



Discovery Piscine

Módulo4 - Python

Resumen: En este Módulo4, aprenderás a manipular objetos simples

Versión: 2.00

Índice general

I.	Unas palabras sobre esta Discovery Piscine	2
II.	Introducción	3
III.	Instrucciones generales	4
IV.	Ejercicio 00: UPCASE_IT	5
V.	Ejercicio 01: Edad	6
VI.	Ejercicio 02: Calculadora	7
VII.	Ejercicio 03: Decimal	8
VIII.	Ejercicio 04: No Me Gustan las Comas	9
IX.	Ejercicio 05: Mayúsculas y Minúsculas	10
X.	Entrega y Evaluación entre Pares	11

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 manipular objetos simples

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: UPCASE_IT

	Ejercicio: 00
Velo en Mayúsculas	
Directorio de entrega: <i>ex00/</i>	
Archivos a entregar: upcase_it.py	
Funciones autorizadas: All	

- Crea un programa llamado `upcase_it.py`.
- Asegúrate de que el programa sea ejecutable (presta atención a los permisos).
- Este programa debe:
 - Solicitar al usuario que ingrese una palabra.
 - Mostrar la palabra en mayúsculas.
- Ejemplo:


```
?> ./upcase_it.py
Give me a word: banana
BANANA
?>
```



Busca "upcase" en Google.

Capítulo V

Ejercicio 01: Edad

	Ejercicio: 01
Obtener y Modificar un Número	
Directorio de entrega: <i>ex01/</i>	
Archivos a entregar: age.py	
Funciones autorizadas: All	

- Crea un programa llamado **age.py**.
- Asegúrate de que el programa sea ejecutable.
- Este programa debe:
 - Solicitar al usuario que ingrese su edad.
 - Mostrar la edad actual del usuario y su edad en 10, 20 y 30 años.


```
?> ./age.py
Please tell me your age: 15
You are currently 15 years old.
In 10 years, you'll be 25 years old.
In 20 years, you'll be 35 years old.
In 30 years, you'll be 45 years old.
?>
```



Busca "string to integer conversion" en Python.

Capítulo VI

Ejercicio 02: Calculadora


	Ejercicio: 02
Realizando Operaciones Básicas	
Directorio de entrega: <i>ex02/</i>	
Archivos a entregar: calculator.py	
Funciones autorizadas: All	

- Crea un programa llamado `calculator.py`.
- Asegúrate de que este programa sea ejecutable.
- El programa debe:
 - Solicitar al usuario que ingrese dos números.
 - Almacenar estos números como valores numéricos en dos variables.
 - Mostrar el resultado de sumar, restar, dividir y multiplicar estos números.

```
?> ./calculator.py
Give me the first number: 10
Give me the second number: 2
Thank you!
10 + 2 = 12
10 - 2 = 8
10 / 2 = 5
10 * 2 = 20
?>
```


Capítulo VII

Ejercicio 03: Decimal

	Ejercicio: 03
Manejando números de coma flotante	
Directorio de entrega: <i>ex03/</i>	
Archivos a entregar: float.py	
Funciones autorizadas: All	

- Crea un programa llamado `float.py`.
- Asegúrate de que el programa sea ejecutable.
- El programa debe:
 - Solicitar al usuario que ingrese un número.
 - Determinar si el número ingresado es un decimal o no y mostrar el resultado.

```
?> ./float.py
Give me a number: 42
This number is an integer.
?>

?> ./float.py
Give me a number: 42.00
This number is an integer.
?>


?> ./float.py
Give me a number: 42.42
This number is a decimal.
?>
```



Google "string to float".

Capítulo VIII

Ejercicio 04: No Me Gustan las Comas

	Ejercicio: 04
No Me Gustan las Comas	
Directorio de entrega: <i>ex04/</i>	
Archivos a entregar: round_up.py	
Funciones autorizadas: All	

- Crea un programa llamado **round_up.py**.
- Asegúrate de que este programa sea ejecutable.
- Este programa debe:
 - Solicitar al usuario que ingrese un número.
 - Mostrar el número redondeado hacia arriba.


```
?> ./round_up.py
Give me a number: 41.42
42
?>

?> ./round_up.py
Give me a number: 42
42
?>

?> ./round_up.py
Give me a number: 0.001
1
?>
```

Capítulo IX

Ejercicio 05: Mayúsculas y Minúsculas

	Ejercicio: 05
La Fusión de Mayúsculas y Minúsculas	
Directorio de entrega: <i>ex05/</i>	
Archivos a entregar: <i>up_low.py</i>	
Funciones autorizadas: <i>All</i>	

- Crea un programa llamado *up_low.py*.
- Asegúrate de que este programa sea ejecutable.
- Este programa debe:
 - Solicitar al usuario que ingrese un string.
 - Mostrar el string con las letras mayúsculas cambiadas a minúsculas y viceversa.

```
?> ./up_low.py
Hello World
hELLO wORLD
?>
?> ./up_low.py
aaaaAAAA
AAAAaaaa
?>
?> ./up_low.py
hello 42
HELLO 42
?>
```

Capítulo X

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 `module4`.
- Dentro de `module4`, 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.