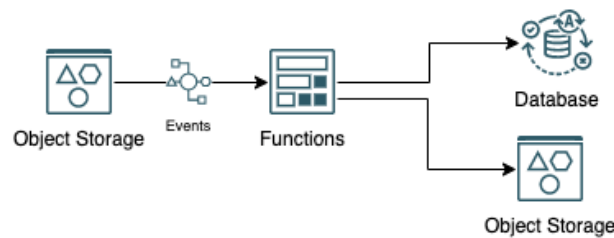


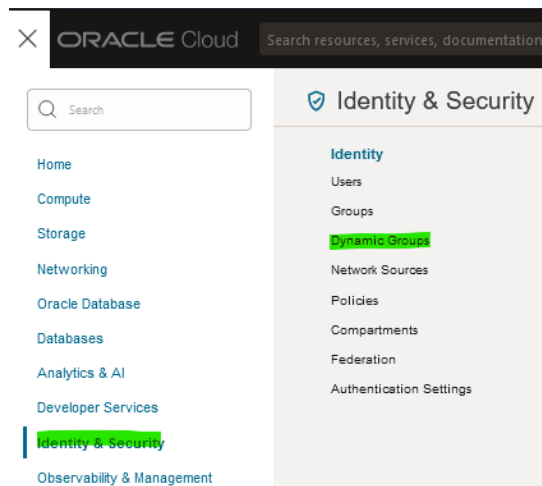
# LABORATORIO CONSTRUCCIÓN DE APLICACIONES ORACLE CLOUD SERVERLESS

Este laboratorio esta enfocado a que puedas construir una aplicación Serverless, que realice la lectura, procesamiento y guardado de un archivo separado por comas(\*.csv) en una base de datos autonoma, dicha aplicación tendra la siguiente aquitectura:



## PRE-REQUISITOS

1. Creación de grupo dinámico con el nombre **FunctionGroup** y políticas de seguridad para manipulación de la infraestructura OCI por parte de la función serverless.



### Create dynamic group

Name  
**FunctionGroup**

The only characters allowed are letters and numbers (for example, a-z, A-Z, 0-9), an

Description  
FunctionGroup

#### Matching rules

Rules define what resources are members of this dynamic group. All automatically.

**Example:** Any {instance.id = 'ocid1.instance.oc1.iad..example'ocid1.compartment.oc1..exampleuniqueid2'}

☒ Match any rules defined below ☐ Match all rules defined

Rule 1  
**ALL {resource.type = 'fnfunc'}**

Create Cancel

Estableciendo la siguiente regla para el grupo dinámico que permitirá a las funciones serverless acceder a los recursos OCI, a nivel de seguridad

**ALL {resource.type = 'fnfunc'}**

- Definición de políticas IAM para la manipulación de la infraestructura OCI por parte de la función serverless.

**IMPORTANTE:** Estas políticas deben ser definidas a nivel del compartiment **ROOT**

**Oracle Cloud** Search resources, services, documentation

Search

**Identity & Security**

- Identity
  - Users
  - Groups
  - Dynamic Groups
  - Network Sources
  - Policies**
  - Compartments
  - Federation
  - Authentication Settings

**Create Policy**

Name: FunctionPolicies

Description: Permisos Serverless

Compartment: cxmleammcrrn (root)

**Policy Builder** Show manual editor ☒

- Allow dynamic-group **FunctionGroup** to manage functions-family in tenancy
- Allow dynamic-group **FunctionGroup** to use virtual-network-family in tenancy
- Allow dynamic-group **FunctionGroup** to manage repos in tenancy
- Allow dynamic-group **FunctionGroup** to inspect object-family in tenancy
- Allow dynamic-group **FunctionGroup** to manage objects in tenancy
- Allow dynamic-group **FunctionGroup** to manage autonomous-database-family in tenancy

