

# ¿Cómo hacer las asignaciones en Workforce mediante Python?

José Córdova-Fredes

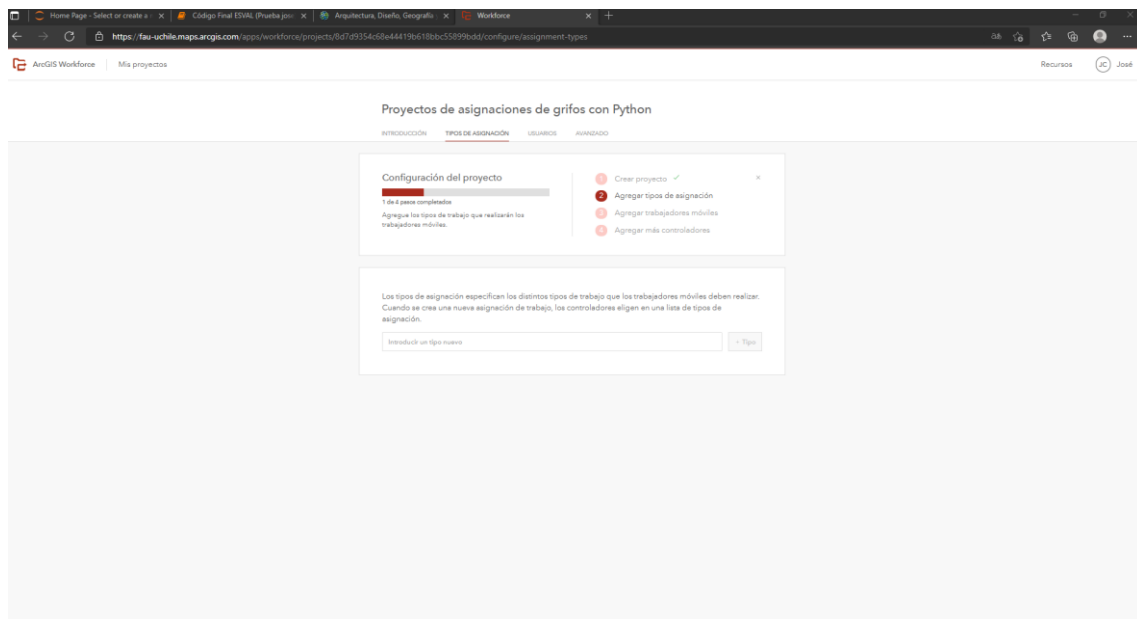
jicordova@uc.cl

El notebook del proyecto se encuentra en el siguiente repositorio:

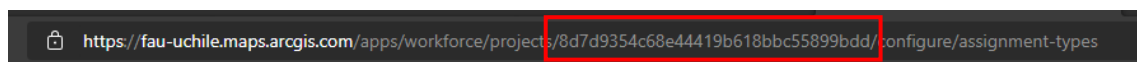
<https://github.com/jicordovafredes/SIG>

Lo importante es empezar con el código y el Workforce al mismo tiempo.

Lo siguiente es crear un Proyecto de Workforce desde 0, este paso es relevante dado que no se puede sobrescribir o actualizar la información ya definida porque arroja un error en la consola Python.



Lo importante del Workforce radica en la url, dado que en este caso tendríamos lo siguiente:



Es importante encontrar el ID de este Workforce, que sería:

8d7d9354c68e44419b618bbc55899bdd

## Código en Python

```
In [1]: import datetime
import json
import arcgis
from arcgis.gis import GIS
from arcgis.geometry import Geometry
from arcgis.apps import workforce
```

1

### Conectando organización

2

```
In [2]: gis = GIS("https://fau-uchile.maps.arcgis.com/", "jicoardova_fau")
Enter password: .....
```

### Workforce vacio

3

```
In [3]: ##Se trae un proyecto vacío, esto es importante, o sino es posible que no funcione
item = gis.content.get("8d7d9354c68e44419b618bbc55899bdd")
project = workforce.Project(item)
project
```

