

python-practica-1

March 15, 2023

1 Python 2023 - Práctica 1

1.1 Objetivos

- Preparar el entorno de trabajo que utilizarán a lo largo de la materia instalando las herramientas básicas.
- Aprender a escribir nuestro primer programa con **Python** y ejecutarlo.
- Crear nuestro primer repositorio local de código y sincronizar el mismo a un repositorio remoto (GitHub).

1.2 Python

Parte principal de esta actividad es tener su entorno de trabajo preparado para trabajar durante la cursada. Por este motivo es fundamental que comencemos por instalar **python** en su máquina.

Este año utilizaremos la versión de **Python 3.11.X**. Para instalar **Python** en su máquina les proveemos [esta guía de instalación](#).

1.2.1 Actividades

1. Realizar la instalación de Python **3.11.X**.
2. Escriba en un archivo llamado `game.py` el siguiente programa en Python.

```
from random import choice, randrange
from datetime import datetime

# Operadores posibles
operators = ["+", "-"]
# Cantidad de cuentas a resolver
times = 5
# Contador inicial de tiempo.
# Esto toma la fecha y hora actual.
init_time = datetime.now()

print(f";Veremos cuanto tardas en responder estas {times} operaciones!")
for i in range(0, times):
    # Se eligen números y operador al azar
    number_1 = randrange(10)
    number_2 = randrange(10)
    operator = choice(operators)
```

```

# Se imprime la cuenta.
print(f"{i+1}- ¿Cuánto es {number_1} {operator} {number_2}?")
# Le pedimos al usuario el resultado
result = input("resultado: ")

# Al terminar toda la cantidad de cuentas por resolver.
# Se vuelve a tomar la fecha y la hora.
end_time = datetime.now()
# Restando las fechas obtenemos el tiempo transcurrido.
total_time = end_time - init_time
# Mostramos ese tiempo en segundos.
print(f"\n Tardaste {total_time.seconds} segundos.")

```

3. Modifique el programa anterior para que:

- Agregar las operaciones multiplicación (*) y división (/) al juego.
- Imprima al lado o debajo de cada respuesta si el resultado del usuario fué correcto o incorrecto.
- Además del tiempo informe al finalizar la cantidad de resultados correctos e incorrectos que tuvo el usuario.

1.3 Git

Como ya vimos **Git** es una herramienta muy buena que nos permite manejar versiones de nuestro código de manera distribuida con nuestro equipo de trabajo.

Para poder realizar esto es necesario contar con un **servidor** de **Git** además de tener la herramienta instalada en su **máquina local**. En esta oportunidad vamos a analizar un poco **GitHub** que es el **servidor** de **Git** más popular actualmente.

Recuerde que se encuentra publicada la [guía básica de Git](#) donde se realizan varias de las tareas que solicitamos para relizar esta entrega.

1.3.1 Actividades

1. Instala **Git** en tu sistema operativo.
2. Arranca un repositorio **local**.
3. Crea tu propio repositorio **remoto** en [Github](#).
4. Agrega el archivo `game.py` con las modificaciones solicitadas a tu repositorio **local**.
5. Agrega el `README.md` con tu nombre y número de alumnx.
6. Sube todos los cambios a tu repositorio **remoto** en Github.
7. Agregar el link del repositorio a la tarea en el curso.

1.4 Pautas

- **Puntos:** 10.
- **Fecha límite de entrega:** Jueves, 23 de marzo de 2023, 23:59
- **Modalidad de entrega:** Copie el enlace de su repositorio remoto con la resolución en la tarea de Cátedras.