Create Cancel ☐ Create Another Policy

Allow dynamic-group **FunctionGroup** to manage functions-family in tenancy

Allow dynamic-group **FunctionGroup** to use virtual-network-family in tenancy

Allow dynamic-group **FunctionGroup** to manage repos in tenancy

Allow dynamic-group **FunctionGroup** to inspect object-family in tenancy

Allow dynamic-group **FunctionGroup** to manage objects in tenancy

Allow dynamic-group **FunctionGroup** to manage autonomous-database-family in tenancy

Allow dynamic-group **FunctionGroup** to use ons-topics in tenancy

Allow dynamic-group **FunctionGroup** to use nosql-family in tenancy

Allow dynamic-group **FunctionGroup** to manage nosql-tables in tenancy

Allow dynamic-group **FunctionGroup** to manage nosql-rows in tenancy

## TOPIC PARA NOTIFICACIONES

En el menu general debemos ir al menú de notificaciones, y crear un topic con el nombre de empresa  
Ejemplo: **ACME CORP**

The image shows two parts of the Oracle Cloud interface. On the left is the 'Developer Services' menu with a search bar and a list of services including Home, Compute, Storage, Networking, Oracle Database, Databases, Analytics & AI, and Developer Services (highlighted). On the right is the 'Create Topic' form. The form has a 'Name' field with the placeholder 'NOMBRE-MI-COMPANIA', a 'Description' field, and a 'Create' button. A note states: 'Once the topic is created, an admin access.'

**Copia el OCID del topic lo necesitaras más adelante**

Dentro del TOPIC se deberá crear una suscripción al correo del **COMISARIO DE CARRERA** ([nataly.diaz@oracle.com](mailto:nataly.diaz@oracle.com)), a este correo llegarán las notificaciones que serán fundamental para los **puntajes y clasificación.**

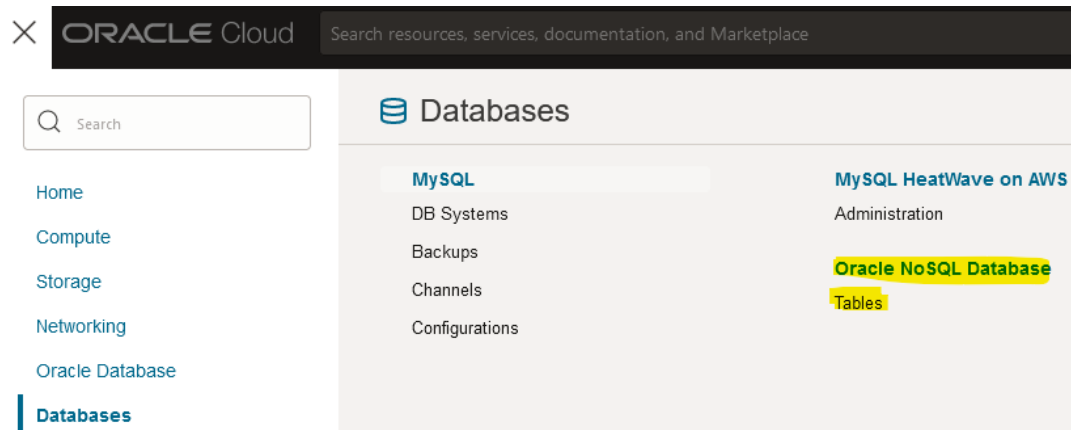
The image shows the 'Create Subscription' form. It has a 'Protocol' dropdown set to 'Email', an 'Email' field with the placeholder 'CORREO COMISARIO DE CARRERO RACING TO THE CLOUD', and a 'Create Subscription' button. A note states: 'Email notifications use the sender "noreply" at a region-specific notif. Example sender: noreply@notification.us-ashburn-1.oci.oraclecloud. Creating a subscription for Email.'

