



Discovery Piscine

Módulo5 - Python

Resumen: En este Módulo5, vemos cómo usar arrays y sus funciones asociadas.

Versión: 2.00

Índice general

I.	Unas palabras sobre esta Discovery Piscine	2
II.	Introducción	3
III.	Instrucciones generales	4
IV.	Ejercicio 00: create_array	5
V.	Ejercicio 01: play_with_arrays	6
VI.	Ejercicio 02: play_with_arrays++	7
VII.	Ejercicio 03: play_with_arrays+=2	8
VIII.	Ejercicio 04: Parámetros	9
IX.	Entrega y Evaluación entre Pares	10

Capítulo I

Unas palabras sobre esta Discovery Piscine

¡Bienvenido!

Comenzarás un módulo de esta Discovery Piscine en programación informática. Nuestro objetivo es introducirte al código que se esconde detrás del software que usas a diario y sumergirte por completo en el aprendizaje entre pares, el modelo educativo de 42.

Programar trata sobre lógica, no sobre matemáticas. Te brinda bloques de construcción básicos que puedes combinar de innumerables formas. No existe una única solución “correcta” para un problema; tu solución será única, al igual que la de cada uno de tus compañeros.

Rápida o lenta, elegante o desordenada, mientras funcione, ¡eso es lo que importa! Estos bloques de construcción formarán una secuencia de instrucciones (para cálculos, visualizaciones, etc.) que el ordenador ejecutará en el orden que diseñes.

En lugar de ofrecerte un curso donde cada problema tiene una única solución, te colocamos en un entorno de aprendizaje entre pares. Buscarás elementos que te ayuden a afrontar el desafío, los refinarás mediante pruebas y experimentación, y finalmente crearás tu propio programa. Habla con otros, comparte tus perspectivas, genera nuevas ideas en conjunto y prueba todo por ti mismo para asegurarte de que funcione.

La evaluación entre pares es una gran oportunidad para descubrir enfoques alternativos y detectar posibles problemas en tu programa que podrías haber pasado por alto (piensa en lo frustrante que puede ser que un programa se bloquee). Cada revisor analizará tu trabajo de manera diferente, como clientes con expectativas variadas, brindándote nuevas perspectivas. Incluso podrías establecer conexiones para futuras colaboraciones.

Al final de esta Piscine, tu recorrido será único. Habrás enfrentado distintos desafíos, validado diferentes proyectos y elegido caminos distintos a los de los demás, ¡y eso está perfectamente bien! Esta es una experiencia tanto colectiva como individual, y todos sacarán algo valioso de ella.

Buena suerte a todos; esperamos que disfruten este viaje de descubrimiento.

Capítulo II

Introducción

Lo que aprenderás en este módulo:

- Aprenderás a usar arrays y sus funciones asociadas.

Capítulo III


Instrucciones generales

A menos que se indique lo contrario, las siguientes reglas se aplican todos los días de esta Piscine.

- Este documento es la única fuente confiable. No te fíes de rumores.
- Este documento puede actualizarse hasta una hora antes del plazo de entrega.
- Las tareas deben completarse en el orden especificado. No se evaluarán tareas posteriores si las anteriores no están correctamente finalizadas.
- Presta mucha atención a los permisos de acceso de tus archivos y carpetas.
- Tus tareas serán evaluadas por tus compañeros de la Piscine.
- Todos los ejercicios de terminal deben ejecutarse con `/bin/bash`.
- No debes dejar ningún archivo en tu espacio de entrega salvo aquellos explícitamente solicitados en las instrucciones.
- ¿Tienes una duda? Pregunta a tu compañero de la izquierda. Si no, prueba con el de la derecha.
- Toda respuesta técnica que necesites se encuentra en las páginas `man` o en línea.
- Recuerda usar el foro de la Piscine en tu intranet y Slack!
- Lee los ejemplos detenidamente, pueden contener requisitos que no sean evidentes en la descripción del ejercicio.
- ¡Por Thor, por Odín! ¡Usa tu cerebro!

