servicio ATP navegando en el menú lateral



Seleccionar “Available” como opción de filtro.

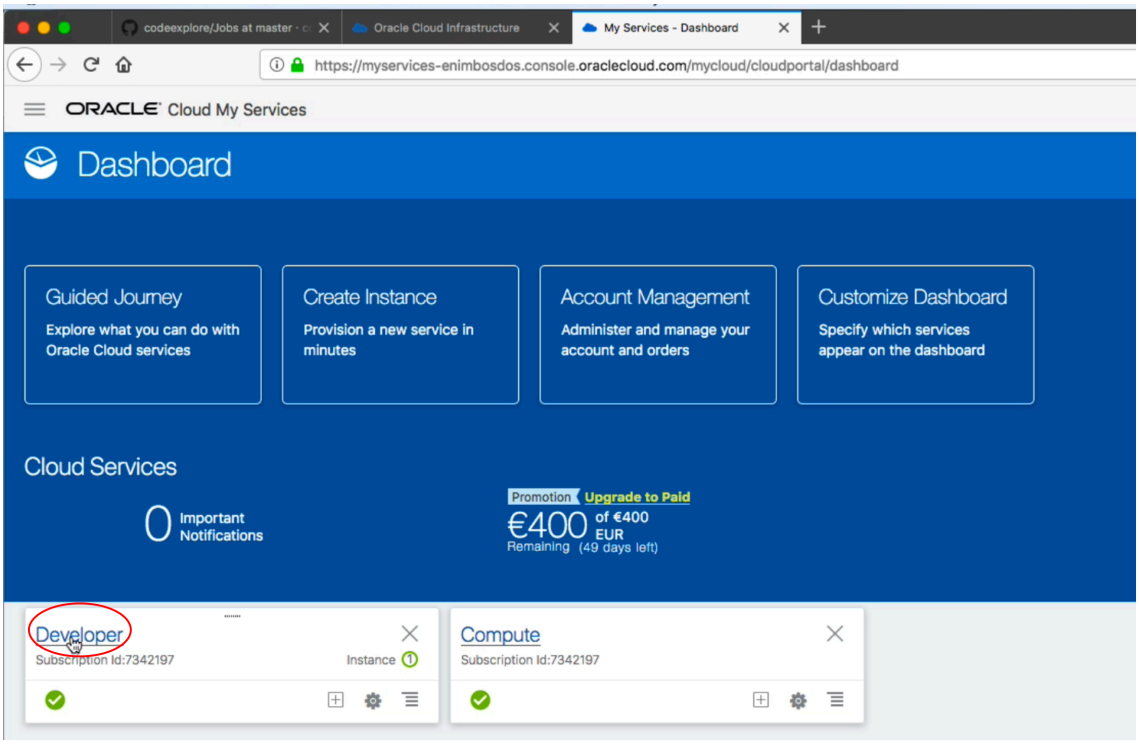


Y aparecerá la lista vacia de servicios ATP.

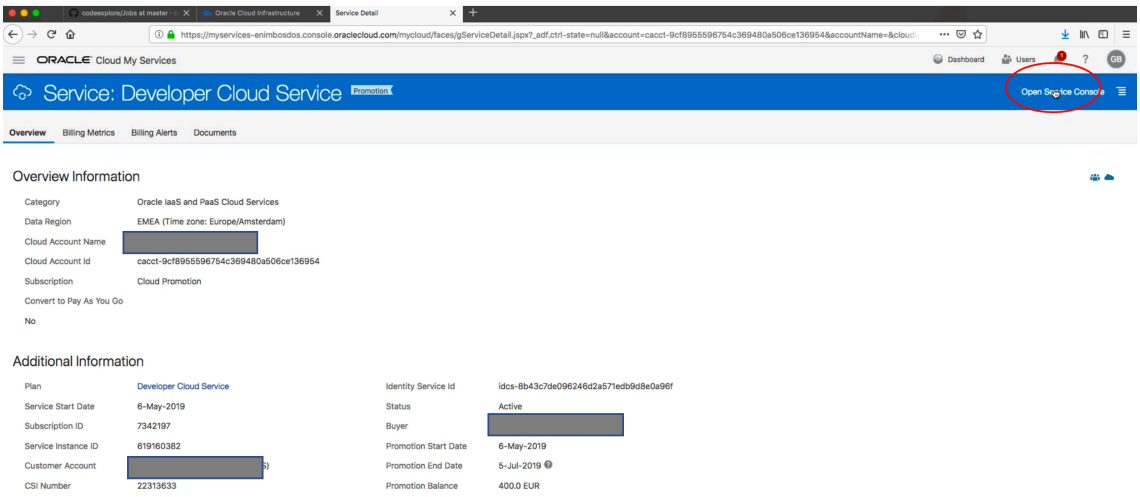


Configuración de script de creación de servicio ATP

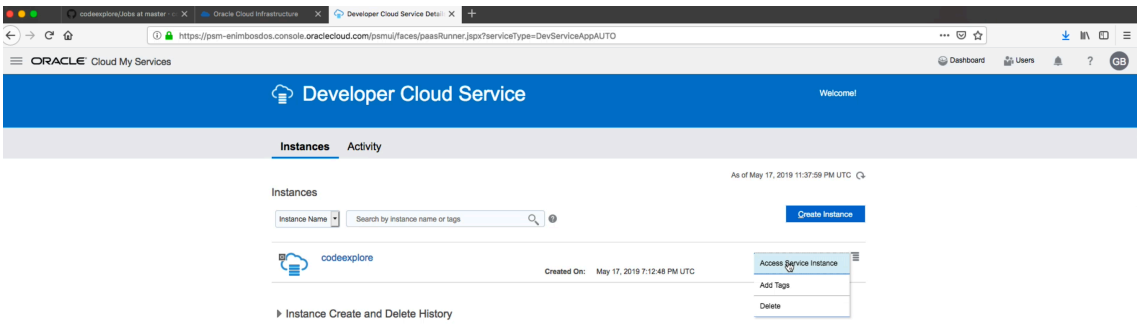
Acceder al servicio “Developer”



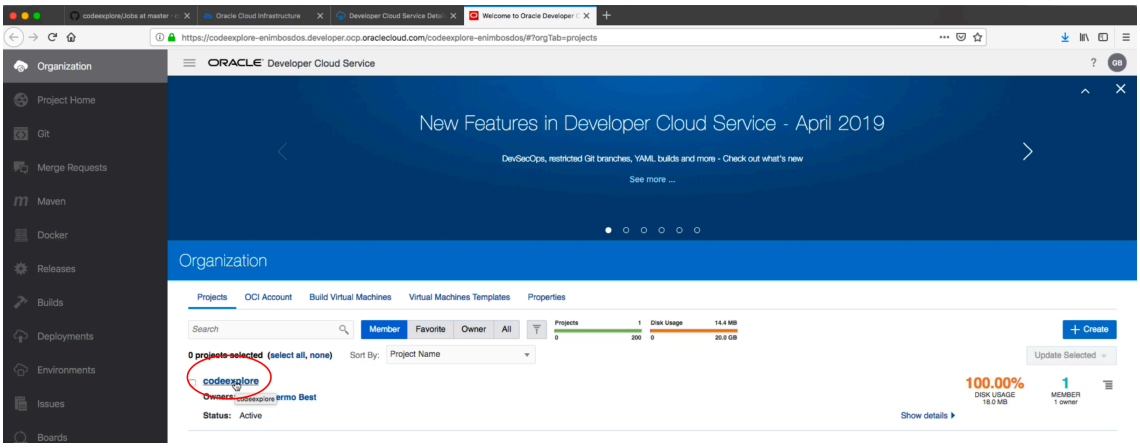
Y luego a la consola del servicio.



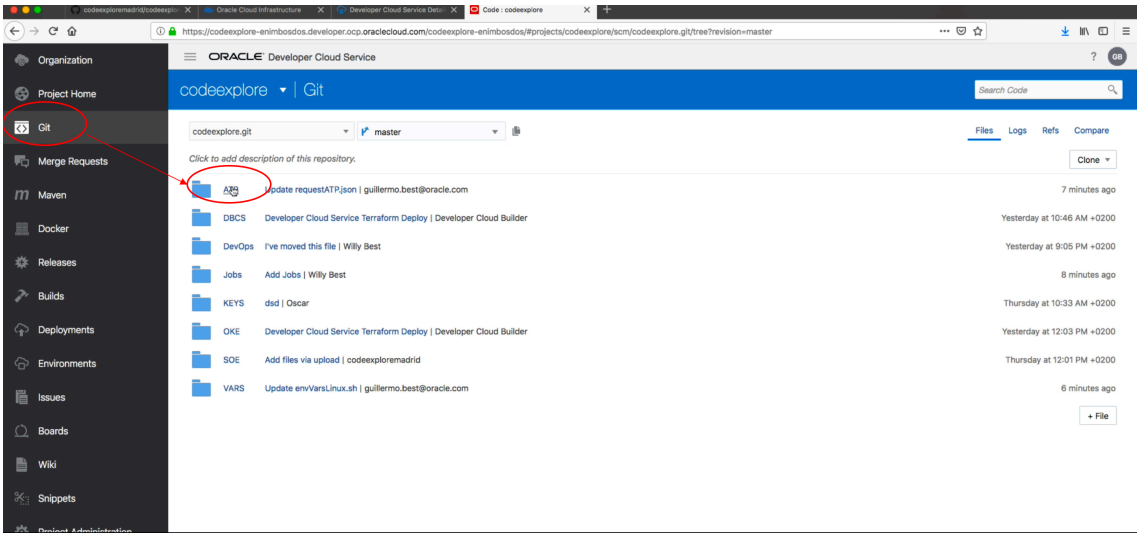
Y por último a la instancia del servicio.