El comisario de carrera deberá **ACEPTAR** la suscripción

Create Subscription		
Subscription OCID	State	Protocol
<a href="#">ocid.....cbn2jtbq</a>	● Active	Email

# BASE DE DATOS NOSQL

Crear una tabla **NOSQL** llamada **Employees**



Con los campos **ID**, **NAME**, **LAST\_NAME**

Table creation mode

Simple input  
Form-based table schema entry

Advanced DDL input  
Supply table schema as a DDL statement

Reserved capacity

Capacity mode ⓘ

☒ Provisioned Capacity ☐ On Demand Capacity

Read capacity (ReadUnits)  Range: 1 to 40,000

Write capacity (WriteUnits)  Range: 1 to 20,000

Disk storage (GB)  Range: 1 to 5,000

Name

Primary key columns ⓘ

Column name  Type  ☐ Set as shard key

+ Another primary key column

Columns ⓘ

Column name	Type	Default value	Optional
<input type="text" value="NAME"/>	<input type="text" value="String"/>	<input type="text"/>	<input type="checkbox"/> Value is not null
<input type="text" value="LAST_NAME"/>	<input type="text" value="String"/>	<input type="text"/>	<input type="checkbox"/> Value is not null

+ Another column

Table time to live (days) Optional

The default TTL for rows in this table

[Show advanced options](#)

Create table Save as stack Cancel

## GENERAR AUTH TOKEN

En la esquina superior derecha del portal encontremos el **profile** del usuario donde podremos generar el token de autenticación:

US East (Ashburn)

### Profile

oracleidentitycloudservice/oscar.bernal@oracle.com

Tenancy: juanitacaballero22

[User settings](#)

[Console settings](#)

[Sign out](#)

### Resources

- [My groups](#)
- [My applications](#)
- [API keys](#)
- [Auth tokens](#)**
- [Customer secret keys](#)
- [My access tokens](#)

### Auth tokens

[Generate token](#) [Delete](#)

<input type="checkbox"/>	Description
<input type="checkbox"/>	bucket
<input type="checkbox"/>	registry

0 Selected

## Generate token

Description

function

**Guardar el valor generado por la consola el cual debemos usar en los pasos posteriores**

Ejemplo: v#1iD<8Ycx+)Z+XUR5av

## Generate token

**Generated token**

Copy this token for your records. It will not be shown again.

\*\*\*\*\* [Show](#) [Copy](#)

# CREACION APLICACIÓN SERVERLESS & SETUP CLOUD SHELL ENVIROMENT

1. Creación o validación de existencias de la capa de red, debe existir una VCN y una subred, si ya tienes creada una VCN puedes usar la existente.

## Start VCN Wizard

[Help](#)

☒ Create VCN with Internet Connectivity

☐ Add Internet Connectivity and Site-to-Site VPN to a VCN

Creates a VCN with a public subnet that can be reached from the internet. Also creates a private subnet that can connect to the internet through a NAT gateway, and also privately connect to the Oracle Services Network.

**Includes:** VCN, public subnet, private subnet, internet gateway (IG), NAT gateway (NAT), service gateway (SG).

Start VCN Wizard

[Cancel](#)

## Configuration



Resource availability checked successfully.

### Basic Information

VCN Name ⓘ

RedBullVCN

Compartment ⓘ

redbullhol

comteamcom (root)/TestEnvironment/redbullhol

### Configure VCN and Subnets

VCN CIDR Block ⓘ

10.0.0.0/16

If you plan to peer this VCN with another VCN, the VCNs must not have overlapping CIDRs. [Learn more.](#)

Public Subnet CIDR Block ⓘ

10.0.0.0/24

The subnet CIDR blocks must not overlap.

Private Subnet CIDR Block ⓘ

10.0.1.0/24

The subnet CIDR blocks must not overlap.