Out[3]: <Project Asignaciones>

1. Llamar todas las librerías necesarias para hacer funcionar el script, si bien se sugiere que todas las librerías vayan en la primera fila por un tema de orden, no hay inconveniente alguno en llamar las librerías en cualquier lugar del script.
2. Se conecta la organización donde la primera parte corresponde al portal en forma de url (paso importante), y la segunda parte corresponde al nombre del usuario; al correr el código pedirá la contraseña para poder conectarse al portal
3. Acá es donde se ingresa el ID del Workforce vacío, para realizar la conexión y para que empiece hablar entre el Workforce y Python mediante el código, para confirmar la “vinculación” de manera correcta debe salir el nombre del proyecto de Workforce de esta forma <Nombre del Proyecto>

Las asignaciones en el Workforce se encuentran vacío, por lo que mediante el siguiente código se ingresa las asignaciones que se creen necesarias, para este caso tenemos tres.

### Agregar tipo de asignaciones

Agregar asignaciones, no las borra, ni las reemplaza Es importante indicar que si arroja error de asignacion, se debe crear un workforce vacio nuevamente

```
In [4]: grifos = project.assignment_types.add(name="Inspección de grifos")
camaras = project.assignment_types.add(name="Inspección de camaras")
riles = project.assignment_types.add(name="Inspección de Riles")
```

Es relevante e importante indicar que una vez que se agregan las asignaciones no es posible volver a modificar o actualizar estas asignaciones, pues arrojarían error en el código, en caso de querer realizar este procedimiento se indica comenzar con un nuevo Workforce desde 0.

Para ver si en el Workforce están las asignaciones se actualiza o derechamente se aprieta f5.

## Proyectos de asignaciones de grifos con Python

INTRODUCCIÓN

TIPOS DE ASIGNACIÓN

USUARIOS

AVANZADO

### Configuración del proyecto

2 de 4 pasos completados

Agregue más tipos ahora o seleccione la pestaña Usuarios para seguir configurando el proyecto.

- 1 Crear proyecto ✓
- 2 Agregar tipos de asignación ✓
- 3 Agregar trabajadores móviles
- 4 Agregar más controladores

Los tipos de asignación especifican los distintos tipos de trabajo que los trabajadores móviles deben realizar. Cuando se crea una nueva asignación de trabajo, los controladores eligen en una lista de tipos de asignación.

+ Tipo

#### Tipo de asignación

Inspección de camaras

✕

Inspección de grifos

✕

Inspección de Riles

✕

Pues bien, las asignaciones se han hecho de forma correcta.

El siguiente paso es agregar los trabajadores y los supervisores ¿Cómo? Pues fácil, tienes que ir al portal, y seleccionar la opción “Grupos”, en la parte izquierda se encontrara un banner que dice “Crear Grupo”, se crea el grupo con las especificaciones de Trabajadores y Supervisores; y se debe agregar los usuarios de arcgis online para cada grupo cuando corresponda.

### Se Agregan Trabajadores

```
In [18]: trabajadores = gis.groups.get("5573ec020617466288bd149990d1ace0")
trabajadores
```

Out[18]:



#### Trabajadores

Summary: Grupo de Trabajadores  
Description: Control de prueba para controlar y organizar las cuadrillas de revisar para así optimizar el flujo de trabajo  
Owner: jicoardova\_fau  
Created: August 03, 2021

```
In [*]: for username in trabajadores.get_members()["users"]:
        user = gis.users.get(username)
        project.workers.add(
            user_id=username,
            name=user.fullName,
            status="not_working",
            title="Trabajador de campo"
        )
```

Para confirmar vamos al workforce y nos vamos a la pestaña “usuarios”

1 - 5 de 5

Usuario	Buscar usuarios	Rol	
AS	Andrés San Martín (andraes.smgo_fau) Trabajador de campo	Trabajador móvil	⊗
DD	Daniela Duque Ramos (INTERGEO_04) Trabajador de campo	Trabajador móvil	⊗
FM	Felipe Mancilla Cofré (INTERGEO_06) Trabajador de campo	Trabajador móvil	⊗
IH	Ihanelly Hernández (ihanaellyhdzv_fau) Trabajador de campo	Trabajador móvil	⊗
JC	José Cordova-Fredes (jicoardova_fau)	Controlador (propietario)	Activa