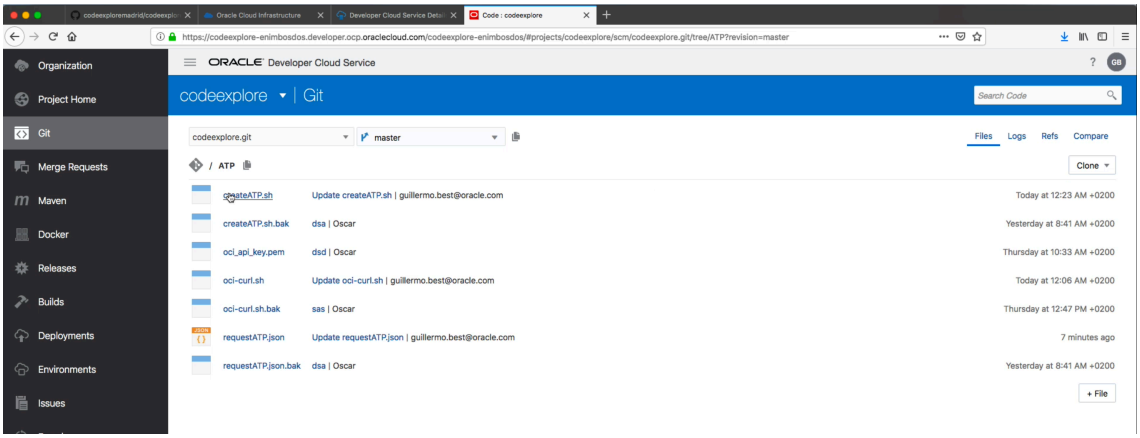
Acceder al proyecto “codeexplore”



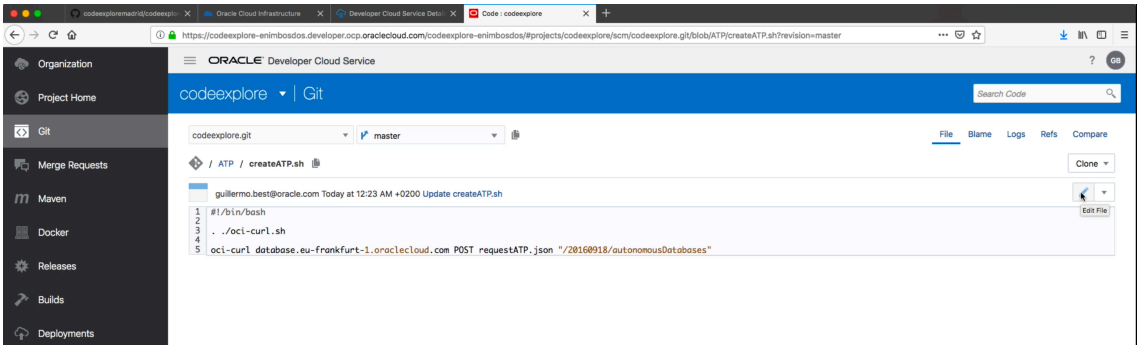
Acceder a la carpeta de “Git”, donde están los ficheros que utilizaremos en el proceso de creación del servicio ATP.
Seleccionar la carpeta “ATP”



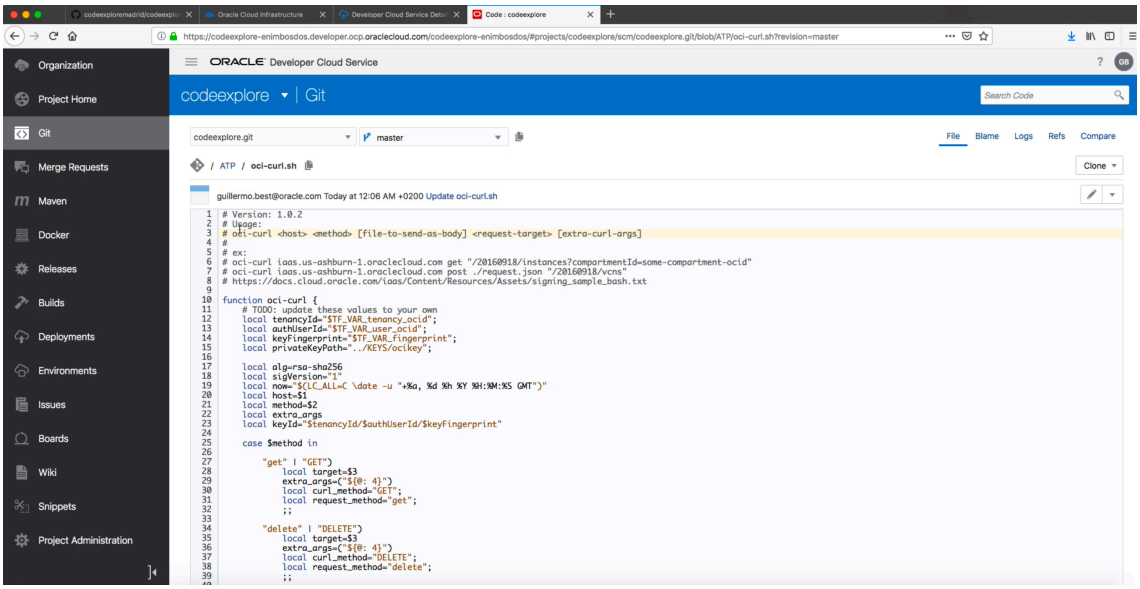
Los ficheros clave del ejercicio son “createATP.sh”, “oci-curl.sh” y “requestATP.json”
En las próximas instrucciones cambiaremos la configuración de los mismos para crear el servicio.



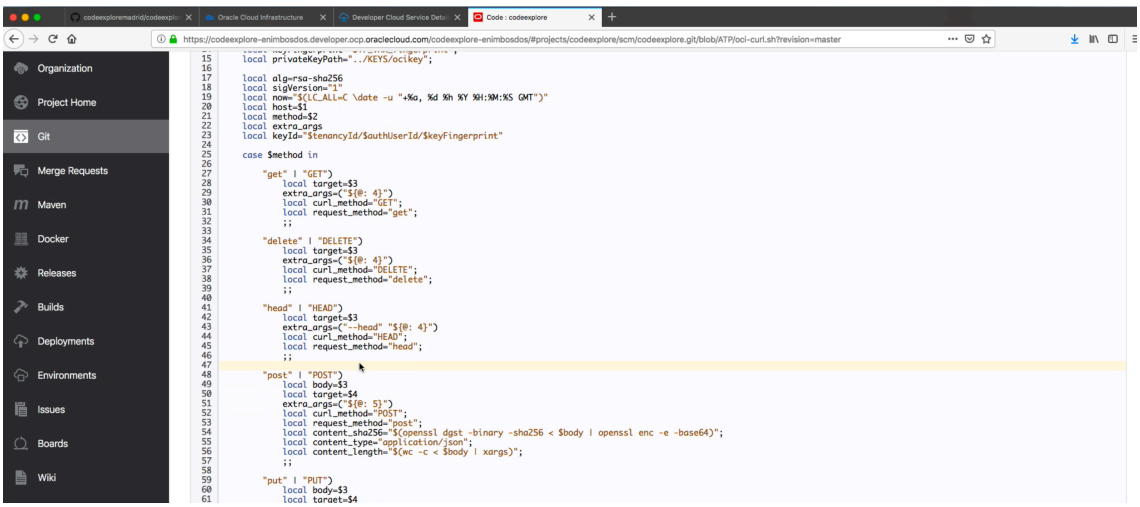
Hacer click en el fichero “createATP.sh” Como vemos, en dicho fichero se invocará al script “oci-curl.sh” No es necesario realizar ningún cambio en este fichero.



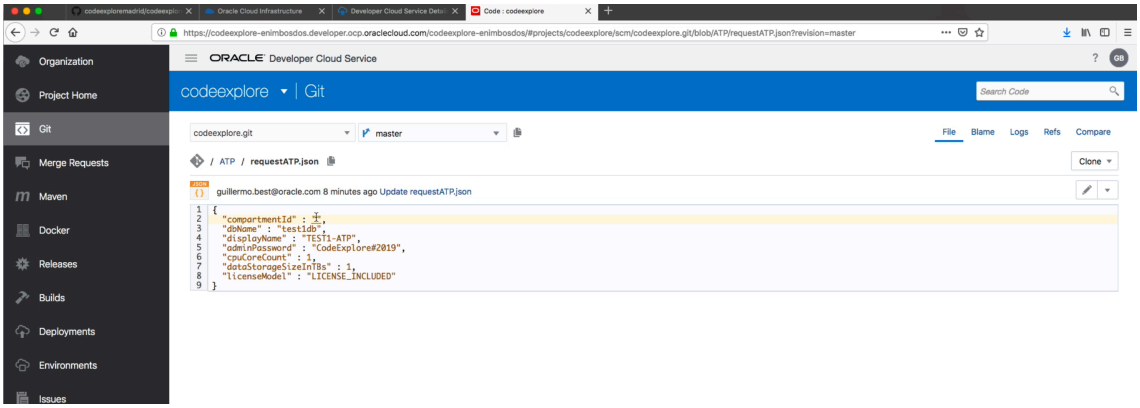
El segundo fichero es “oci-curl.sh” (será invocado por el “createATP.sh”)



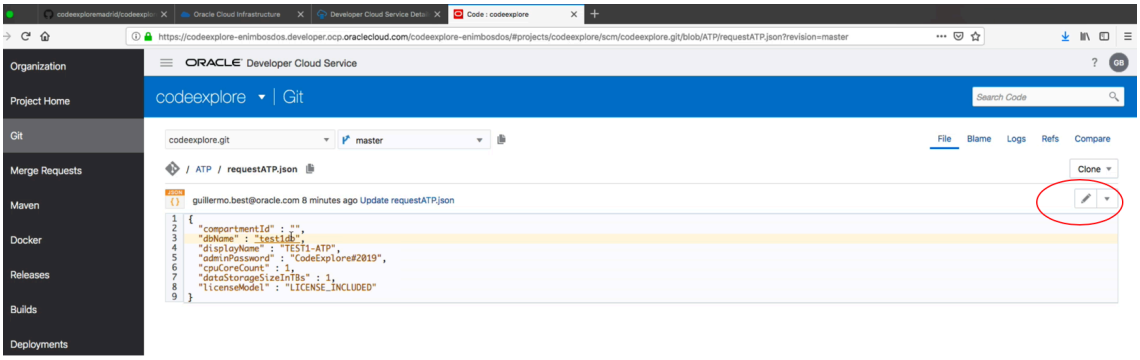
Podemos ver en dicho fichero el método “POST” que será el invocado en el script. No es necesario realizar ningún cambio en dicho fichero tampoco.



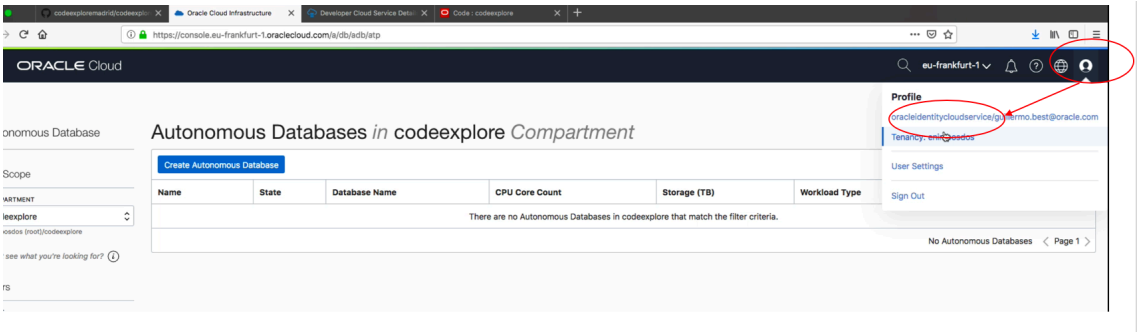
Hacer click en el fichero “requestATP.json”. En este fichero sí deberemos realizar algunos cambios.