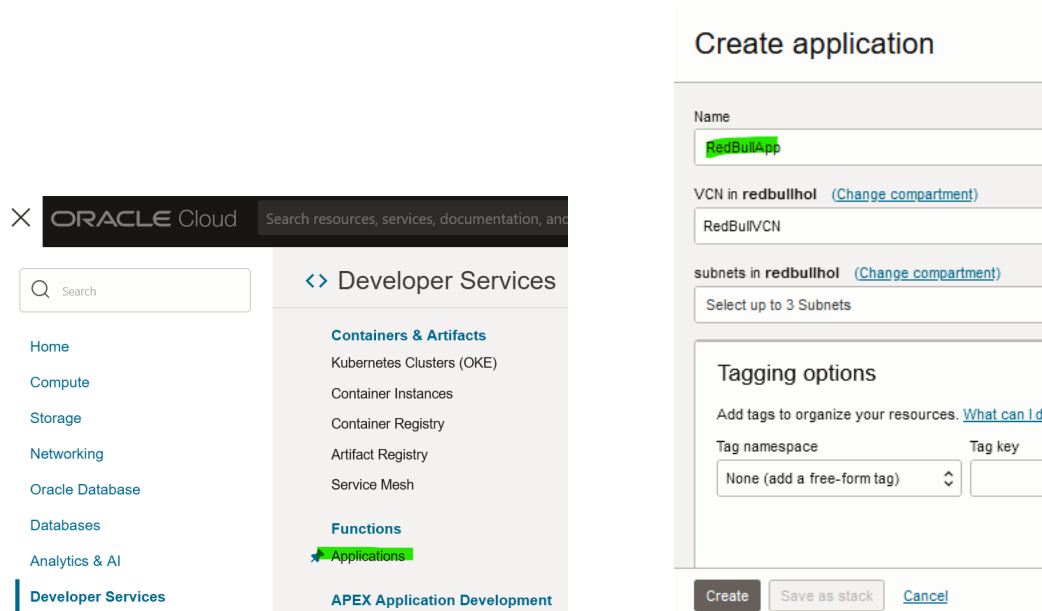
DNS Resolution

☒ Use DNS hostnames in this VCN

Required for Instance hostname assignment if you plan to use VCN DNS or a third-party DNS. This choice cannot be changed after the VCN is created. [Learn more.](#)

[Show Tagging Options](#)

2. Crear de Aplicación Serverless con el nombre **RedBullAppNoSQL**



ORACLE Cloud Search resources, services, documentation, and more

Search

Developer Services

- Containers & Artifacts
  - Kubernetes Clusters (OKE)
  - Container Instances
  - Container Registry
  - Artifact Registry
  - Service Mesh
- Functions
  - Applications
- APEX Application Development

Create application

Name: RedBullApp

VCN in redbullhol (Change compartment): RedBullVCN

subnets in redbullhol (Change compartment): Select up to 3 Subnets

Tagging options

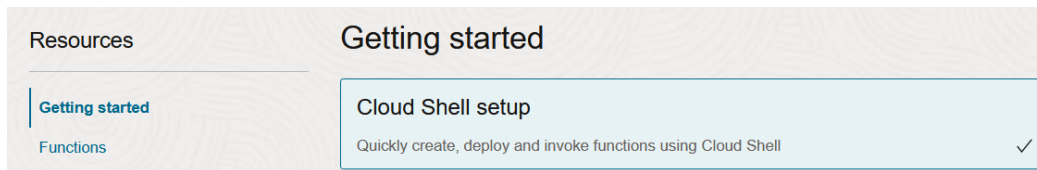
Add tags to organize your resources. [What can I do](#)

Tag namespace: None (add a free-form tag)

Tag key:

Create Save as stack Cancel

3. Setup del ambiente Cloud Shell para esto se deben seguir las instrucciones dadas en la consola en el siguiente apartado:

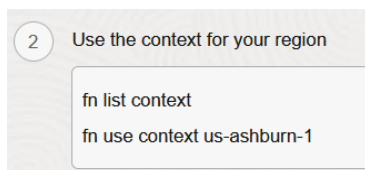


Aquí un ejemplo de los comandos y el resultado de cada una de las ejecuciones en Cloud Shell:

Dara clic en el botón “Launch Cloud Shell”