Capítulo IV

Ejercicio 00: create_array

	Ejercicio: 00
Crear un array	
Directorio de entrega: <i>ex00/</i>	
Archivos a entregar: create_array.py	
Funciones autorizadas: All	

- Crea un programa llamado `create_array.py`.
- Asegúrate de que el programa sea ejecutable.
- Define un array de números.
- Muestra el array en pantalla:


```
?> ./create_array.py | cat -e  
[2, 8, 9, 48, 8, 22, -12, 2]$  
?>
```



Print.

Capítulo V

Ejercicio 01: play_with_arrays


	Ejercicio: 01
Manipulando arrays	
Directorio de entrega: <i>ex01/</i>	
Archivos a entregar: play_with_arrays.py	
Funciones autorizadas: All	

- Crea un programa llamado `play_with_arrays.py`.
- Asegúrate de que el programa sea ejecutable.
- Define un array de números.
- Itera sobre este array, creando un nuevo array sumando 2 a cada valor en el array original.
- Tu programa debe contener dos array: el array original y el array modificado.
- Muestra ambos arrays en pantalla.
- Por ejemplo, si el array original es `[2, 8, 9, 48, 8, 22, -12, 2]`, el resultado debería ser:

```
?> ./play_with_arrays.py
Original array: [2, 8, 9, 48, 8, 22, -12, 2]
New array: [4, 10, 11, 50, 10, 24, -10, 4]
?>
```

Capítulo VI

Ejercicio 02: play_with_arrays++


	Ejercicio: 02
Manipulando arrays, una vez más	
Directorio de entrega: <i>ex02/</i>	
Archivos a entregar: play_with_arrays.py	
Funciones autorizadas: All	

- Modifica tu programa anterior para procesar solo los valores mayores a 5 en el array original.
- Por ejemplo, si tu array original es [2, 8, 9, 48, 8, 22, -12, 2], el resultado debería ser:

```
?> ./play_with_arrays.py | cat -e
[2, 8, 9, 48, 8, 22, -12, 2]$
[10, 11, 50, 10, 24]$
?>
```


Capítulo VII

Ejercicio 03: play_with_arrays+=2

	Ejercicio: 03
¡Manipulando arrays, aún!	
Directorio de entrega: <i>ex03/</i>	
Archivos a entregar: play_with_arrays.py	
Funciones autorizadas: All	

- Modifica tu programa anterior para eliminar los duplicados en el resultado. No debes eliminar explícitamente los valores de tus arrays.
- Por ejemplo, si tu array original es [2, 8, 9, 48, 8, 22, -12, 2], el resultado debería verse de la siguiente manera:


```
?> ./play_with_arrays.py | cat -e
[2, 8, 9, 48, 8, 22, -12, 2]$
{10, 11, 50, 24}$
?>
```



Busca sobre set.

Capítulo VIII

Ejercicio 04: Parámetros

	Ejercicio: 04
¿Sí, pero cuántos?	
Directorio de entrega: <i>ex04/</i>	
Archivos a entregar: parameters.py	
Funciones autorizadas: All	

- Crea un programa llamado `parameters.py`.
- Asegúrate de que este programa sea ejecutable.
- El programa debe mostrar el número de parámetros que se le pasaron, seguido de un salto de línea.

```
?> ./parameters.py
Number of parameters: 0.
?> ./parameters.py "initiation"
Number of parameters: 1.
?> ./parameters.py "this" "is" "crazy" "there's" "everywhere!"
Number of parameters: 5.
?>
```



Consulta la lista `sys.argv` y la función `len()`.

Capítulo IX

Entrega y Evaluación entre Pares

- Debes tener una carpeta llamada `discovery_piscine` en la raíz de tu directorio personal.
- Dentro de `discovery_piscine`, debe existir una carpeta llamada `module5`.
- Dentro de `module5`, debe haber una carpeta para cada ejercicio.
- El Ejercicio 00 debe estar en la carpeta `ex00`, el Ejercicio 01 en `ex01`, y así sucesivamente.
- Cada carpeta de ejercicio debe contener los archivos solicitados en el enunciado.



Durante tu defensa, todo lo que no esté en la carpeta correspondiente para el día no será revisado.