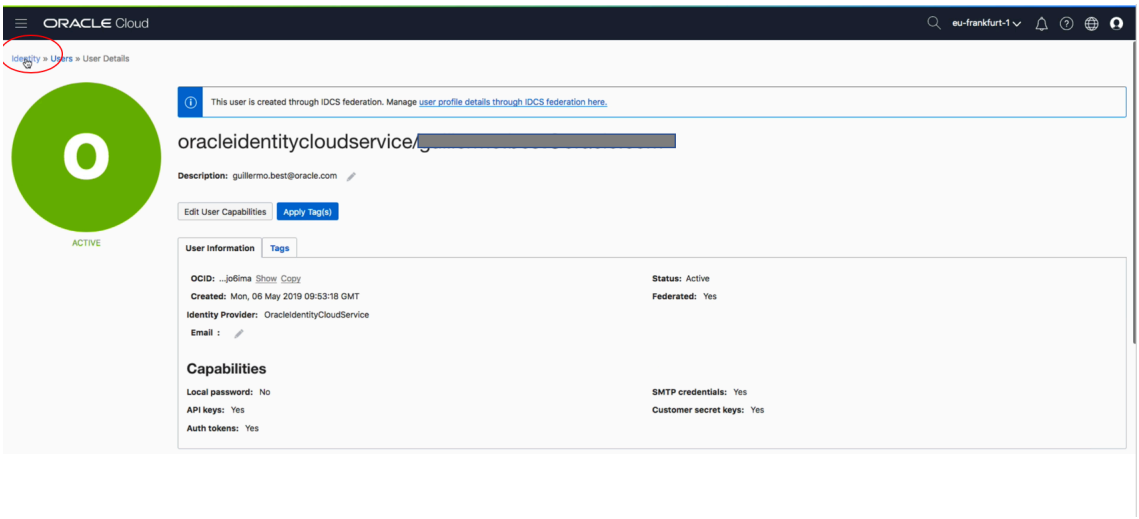
Seleccionar el icono para editar el fichero. El objetivo es incluir el parámetro “compartmentId”



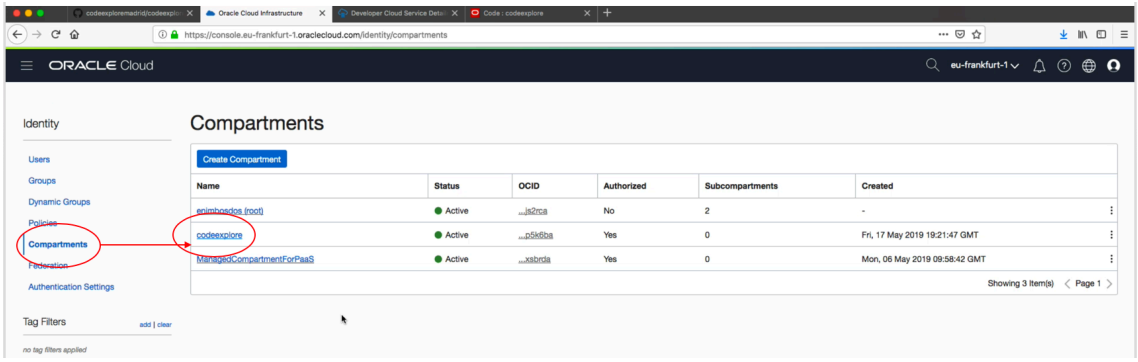
Para acceder a dicho parámetro debemos navegar a la pantalla de configuración del servicio.



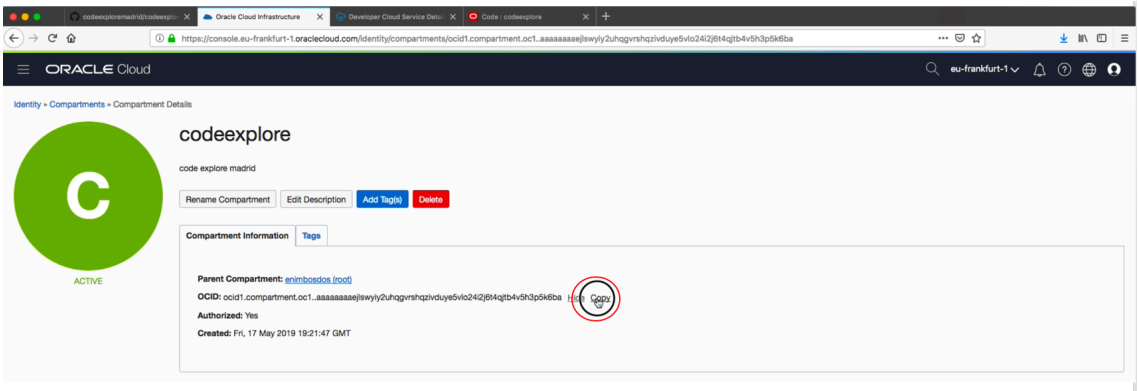
Seleccionar el link de “Identity”



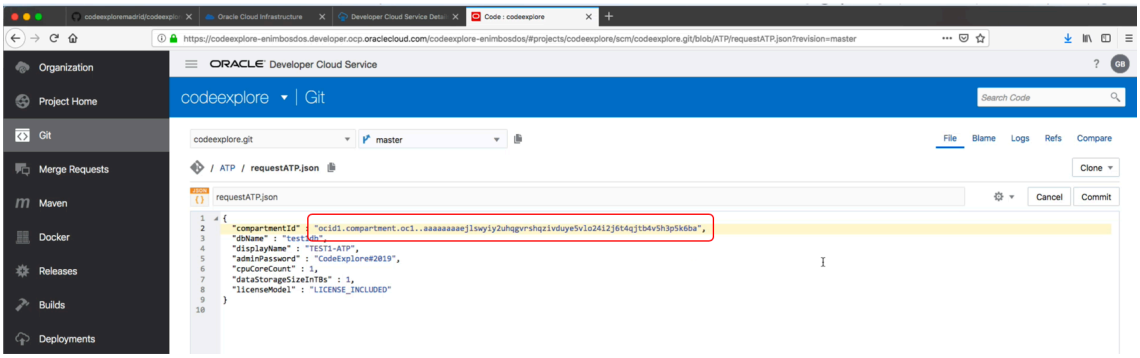
Ir al enlace “Compartments” y seleccionar el compartment “codeexplore”



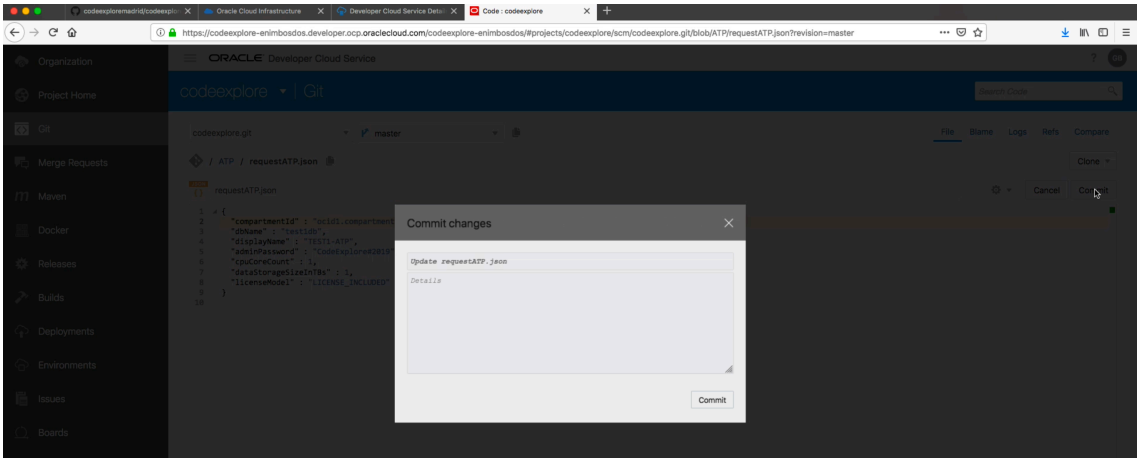
Para obtener el id del compartment clickar en “Show” y luego “Copy”



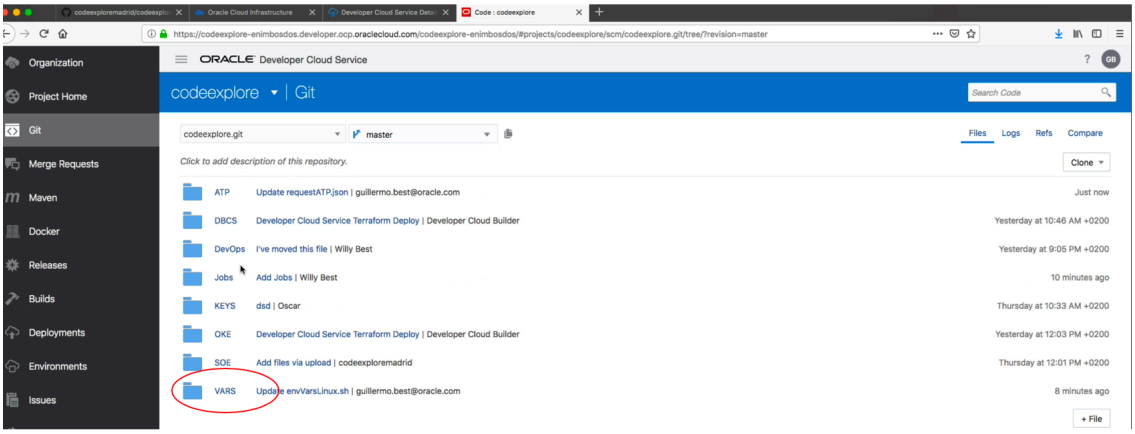
Pegar en el script JSON tal como se muestra en la figura. Después clickar en “Commit”
IMPORTANTE: este identificador lo necesitaremos posteriormente, por lo que se recomienda salvarlo en un fichero.