Listar los diferentes contextos serverless correspondientes a cada una de las regiones que se estén usando, ***copiar y ejecutar los comandos que aparecen en la consola:***





```

Your Cloud Shell machine comes with 5GB of storage for your home directory. Your Cloud Shell (machine and home directory) are located in: US East (Ashburn).
You are using Cloud Shell in tenancy cxteammicrn as an OCI Local user oscar.bernal@oracle.com

Type 'help' for more info.
oscar_bern@cloudshell:~ (us-ashburn-1)$ fn list context
CURRENT NAME      PROVIDER      API URL
default           oracle-cs
* us-ashburn-1    oracle-cs     https://functions.us-ashburn-1.oci.oraclecloud.com
us-phoenix-1      oracle-cs     https://functions.us-phoenix-1.oci.oraclecloud.com
oscar_bern@cloudshell:~ (us-ashburn-1)$ fn use context us-ashburn-1

Fn: Context us-ashburn-1 currently in use

```

## Actualizar el contexto para ser usado

3 Update the context with the function's compartment ID

```
fn update context oracle.compartment-id ocid1.compartment.oc1..aaaaa
```

```

oscar_bern@cloudshell:~ (us-ashburn-1)$ fn update context oracle.compartment-id ocid1.compart
Current context updated oracle.compartment-id with ocid1.compartment.oc1..aaaaaaa576tr4yeb6

```

## Establecer un **pre-fijo** para el contexto del repositorio de imágenes Docker

4 Provide a unique repository name prefix to distinguish your function images from other people's. For example, with 'jdoe' as the prefix, the image path for a 'hello' function image is '<region-key>.ocir.io/<tenancy-namespace>/jdoe/hello:0.0.1'

```
fn update context registry iad.ocir.io/idikzonisftg/[repo-name-prefix]
```

[Copy](#)

Establecer el repositorio de imágenes, para este caso **debes reemplazar [repo-name-prefix] por redbull**, en el comando dado por la consola quedado así:

```

oscar_bern@cloudshell:~ (us-ashburn-1)$ fn update context registry iad.ocir.io/idikzonisftg/redbull
Current context updated registry with iad.ocir.io/idikzonisftg/redbull
oscar_bern@cloudshell:~ (us-ashburn-1)$

```

Como en los pre-requisitos ya tenemos generado el token de autorización el paso 5 lo **omitiremos**

5 [Generate an Auth Token](#)

## Establecer conexión al repositorio de imágenes

6 Log into the Registry using the Auth Token as your password

```
docker login -u 'idikzonisftg/oscar.bernal@oracle.com' iad.ocir.io
```

Debemos ingresar el **token** creado en pasos previos cuando el sistema nos solicite el **password**

```
oscar_bern@cloudshell:~ (us-ashburn-1)$ docker login -u 'idikzonisftg/oscar.bernal@oracle.com' iad.ocir.io
Password:
WARNING! Your password will be stored unencrypted in /home/oscar_bern/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
```

Para este ejercicio **omitiremos los pasos posteriores** ya que el código de la aplicación ya fue construido y se encuentra en un repositorio de Gitlab, solamente tendremos que importarlo y compilarlo en nuestro ambiente.

## CONSTRUCCIÓN DE APLICACIÓN

1. En la sesión de Cloud Shell vamos importar el código de la aplicación, con el siguiente comando:

`git clone https://gitlab.com/oscarbm7/oci-serverless-python-nosql.git`

```
Cloud Shell
oscar_bern@cloudshell:~ (us-ashburn-1)$ git clone https://gitlab.com/oscarbm7/oci-serverless-python-nosql.git
Cloning into 'oci-serverless-python-nosql'...
remote: Enumerating objects: 11, done.
remote: Counting objects: 100% (11/11), done.
remote: Compressing objects: 100% (9/9), done.
remote: Total 11 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (11/11), 5.57 KiB | 1.11 MiB/s, done.
```

2. Descargar el archivo **func.yaml** el cual tiene los parámetros de configuración de la aplicación, ingresando al siguiente link:

<https://gitlab.com/oscarbm7/oci-serverless-python-nosql/-/raw/main/func.yaml?inline=false>