Y tenemos que todos los trabajadores, se le ha asignado una etiqueta llamada “Trabajador de campo”, que está definida en el código y su status es “not\_working”

El paso anterior se repite con los supervisores

### Se agregan los supervisores

```
In [20]: supervisores_grifos = gis.groups.get("be5f84c01c2d404ca78ec97ad5c17ee0")
supervisores_grifos
```

```
Out[20]:
```



[Supervisores](#)

**Summary:** Grupo de Supervisores

**Description:** La función de los supervisores es poder controlar la acción de los trabajadores con la finalidad de que hagan bien su trabajo

**Owner:** jicoardova\_fau

**Created:** August 03, 2021

```
In [*]: for username in supervisores_grifos.get_members()["users"]:
    if username != "jicoardova_fau":
        user = gis.users.get(username)
        project.dispatchers.add(
            user_id=username,
            name=user.fullName
        )
```

Se actualiza y se confirma en el Workforce.

1 - 7 de 7

Usuario	Buscar usuarios	Rol	
AS	Andrés San Martín (andraes.smgo_fau) Trabajador de campo	Trabajador móvil	⊗
DD	Daniela Duque Ramos (INTERGEO_04) Trabajador de campo	Trabajador móvil	⊗
FM	Felipe Mancilla Cofré (INTERGEO_06) Trabajador de campo	Trabajador móvil	⊗
IH	Ihanelly Hernández (ihanaellyhdzv_fau) Trabajador de campo	Trabajador móvil	⊗
JC	José Cordova-Fredes (jicoardova_fau)	Controlador (propietario)	Activa
MH	Michel Harold Arredondo (michael_a_g_fau)	Controlador	⊗
RA	Rodrigo Andrés Pineda (rodraigopineda1...)	Controlador	⊗

Entonces ahora tenemos a los Trabajadores (Trabajador móvil) y a los Supervisores (Controlador).

Para ver el mapa donde se quiere trabajar, es necesario vincular e integrarlo, quedando de la siguiente manera.

## Agregar integracion de explorer

Basicamente se agrega un mapa con la informacion necesaria

```
In [22]: from arcgis.apps import build_explorer_url
from arcgis.mapping import WebMap

webmap = WebMap(gis.content.get("cbcbc17fdbff4e3697300d585366b817"))
webmap.item.share(groups=[project.group])
webmap
```



```
In [23]: url = build_explorer_url(
webmap=webmap,
center="{assignment.latitude},{assignment.longitude}",
url_type="App",
scale=9000
)
```

```
In [24]: project.integrations.add("default-explorer", "Explorer At Assignment", url)
```

```
Out[24]: <Integration default-explorer>
```

La segunda parte es realizar las asignaciones mediante un archivo csv

Para leer el archivo es necesario contar con la librería Pandas.

## Conectar y leer csv

```
In [25]: import pandas as pd
```

```
In [26]: assignments = f'c:/Users/jcord/OneDrive/Escritorio/XYGO/Scripts/asignaciones.csv'
df = pd.read_csv(assignments, sep=";")
df
```

```
Out[26]:
```

		Location	Description	Assigned	Priority	Work Order Id	Due Date
0	2	Papudo 815, La Ligua	Zona 1: Colo Colo	andreaes.smgo_fau	high	1	30/8/2021 22:43
1	3	Avenida Holanda 34, La Ligua	Zona 2: San Luis	INTERGEO_04	medium	2	31/8/2021 21:43
2	4	Laulie 098, La Ligua	Zona 3: Deportes PM	INTERGEO_04	low	3	1/9/2021 19:43
3	5	Diego Portales 699, La Ligua	Zona 4: Naval	INTERGEO_06	critical	4	2/9/2021 10:43
4	6	Luis Cruz Martinez 690, La Ligua	Zona 5: Tricolor	INTERGEO_06	high	5	3/9/2021 18:43
5	7	Ignacio Carrera Pinto 830, La Ligua	Zona 6: Lozapenco FC	andreaes.smgo_fau	low	6	4/9/2021 14:00

Si Bien ya tenemos el grupo de usuarios creados, es relevante e importante que le asignemos a una variable de la siguiente forma:

## Buscar y asignar a los usuarios

Borrar si no sirve

```
: Daniela = project.workers.get(user_id="INTERGEO_04")
Felipe = project.workers.get(user_id="INTERGEO_06")
Andres = project.workers.get(user_id="andreaes.smgo_fau")
```

Una vez que asignados las variables usuarias, el paso siguiente es crear asignaciones para cada fila

### Crear una asignación para cada fila

Para cada asignación, primero codifique geográficamente la dirección para obtener la ubicación x, y en (WGS84 Web Mercator) de la asignación. Luego proporcione atributos adicionales

Finalmente, use el método batch\_add para agregar múltiples asignaciones a la vez (esto es más rápido que usar el método add ya que la validación se realiza una vez para todas las asignaciones)

```
In [29]: assignment_types = project.assignment_types.add(name="Inspección de Grifo")
```

```
In [30]: from arcgis.geocoding import geocode
         from datetime import datetime
```

Es importante que tipo de asignaciones quede de la siguiente forma:

Assignment\_types = Project.assignment\_types.add (nombre de variable)

Se importa la librería geocode y datetime

Para hacer la asignación para cada Trabajador es relevante lo siguiente:

#### Asignación para el Trabajador "Daniela"

```
In [58]: assignments = []
         for index, row in df.iterrows():
             geometry = geocode(f"{row['Location']}", out_sr=3857)[0]["location"]
             assignments.append(
                 workforce.Assignment(
                     project,
                     geometry=geometry,
                     location=row["Location"],
                     description=row["Description"],
                     priority=row["Priority"],
                     work_order_id=row["Work Order Id"],
                     assignment_type="Inspección de Grifo",
                     status="assigned",
                     worker="Daniela",
                     due_date=pd.to_datetime(row["Due Date"])
                 )
             )
```

```
In [59]: #Agregar asignaciones con el metodo batch_add
         project.assignments.batch_add(assignments)
```

```
Out[59]: [<Assignment 31>,
         <Assignment 32>,
         <Assignment 33>,
         <Assignment 34>,
         <Assignment 35>,
         <Assignment 36>]
```

Cambiar los siguientes campos:

description = Zona 1 (en este caso le daremos una zona un usuario, en el script anterior se le asigno toda la tarea al trabajador "Daniela").

status: "assigned" para asignado) y "unassigned" para no asignado

worker=(nombre del trabajador,en este caso Daniela).

Este paso se repite n veces, donde n es la cantidad de trabajadores que haya que asignar.

Solo para tener un control de la cantidad total de asignaciones realizadas se tiene el siguiente código:

```
assignments = project.assignments.search()
```

```
assignments
```

Luego para ver un resumen de la descripción de las asignaciones, se tiene lo siguiente:

### Descripción de las asignaciones

```
In [51]: assignment = assignments[0]
print(f'Status: {assignment.status}')
print(f'Description: {assignment.description}')
print(f'Priority: {assignment.priority}')
print(f'Assigned To: {assignment.worker}')
print(f'Type: {assignment.assignment_type}')
print(f'Due Date: {assignment.due_date}')
```

Status: assigned  
Description: Zona 1: Colo Colo  
Priority: high  
Assigned To: Daniela Duque Ramos (INTERGEO\_04)  
Type: Inspección de Grifo  
Due Date: 2021-08-30 22:43:00+00:00

El ultimo paso es verificar las asignaciones en el mapa

### Verificar asignaciones en el mapa

```
In [57]: webmap = gis.map("La Ligua", zoomlevel=12)
webmap.add_layer(project.assignments_layer)
webmap
```



Finalmente es posible apreciar en el Workforce lo siguiente:

## Proyectos de asignaciones de grifos con Python

INTRODUCCIÓN TIPOS DE ASIGNACIÓN USUARIOS AVANZADO



### Acerca de este proyecto

Proyecto de pruebas para XYGO a cargo de José Córdova-Fredes

[Abrir proyecto](#)

## Usuarios

7

4 trabajadores móviles,  
3 controladores

## Asignaciones

36

0 completadas, 24 asignadas,  
12 sin asignar

## Mapas

 Mapa del controlador

 Mapa del trabajador

Crear área de mapa sin conexión