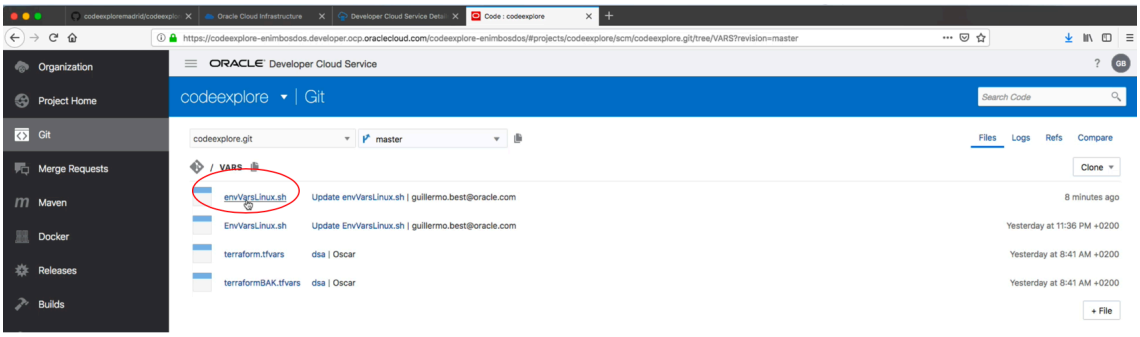
Ejecutar el commit.



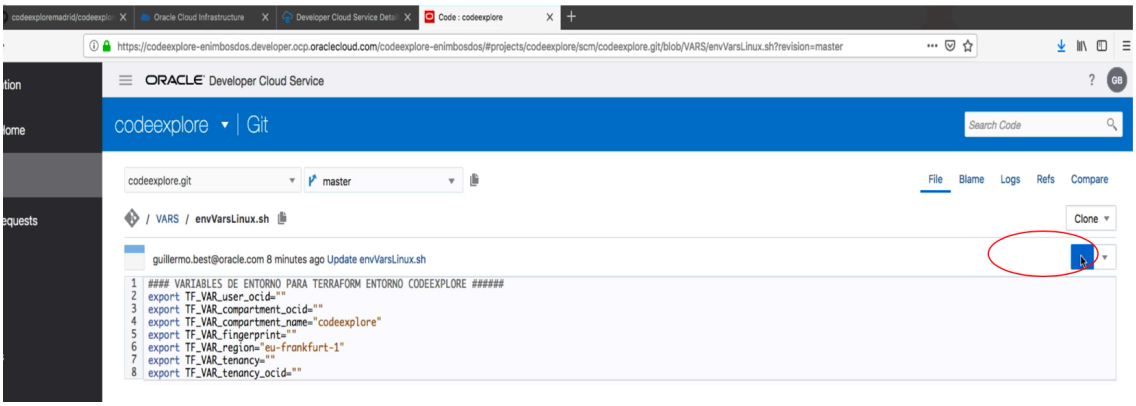
El siguiente paso consiste en realizar algunas acciones de configuración sobre las variables de entorno. Seleccionar en la pestaña de Git, la varpeta “VARS”



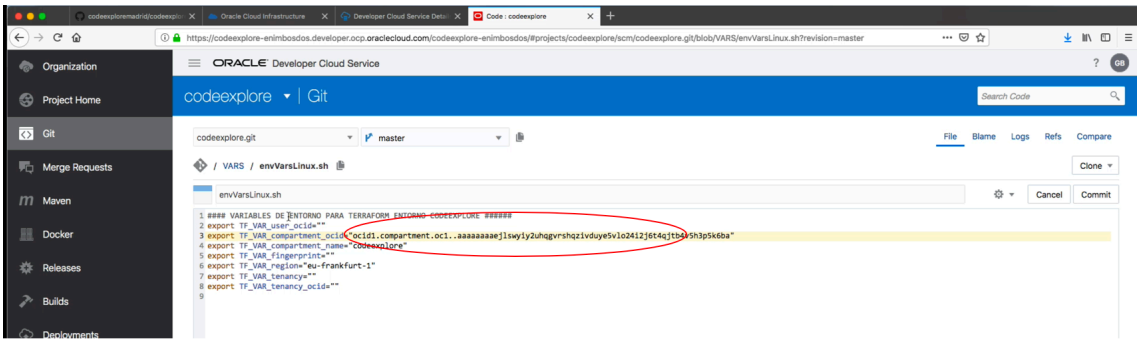
Editar el fichero “envVarsLinux.sh” **IMPORTANTE: el que comienza por minúscula**.



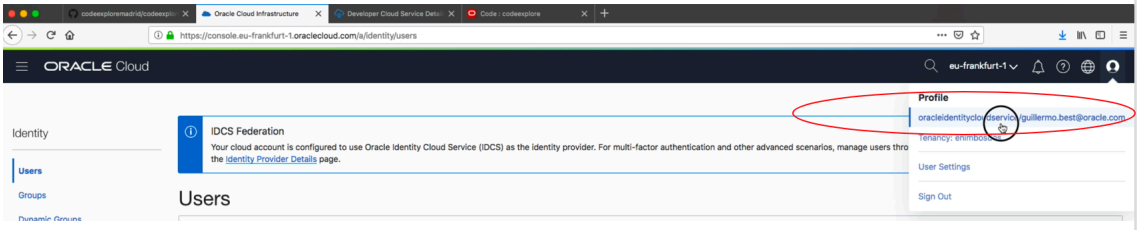
Seleccionar la acción de editar.



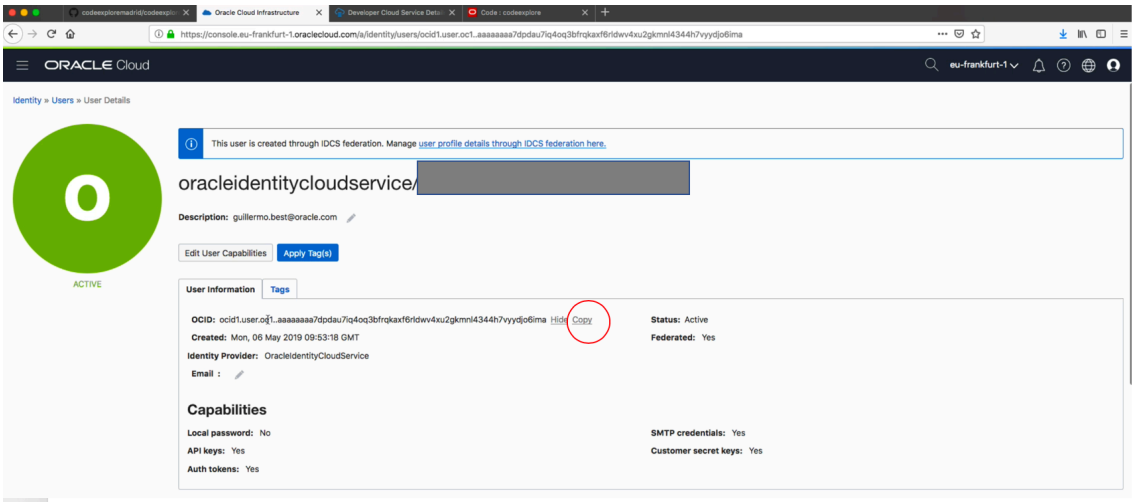
Introducir en primer lugar el identificador del compartimento que conseguimos anteriormente.



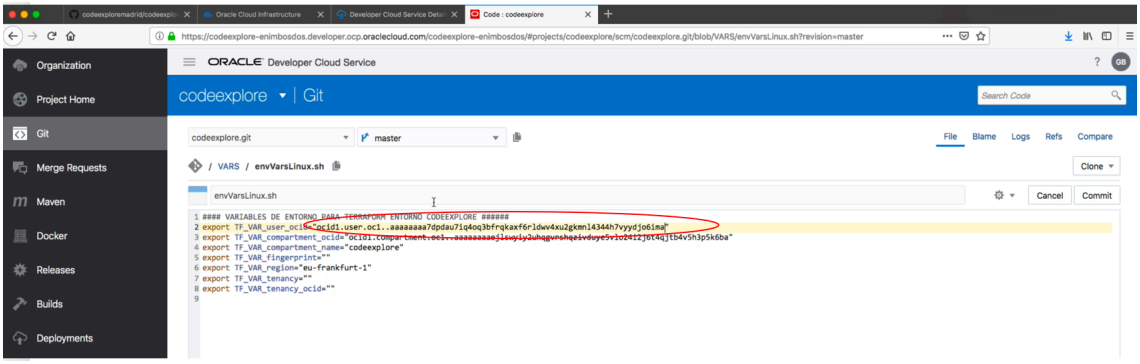
A continuación introduciremos el “user_ocid”. Para acceder al mismo, debemos navegar a la pantalla de variables de entorno.



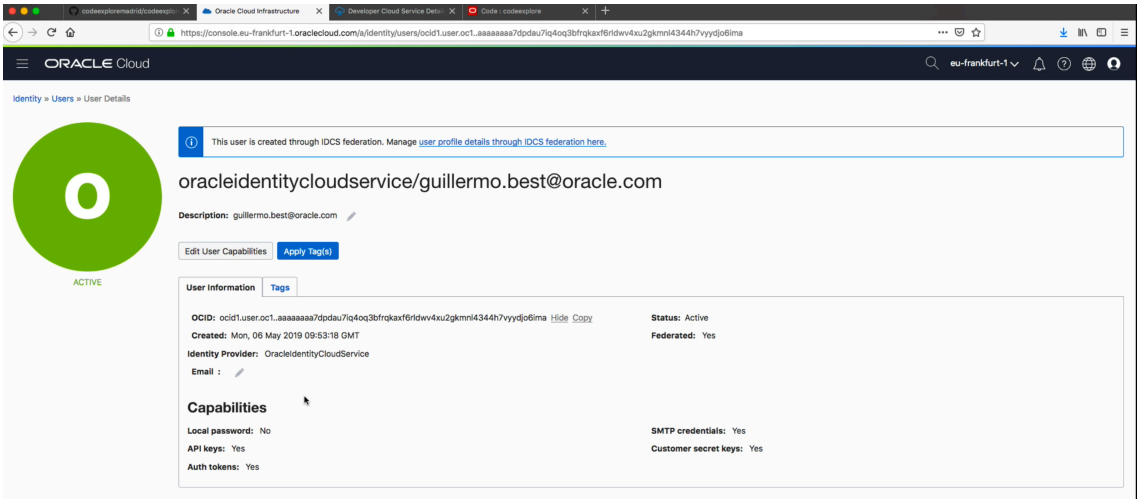
Navegar a Identity > Users > User Details y copier el parámetro “Show > Copy” tal como se muestra en pantalla.



Pegamos el dato en el fichero de configuración.