3. Editar el archivo (En Notepad) remplazando los valores resaltados por datos recopilados previamente:

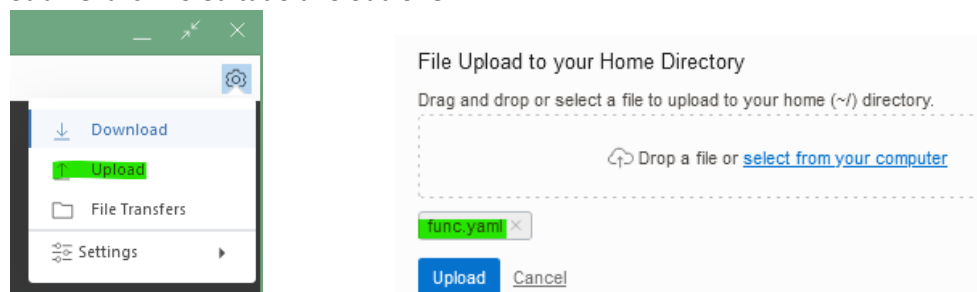
```
func
File Edit View

schema_version: 20180708
name: nosql-load-file
version: 0.0.16
runtime: python
build_image: fnproject/python:3.9-dev
run_image: fnproject/python:3.9
entrypoint: /python/bin/fdk /function/func.py handler
memory: 2048
timeout: 300
config:
  COMPANY: ACME CORP XXXXXX
  TABLE_NAME: Employees
  TOPIC_OCID: ocid1.onstopic.oc1.iad.aaaaaaaagruoyyzvxufczc5o42
```

COMPANY\_NAME: **NOMBRE DE LA EMPRESA**

TOPIC\_OCID: Es el **tema de notificaciones** que creamos en pasos anteriores, al igual que

4. Subir el archivo editado al Cloud Shell:



Este subirá al directorio home del Cloud Shell

5. Remplazar el archivo de la función con el subido previamente:

***mv func.yaml oci-serverless-python-nosql***

6. Ingresamos a la carpeta importada, con el siguiente comando: ***cd oci-serverless-python-nosql***

Cloud Shell

```
oscar_bern@cloudshell:~ (us-ashburn-1)$ cd oci-serverless-python-nosql/
```

7. Compilar la aplicación serverless, con el comando: ***fn -v deploy --app RedBullAppNoSQL***

```
oscar_bern@cloudshell:oci-serverless-python-nosql (us-ashburn-1)$ fn -v deploy --app RedBullAppNoSQL
Deploying nosql-load-file to app: RedBullAppNoSQL
Bumped to version 0.0.17
Using Container engine docker
Building image iad.ocir.io/idikzonisftg/redbull/nosql-load-file:0.0.17
Dockerfile content
```

Si todo esta correcto el resultado en el CloudShell debe ser:

***Successfully created function***

***Updating function nosql-load-file using image.....***

8. Habilitar LOGS para rastreo de errores e información relevante

### Create application

Name

RedBullAppNoSQL

VCN in **redbullhol** [\(Change compartment\)](#)

No data available

subnets in **redbullhol** [\(Change compartment\)](#)

Loading...

#### Tagging options

Add tags to organize your resources. [What can I do with tagging?](#)

Tag namespace Tag key

None (add a free-form tag)

Create Save as stack Cancel

### Enable Log

For more information about service logs, see [documentation](#).

