

ESCALA PARA EL PROYECTO PYTHON - 1 - MATRIZ (/PROJECTS/PYTHON-1-ARRAY)

Debes evaluar a 1 estudiante en este equipo.



Repositorio Git

git@vogsphere.42urduliz.com:vogsphere/intra-uuid-6b45a4ae-81c1-4fc4-

Introducción

- Mantén una actitud educada, cortés, respetuosa y constructiva durante todo el proceso de evaluación. El bienestar de la comunidad depende de ello.

- Identifica con la persona (o el grupo) evaluado las eventuales disfunciones del trabajo. Tómate el tiempo para discutir y debatir los problemas que hayas identificado.

- Debes considerar que puede haber alguna diferencia en cómo tus pares pudieron haber entendido las instrucciones del proyecto y el alcance de sus funcionalidades. Mantén siempre una mente abierta y califícalo de la manera más honesta posible. La pedagogía es válida sólo y únicamente si la evaluación entre pares se realiza con seriedad.

Pautas

- Califica únicamente el trabajo que se encuentre en el repositorio GiT del alumno o grupo.

- Comprueba que el repositorio GiT pertenece al alumno o grupo. Asegúrate de que el trabajo sea para el proyecto correspondiente y comprueba también que se utilice "git clone" en una carpeta vacía.

- Comprueba con atención que no se hayan utilizado alias maliciosos para engañarte y hacerte evaluar algo distinto al contenido del repositorio oficial.

- Para evitar sorpresas, comprueba con atención que tanto el alumno evaluado como el alumno evaluado hayan revisado los posibles scripts utilizados para facilitar la calificación.

- Si el alumno evaluador aún no ha completado ese proyecto en particular, es obligatorio que este alumno lea todo el tema antes de comenzar la defensa.

- Utiliza las banderas disponibles en esta escala para señalar un repositorio vacío, un programa que no funciona, trampas, etc. En estos casos, la calificación finaliza y la calificación final es 0, o -42 en caso de trampas. Sin embargo, salvo la excepción de hacer trampa, se le anima a seguir discutiendo su trabajo incluso si este último está en curso para identificar cualquier problema que pueda haber causado el fracaso del proyecto y evitar repetir el mismo error en el futuro.

- Recuerde que durante la duración de la defensa, no se permite ninguna otra terminación inesperada, prematura o descontrolada del programa, de lo contrario la calificación final es 0 para el ejercicio, y continúe con la evaluación.

- Nunca debe tener que editar ningún archivo excepto el archivo de configuración si este último existe. Si desea editar un archivo, tómese el tiempo para explicar por qué con el estudiante evaluado y asegúrese de que ambos estén de acuerdo en esto.

- Las importaciones de lib deben ser explícitas, por ejemplo, no se permite importar "from pandas import *" , debe poner 0 al ejercicio y continuar con la evaluación.

- Sus ejercicios van a ser evaluados por otros estudiantes, asegúrese de que los nombres de sus variables y nombres de funciones sean apropiados y civiles.

Archivos adjuntos

tema.pdf (https://cdn.intra.42.fr/pdf/pdf/129305/en.subject.pdf)

paisaje.jpg (https://cdn.intra.42.fr/document/document/25160/landscape.jpg)

animal.jpeg (https://cdn.intra.42.fr/document/document/25161/animal.jpeg)



Parte obligatoria

Gestión de errores

Realice AL MENOS las siguientes pruebas para intentar estresar la gestión de errores

- El repositorio no está vacío.
- Sin trampas.
- No hay ninguna función/biblioteca prohibida.
- No hay ninguna variable global.
- El ejecutable tiene el nombre esperado.
- Norminette no muestra errores. (pip install flake8, alias norminette=flake8, usa el indicador Norme)
- Las importaciones de bibliotecas deben ser explícitas, por ejemplo, debe "importar numpy como np". (No se permite importar "from pandas import *" y obtendrá 0 en el ejercicio).
- Si un ejercicio está mal, pasa al siguiente.

✓ Si

✗ No

ex00 Dame mi IMC

El programa debe calcular el IMC de cada persona y devolver Verdadero si el IMC excede el límite dado en int.