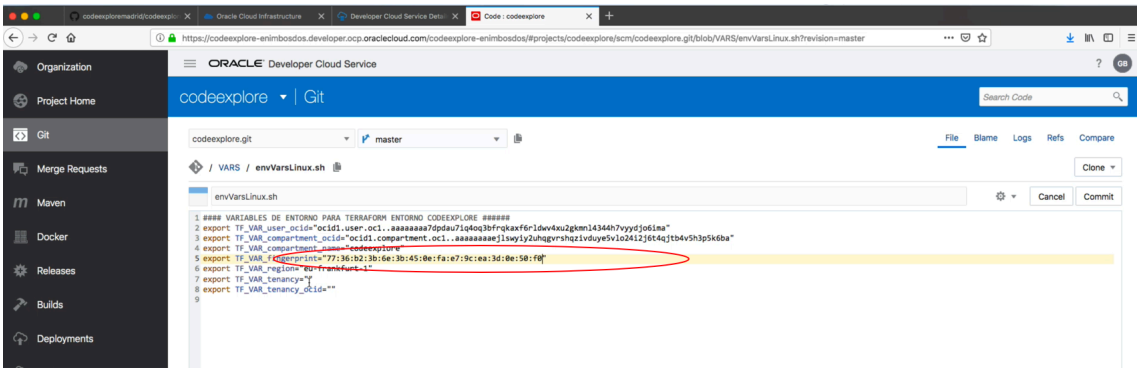


A continuación conseguiremos la variable “fingerprint” que puede conseguirse en la pantalla de detalles de usuario.

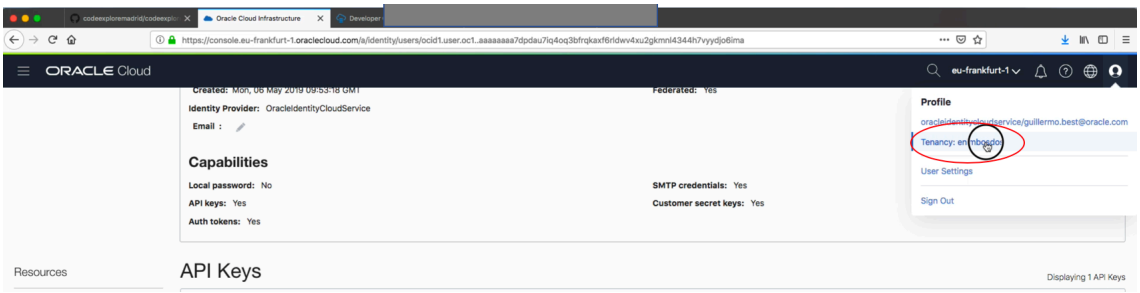


En dicha pantalla podemos conseguir el parámetro “fingerprint”

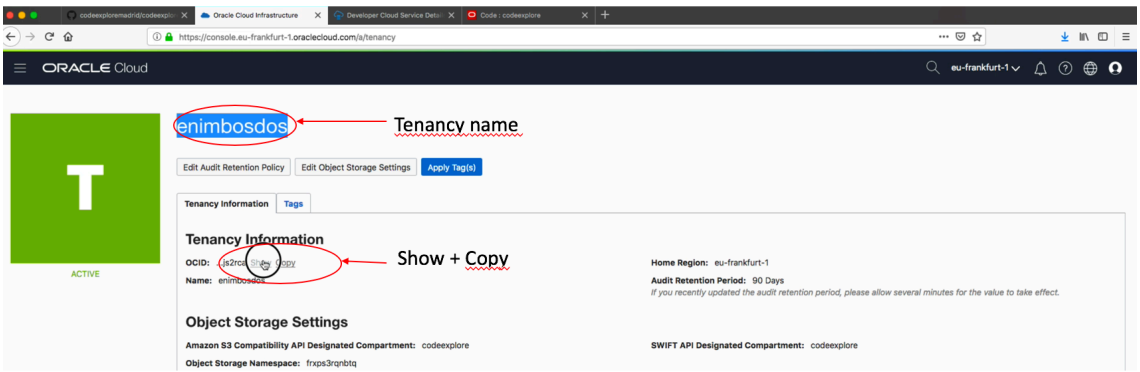
Pegar el parámetro en el fichero.



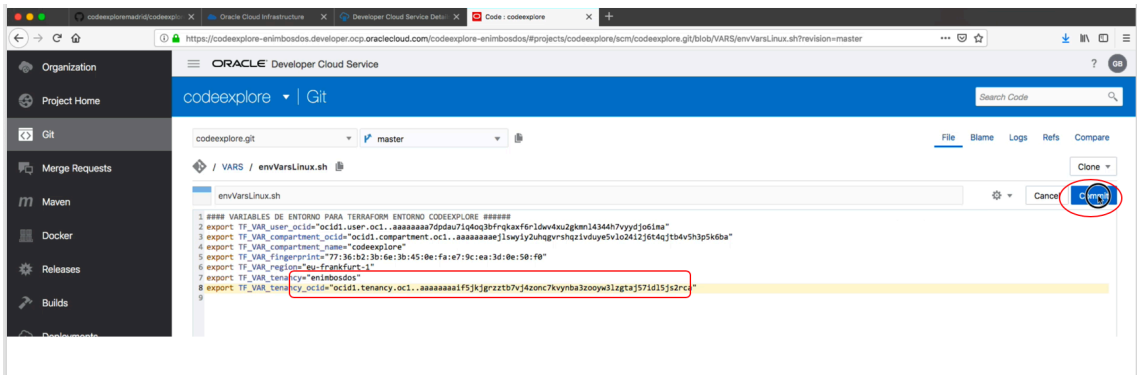
Los dos últimos parámetros (“tenancy” y “tenancy_ocid”) los obtendremos de la pantalla de configuración tal como se muestra en la imagen.



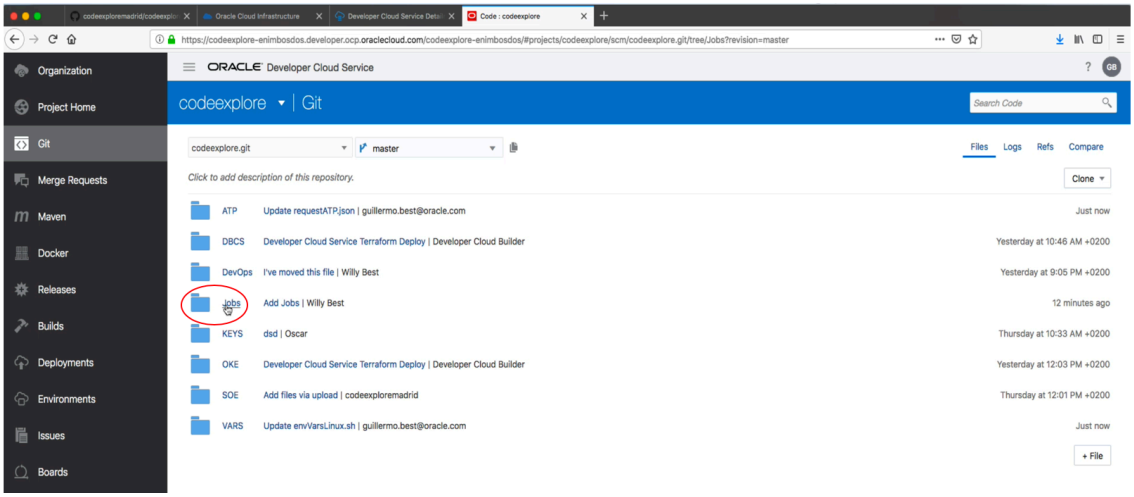
Obtener los datos tal como se muestra en la figura.



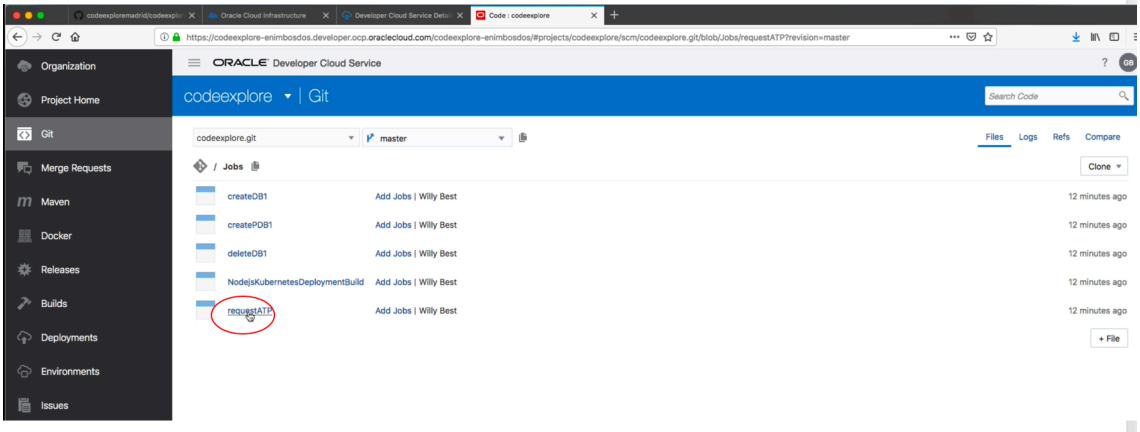
Incluir ambos parámetros y ejecutar “Commit”



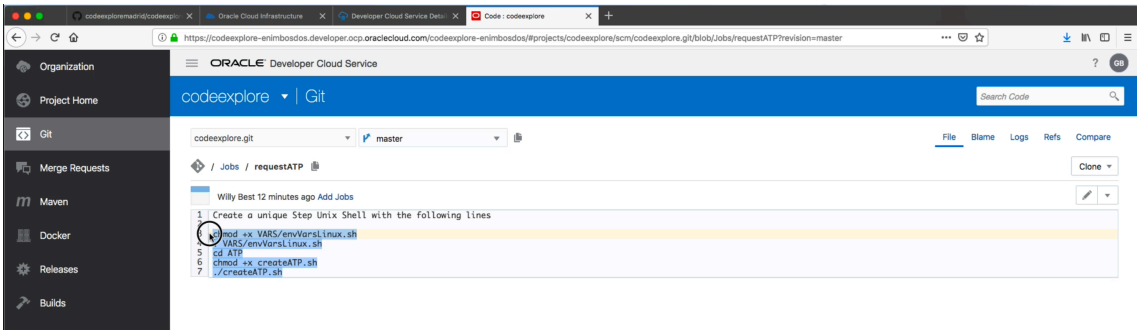
Por último ir a la carpeta “Jobs”