Compartment

redbullhol

cxmteammcn (root)/TestEnvironment/redbullhol

Log Group ⓘ

Select a log group

☒ Auto-create a default Log Group

☐ Create a new Log Group

Log Name

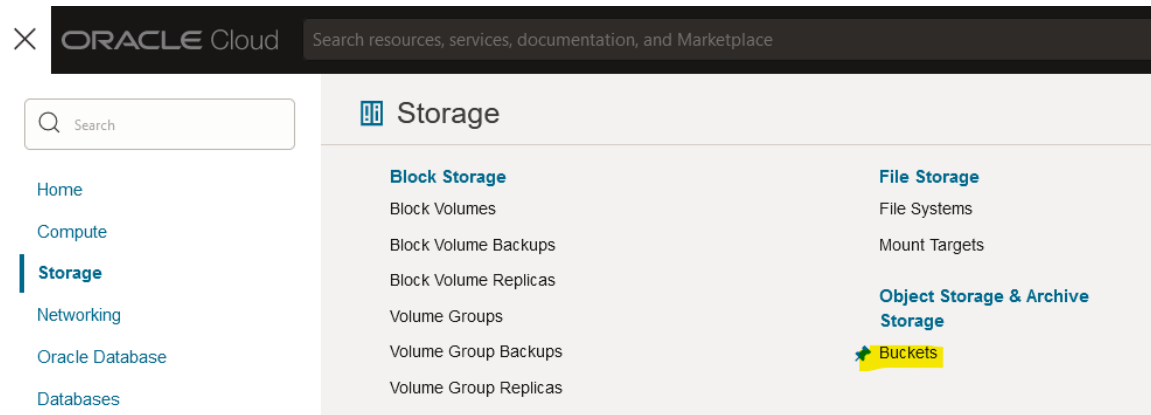
RedBullAppNoSQL\_invoke

Log Retention

Enable Log Cancel

## CREACION BUCKET PARA ARCHIVOS

Crear Bucket en el servicio de Object storage



**IMPORTANTE:** habilitar la opción de emisión de eventos, ya que esto es lo que ejecutara la función serverless para la carga del respectivo archivo.

### Create Bucket

Bucket Name

Files

Default Storage Tier

☒ Standard

☐ Archive

The default storage tier for a bucket can only be specified during creation. Once set, you cannot change it.

☐ Enable Auto-Tiering

Automatically move infrequently accessed objects from the Standard tier to less expensive storage tiers.

☐ Enable Object Versioning

Create an object version when a new object is uploaded, an existing object is overwritten, or an object is deleted.

☒ Emit Object Events

Create automation based on object state changes using the [Events Service](#).

☐ Uncommitted Multipart Uploads Cleanup

Create a lifecycle rule to automatically delete uncommitted multipart uploads older than 7 days.

Encryption

☒ Encrypt using Oracle managed keys

Leaves all encryption-related matters to Oracle.

☐ Encrypt using customer-managed keys

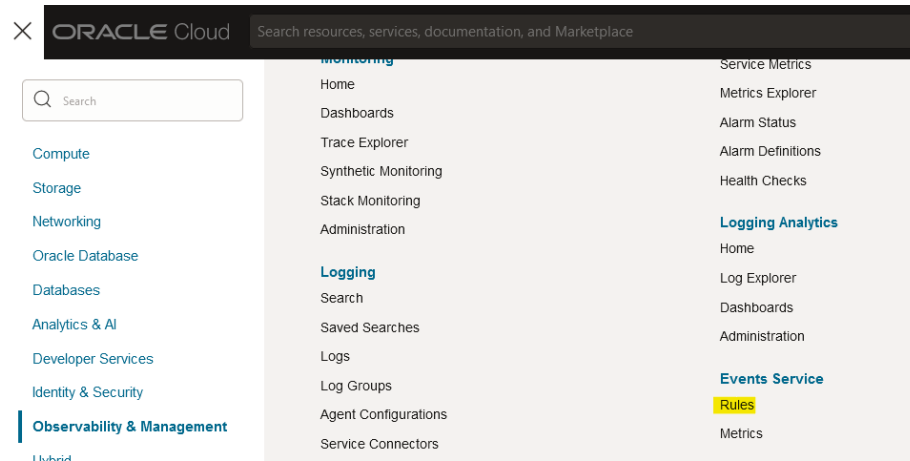
Requires a valid key from a vault that you have access to. [Learn more](#)

Tags

Create Cancel

# CONFIGURACION SERVICE CONNECTOR HUB

En el módulo de Observability



Crear la regla que ejecutara la función cada vez que se cargue el archivo

## Create Rule

[Help](#)

Display Name  
load\_files

Description  
Describe what the rule does. Example: Sends a notification when backups complete.