Su tester.py:

```
from give_bmi import give_bmi, apply_limit

height = [1.71, 1.65, 1.73, 1.95, 1.63]
weight = [65.3, 58.4, 63.4, 94.5, 72.9]

bmi = give_bmi(height, weight)
print(bmi, type(bmi))
print(apply_limit(bmi, 26))
```

Resultado esperado:

```
$> python tester.py
[22.33165760404911, 21.45087235996327, 21.183467539844298, 24.85207100591716,
27.43799164439761] <class 'list'>
[False, False, False, False, True]
$>
```

Puedes probar la función enviando listas que no sean del mismo tamaño, con cadenas en lugar de int, listas vacías, pero no seas demasiado cruel.

✓ Si

✗ No

matriz 2D ex01

El programa debe tomar como parámetros una matriz 2D que imprima su forma y devuelva una versión truncada con los argumentos inicial y final. Compruebe en el código que el estudiante ha utilizado el método de corte.

Su tester.py:

```
from array2D import slice_me

family = [[2.10, 78.45],
          [4.15, 6.70],
          [2.10, 98.5],
          [1.88, 75.2]]

print(slice_me(family, 0, 2))
print(slice_me(family, 1, -2))
```

Resultado esperado:

```
$> python test_array2D.py
My shape is : (4, 2)
My new shape is : (2, 2)
[[2.1, 78.45], [4.15, 6.7]]
My shape is : (4, 2)
My new shape is : (1, 2)
[[4.15, 6.7]]
```

✓ Si

✗ No

ex02 carga mi imagen

El programa debe tomar en parámetros una ruta de una imagen e imprimir su forma y devolver su array, puedes probar el programa con imágenes en jpeg y jpg, probar también un error de ruta e imagen corrupta.



CamScanner

Su tester.py:

```
from load_image import ft_load

print(ft_load("landscape.jpg"))
```

Resultado esperado:

```
\begin{42console}
$> python tester.py
The shape of image is: (257, 450, 3)
[[[19 42 83]
  [23 42 84]
  [28 43 84]
  ...
  [ 0  0  0]
  [ 1  1  1]
  [ 1  1  1]]]
$>\end{42console}
```

☒ Sí

☐ No

ex03 Hazme zoom

Observa en el sujeto el renderizado que debe tener la imagen luego del "zoom".
Si el "zoom" es ligeramente diferente al del sujeto, no es importante.

```
$> python zoom.py
The shape of image is: (768, 1024, 3)
[[[120 111 132]
  [139 130 151]
  [155 146 167]
  ...
  [120 156  94]
  [119 154  90]
  [118 153  89]]]
New shape after slicing: (400, 400, 1) or (400, 400)
[[[167]
  [180]
  [194]
  ...
  [102]
  [104]
  [103]]]
$>
```

☒ Sí

☐ No

ex04 girame

Observa en el sujeto el renderizado que debe tener la imagen luego de la "rotación".
Si la "rotación" es ligeramente diferente a la del sujeto, no es importante.
Pero en cualquier caso, se tuvo que hacer una rotación de 90 grados en sentido antihorario y un efecto espejo.
El estudiante debe realizar la transposición por sí mismo, está prohibido el uso de `numpy.transpose` y es 0 el ejercicio.

☒ Sí

☐ No

ex05 Engalana mi imagen

Observa en el tema cómo debe quedar la imagen después de los 5 tipos de transformación de imagen.
Observa en el código si se han respetado las restricciones del tema para cada función.
Darás 1 punto por cada buena visualización de la transformación de imagen.

Su tester.py:

```
from load_image import ft_load
from pimp_image import ft_invert, ft_red, ft_green, ft_blue, ft_grey

array = ft_load("landscape.jpg")

ft_invert(array)
ft_red(array)
ft_green(array)
ft_blue(array)
ft_grey(array)
```



Calificaciones

No olvides revisar la bandera correspondiente a la defensa.

✓ De acuerdo

★ Proyecto destacado

Trabajo vacío

🚩 Trabajo incompleto

📋 Norma

📄 Engañar

🚫 Chocar

⬆️ Situación preocupante

🚫 Función prohibida

Conclusión

Deja un comentario sobre esta evaluación

Finalizar evaluación

Declaración sobre el uso de cookies
(<https://profile.intra.42.fr/legal/terms/2>)

Política de privacidad
(<https://profile.intra.42.fr/legal/terms/5>)

Condiciones generales de uso del
sitio
(<https://profile.intra.42.fr/legal/terms/6>)

Reglamento de procedimiento
(<https://profile.intra.42.fr/legal/terms/4>)

Condiciones de uso de la
videovigilancia
(<https://profile.intra.42.fr/legal/terms/1>)



CamScanner