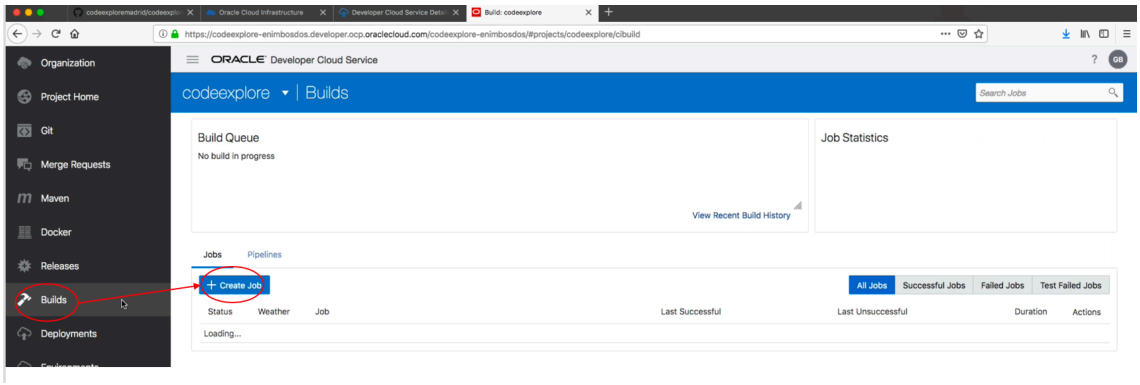
Seleccionar el job “Request ATP”



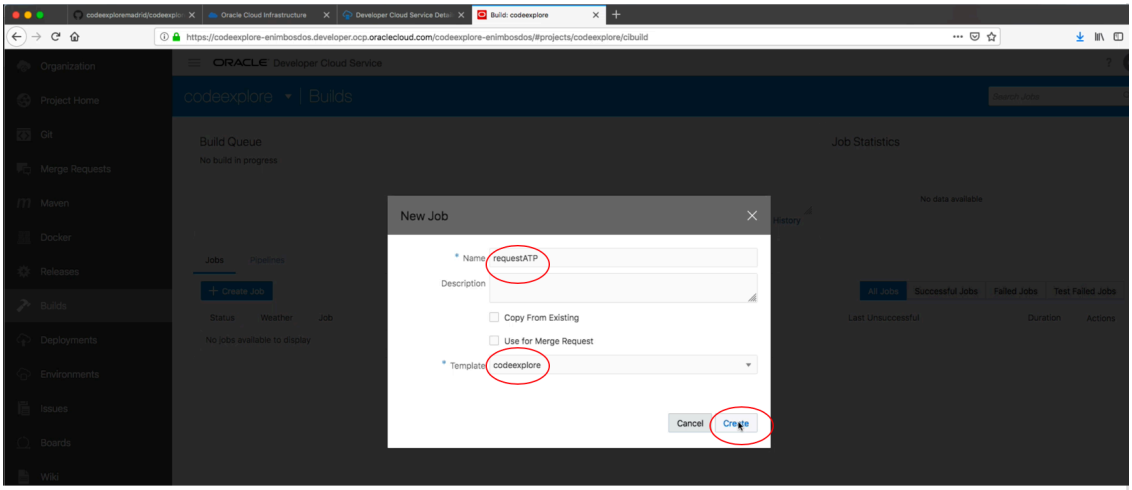
Vamos a crear un script Unix tomando como base la información de dicho Job. Seleccionar todas las líneas del fichero.



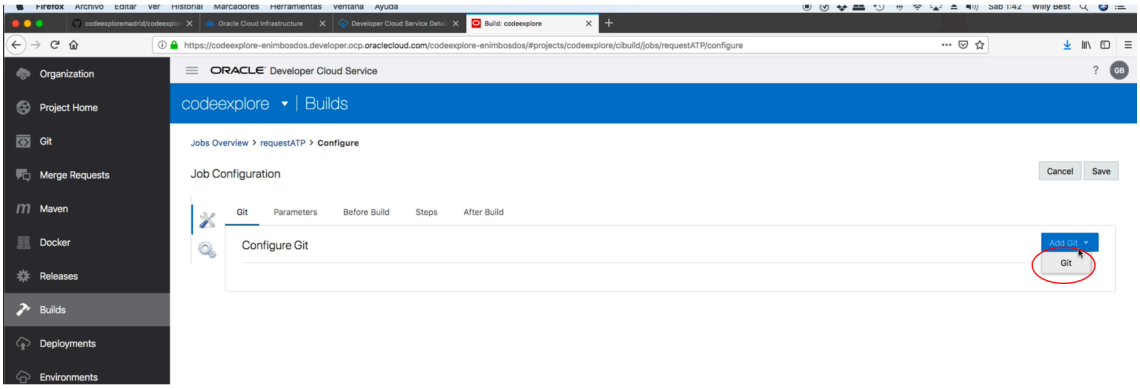
Ir a la carpeta “Builds” y creamos un “Job”



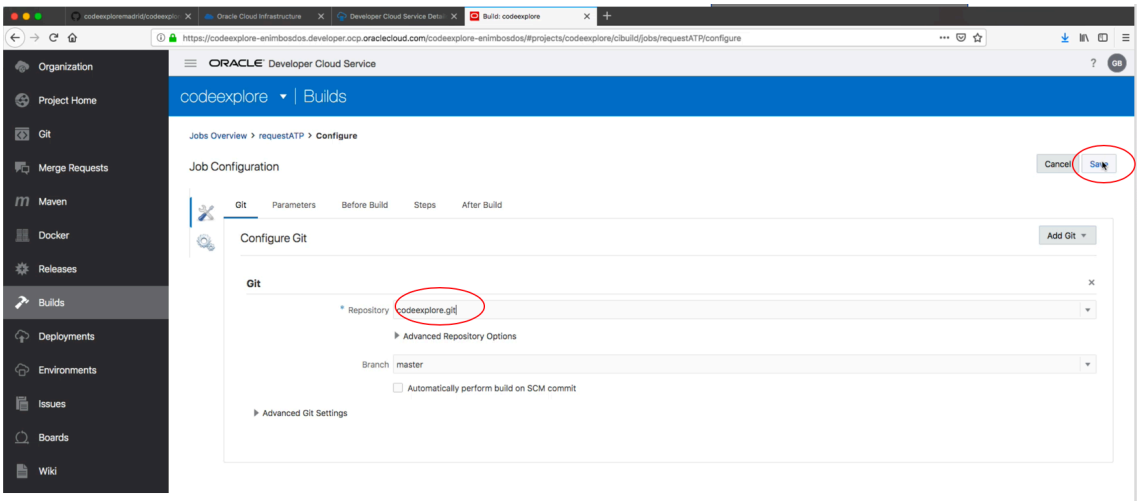
Introducir los parámetros tal como se muestra en la figura y pulsar “Create”



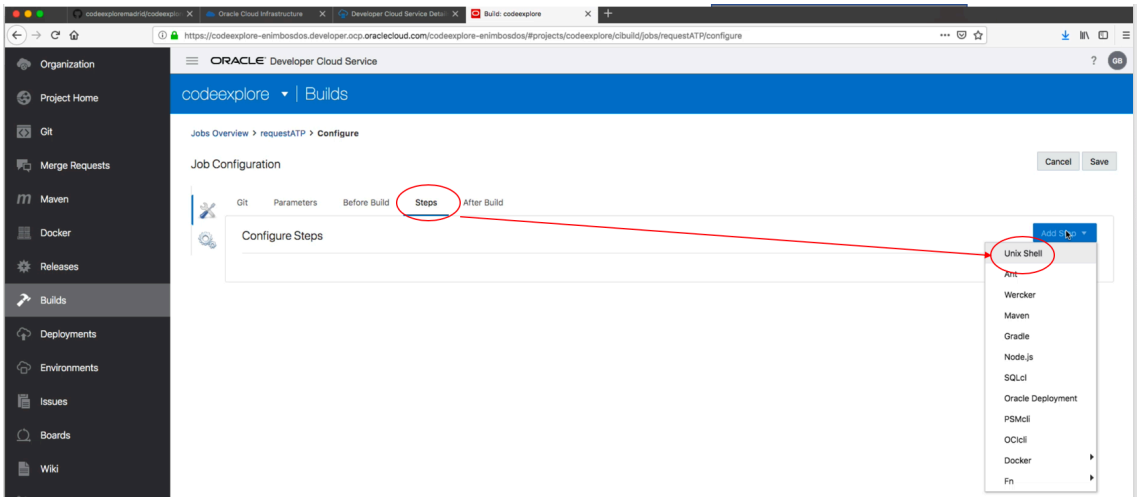
Configurar como opción de obtención de información el reositorio “Git”



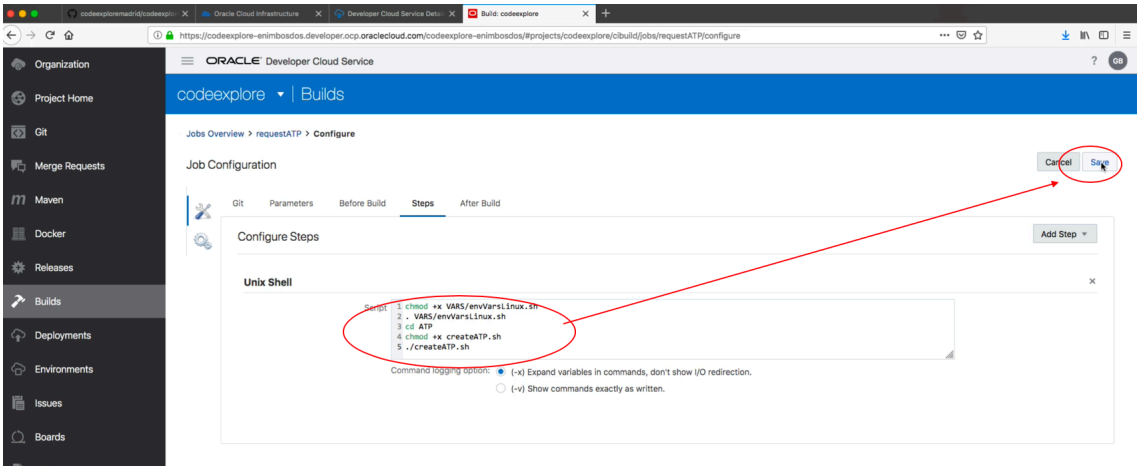
Seleccionar el repositorio “codeexplore.git” y pulsar “Save”



Es necesario añadir un paso adicional de tipo “Unix shell”