**Rule Conditions**  
Limit the events that trigger actions by defining conditions based on event types, attributes, and filter tags. [Learn more](#)

Condition : Service Name : Event Type  
Event Type : Object Storage : Object - Create

To emit events for object state changes, enable Emit Object Events on the bucket details page. [Learn more](#).

Condition : Attribute Name : Attribute Values  
Attribute : bucketName : Files

[+ Another Condition](#)

**Rule Logic**  
MATCH event WHERE (  
 eventType EQUALS ANY OF (  
 com.oraclecloud.objectstorage.createobject  
 )  
 AND (  
 bucketName MATCHES ANY OF (  
 Files  
 )  
 )  
)  
[View example events \(JSON\)](#)  
[Validate Rule](#)

**Actions**  
Actions trigger for the specified event conditions. [Learn more](#).

Action Type : Function Compartment : Function Application : Function  
Functions : redbullhol : RedBullApp : load-file

[Create Rule](#) [Save as stack](#) [Cancel](#)

## PROBAR APLICACIÓN

Finalmente puedes probar tu aplicación únicamente cargando el archivo en el bucket creado previamente del servicio object storage.

1. Descargar archivo de prueba CSV:

<https://objectstorage.us-ashburn-1.oraclecloud.com/p/MSmBkReA-TET1pfUpsvX5ZsC6uTFZpU140p7t7uitURUJ9hwOKOI0z0O5mn7stXJ/n/idikzonisftg/b/DataFile/o/Employees.csv>

2. Cargar el archivo en el bucket creado

The screenshot displays the Oracle Cloud Object Storage console. On the left, the 'Bucket Details' page for a bucket named 'B' is visible, showing general information like namespace, compartment, and creation date. The 'Files' tab is active, showing a list of objects. On the right, the 'Upload Objects' modal is open, showing the 'Object Name Prefix' field, 'Storage Tier' set to 'Standard', and a list of files to be uploaded. The file 'Employees.csv' is highlighted, showing its size as 128 bytes. The 'Upload' button is visible at the bottom of the modal.

Después de cargado el archivo será procesado por la función Serverless y cargado en la base de datos en la tabla creada previamente **Employess**

The screenshot shows the Oracle Cloud console interface. The 'Databases' section is highlighted in the left sidebar. The main content area displays a list of database services, including 'MySQL', 'MySQL HeatWave on AWS', and 'Oracle NoSQL Database'. The 'Oracle NoSQL Database' service is highlighted, showing its 'Tables' tab.

Oracle NoSQL Database

Tables

Tables

List scope

Compartment

redbullhol

cxmteamocrn (root)/TestEnvironment/redbullhol

Create table

Name	Status
<a href="#">Employees</a>	● Active

Ingresamos a la tabla y en la opción **Explore Data** y dar click en el botón **Execute**

NoSQL > Tables > Table details

T

ACTIVE

Insert row

Upload data

View table DDL

Move table

More actions

Table information

Tags

General information

OCID: [\\_ozdlrdb3jg](#)

Compartment: redbullhol

Compartment OCID: [\\_qjlfsv9na](#)

Created: 2023-04-20T04:48:09.888Z

Resources

Columns

Indexes

Explore data

Child tables

Metrics

Columns

Add columns

Primary key	Column name
Yes	ID
No	NAME
No	LAST_NAME

Explore data

SQL statement

SELECT \* FROM Employees

Execute

Show query execution plan

Query results do not automatically update when the table is modified.



**Bibliografía:**

<https://docs.oracle.com/en-us/iaas/Content/Functions/Concepts/functionsoverview.htm>

<https://oracle.github.io/python-oracledb/>

<https://fnproject.io/>

<https://medium.com/oracledevs/an-exploration-using-oci-functions-4c5d4e70d00c>

<https://oracle-cloud-infrastructure-python-sdk.readthedocs.io/en/latest/api>