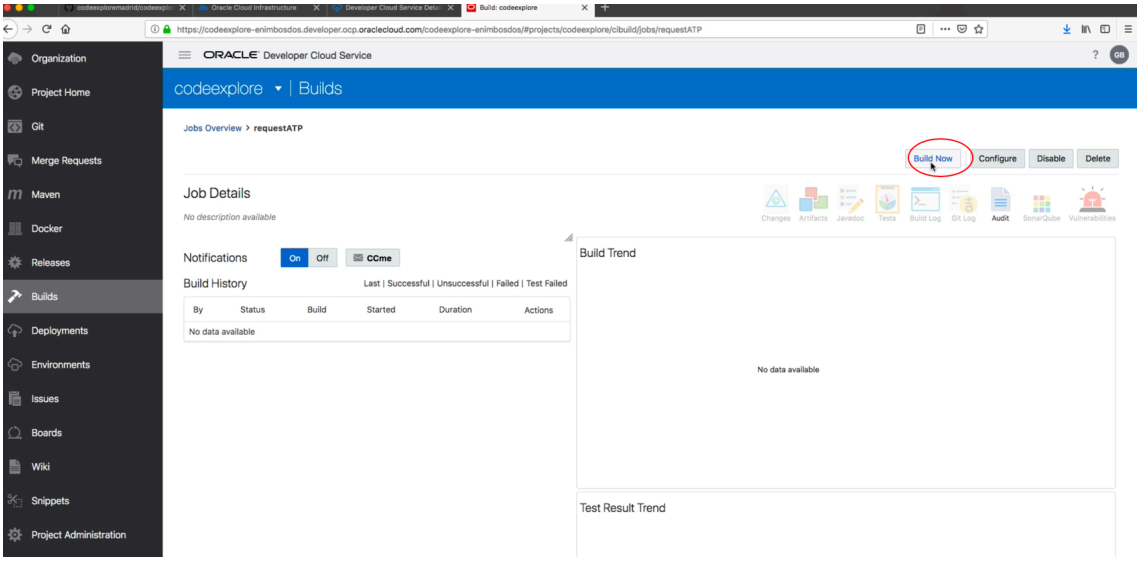


Introducir las líneas que copiamos previamente y puksar “Save”

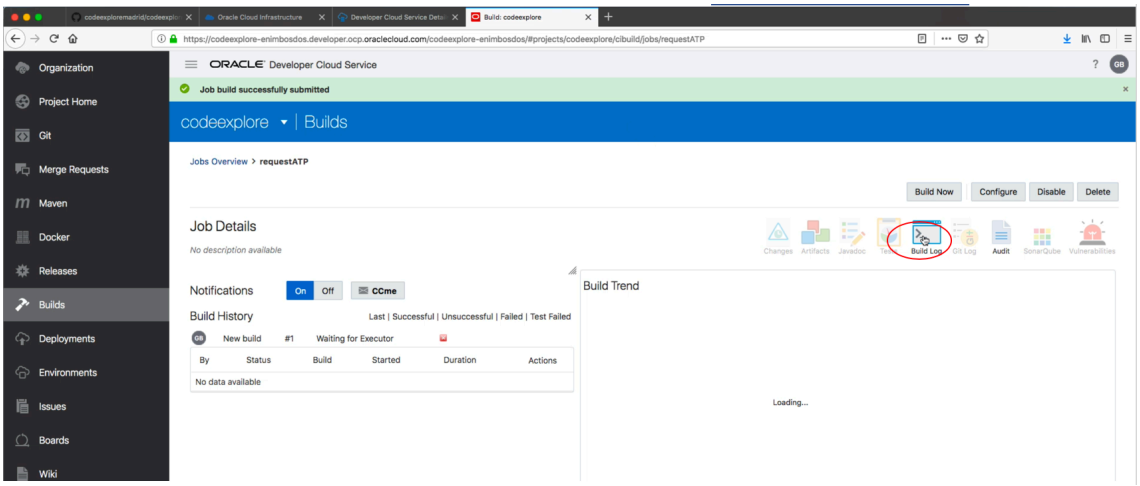


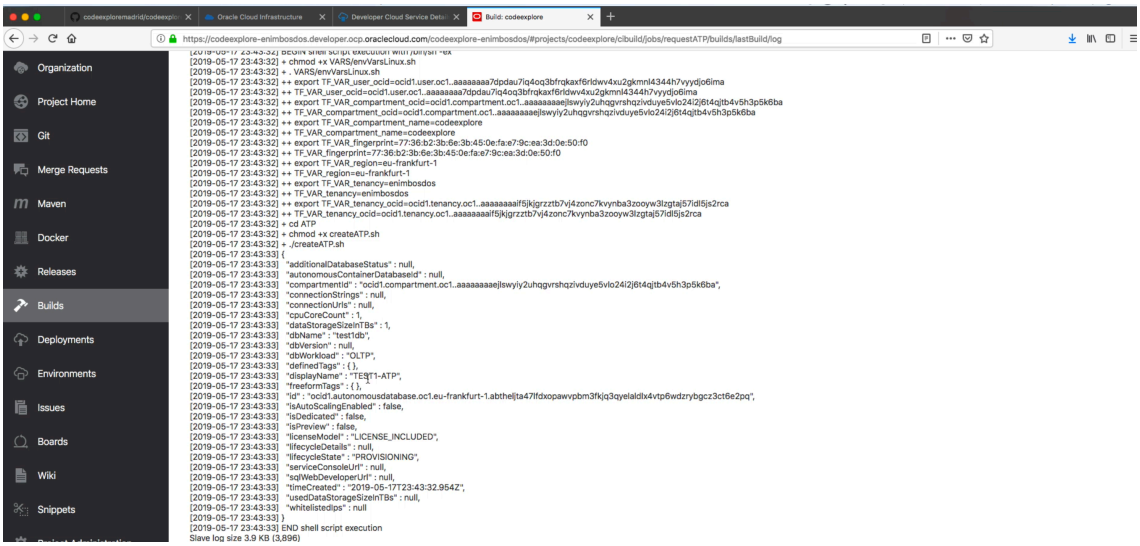
Ejecución del proceso de construcción del servicio ATP

Lanzar el proceso clickando en “Build Now”

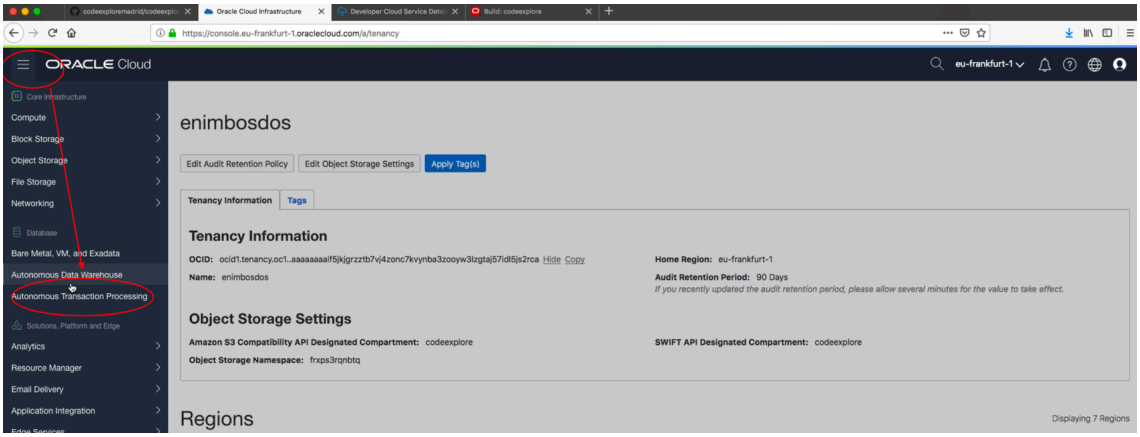


Puede consultarse el log de creación durante su ejecución.

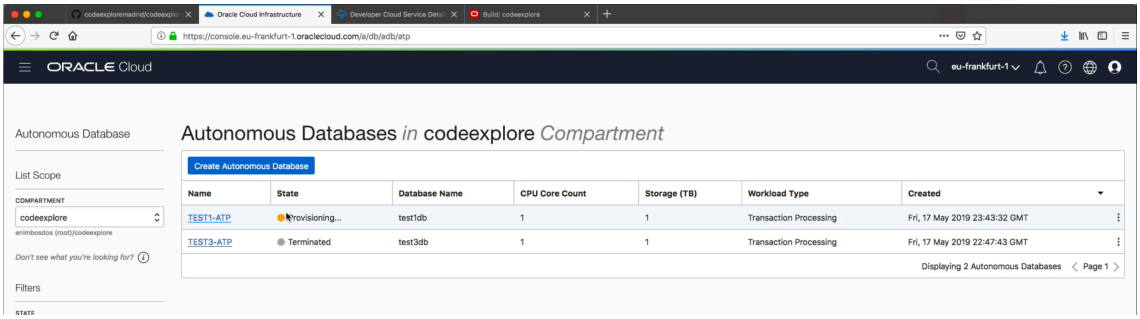




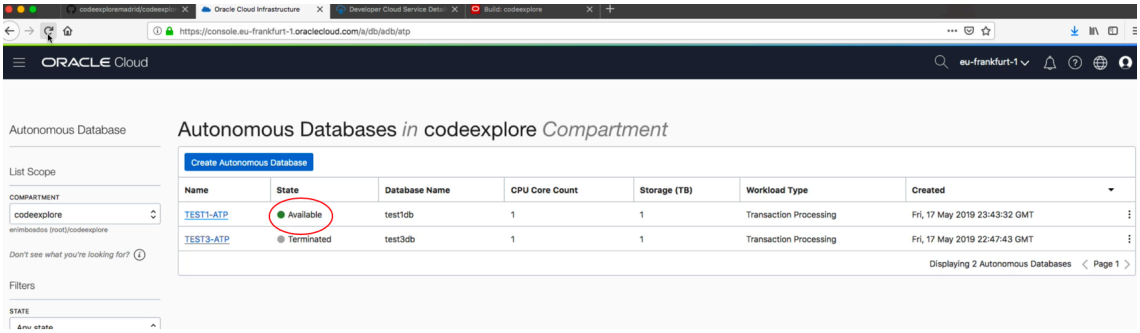
Puede comprobarse el estado del proceso navegando al catálogo de servicios.



Esta pantalla nos informará del estado del servicio.

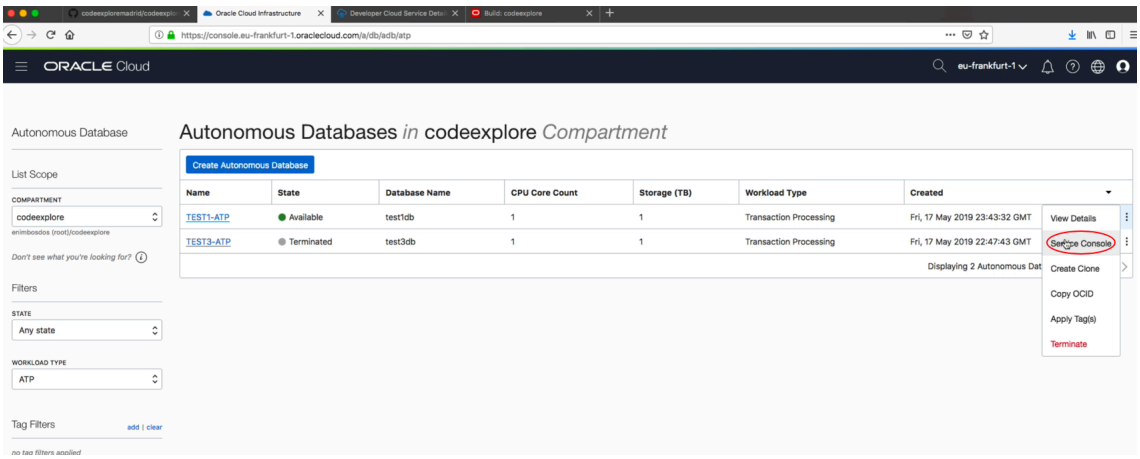


Finalizado el proceso, el servicio estará listo para ser utilizado.

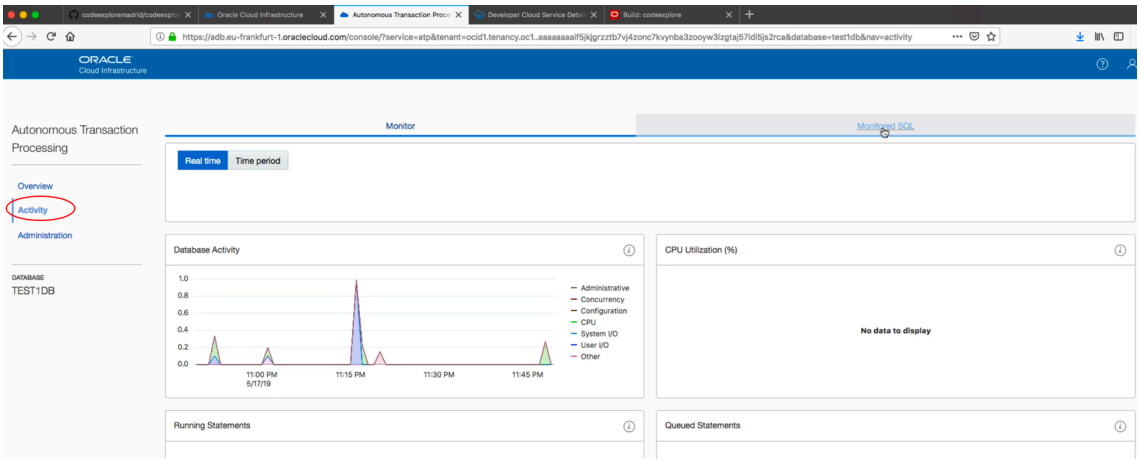


Utilización del servicio

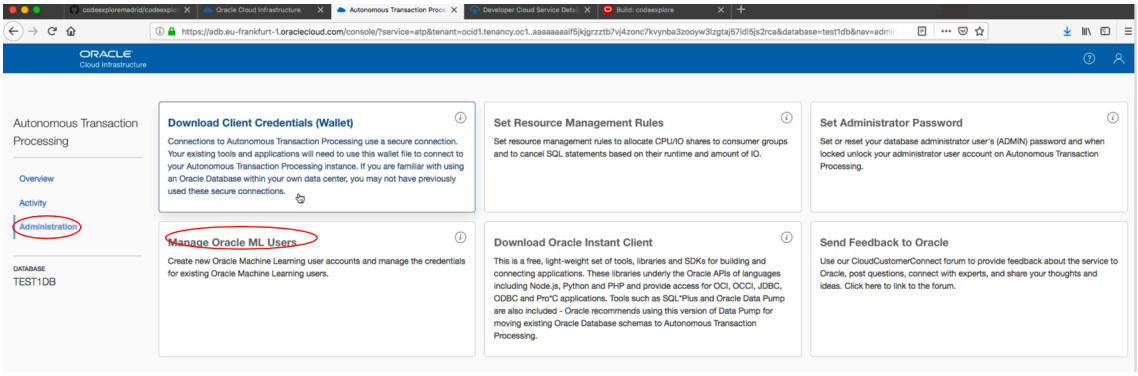
Accedemos en primer lugar a la consola del servicio.



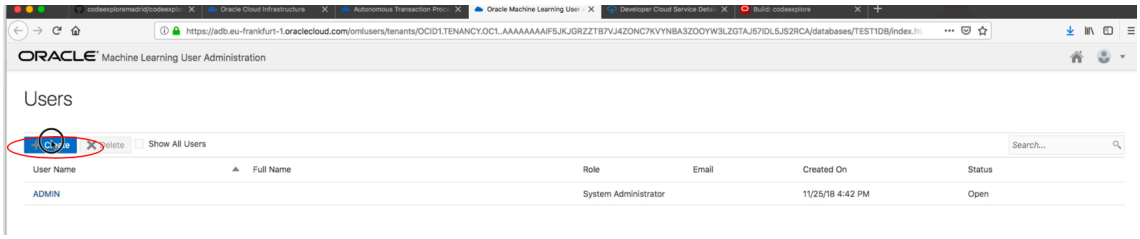
Podemos monitorizar la actividad del servicio.



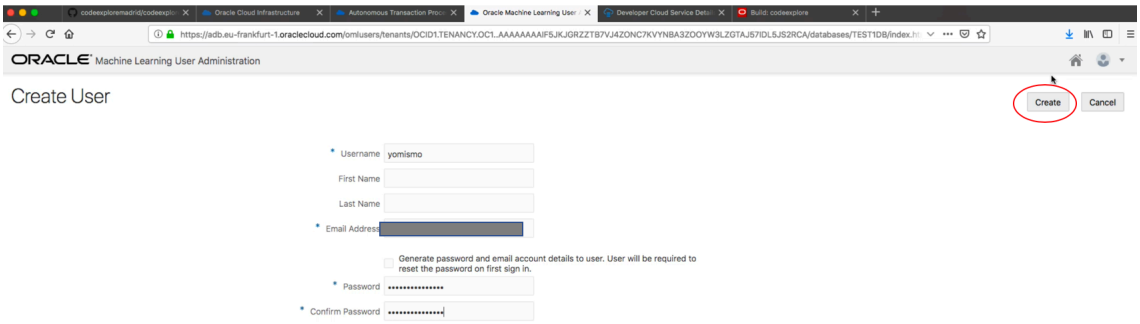
En la pestaña de administración tendremos una serie de opciones disponibles como se muestra en la figura. En nuestro caso crearemos un usuario con la opción “Manage Oracle ML Users”



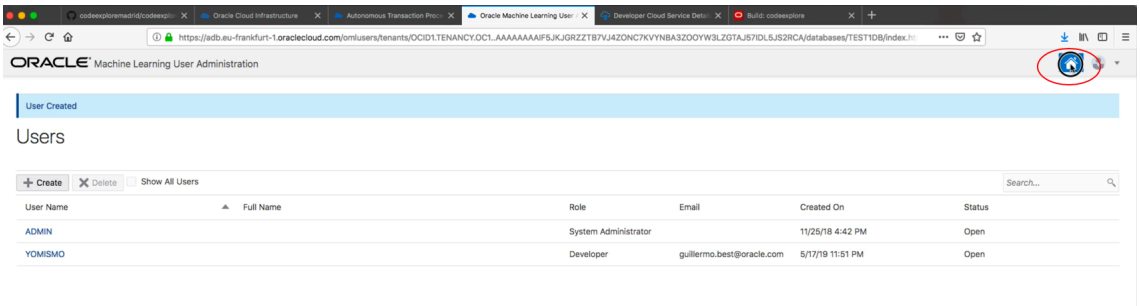
Seleccionamos la opción “Create”



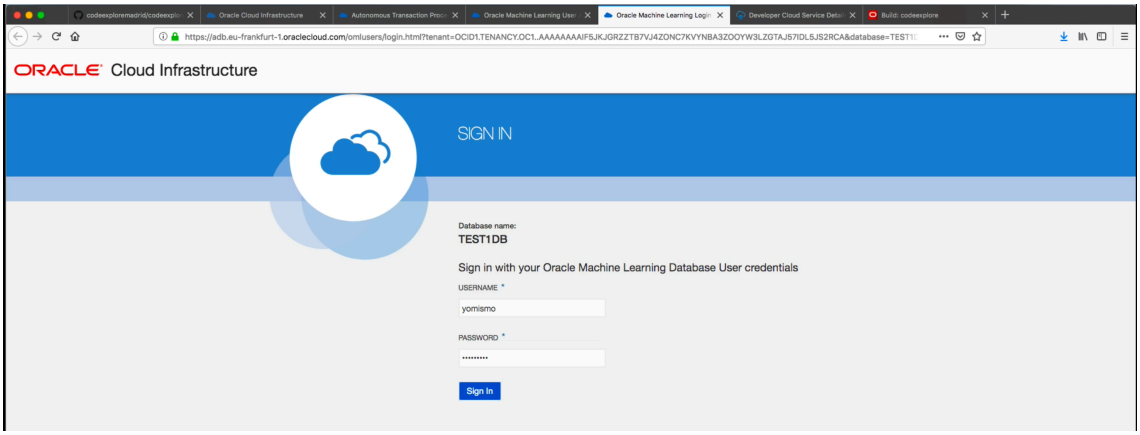
Introducimos las credenciales y seleccionamos la opción “Create”



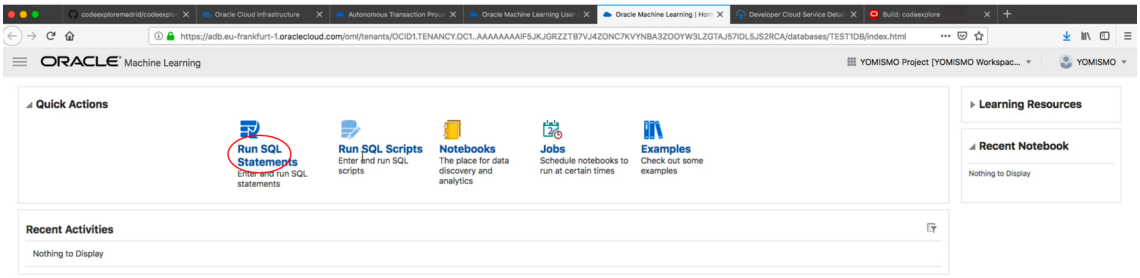
Cuando el proceso de creación ha finalizado, seleccionamos el icono de “Home” para logarnos con las credenciales del nuevo usuario.



Hacer login con el nuevo usuario.



En nuestro espacio de trabajo aparecerán las herramientas a nuestra disposición. Vamos a realizar la prueba de lanzar una sentencia SQL.



Podremos lanzar cualquier sentencia sobre la base de datos utilizando el notebook.