



Ciclo 1 Fundamentos de programación

Reto 2 – Sistematizar la información de los estudiantes

Descripción del problema: La era digital ha posibilitado la automatización de muchas prácticas administrativas, haciendo más accesibles y eficientes los procesos que los usuarios deben realizar ante entidades administrativas públicas y privadas. Por otra parte, los sistemas de información posibilitan a estas entidades una recolección de la información más eficiente, efectiva y, en ese sentido, mucho más fácil de gestionar. El Colegio Remedios Catalina Amoroso se ha caracterizado por tener el registro de sus estudiantes en libros de datos y estos son escritos a mano. El ministerio de educación al enterarse de la realidad que padece el Colegio Remedios Catalina Amoroso ha decidido contratar a un Ingeniero de Sistemas para realizar un algoritmo en Python, que ayude a realizar los registros de sus estudiantes. Los estudiantes se caracterizan por tener un código, nombre, edad, promedio y colegiatura. Si el promedio del estudiante es mayor o igual a 4.6 se le realizará un descuento del 20% en la colegiatura y no pagará IVA, pero si el estudiante tiene un promedio menor a 4.6, a la colegiatura se le sumará un IVA del 10%.

Para realizar el ejercicio se realizarán dos funciones, las cuales son:

Primera Función: Esta función recibe los siguientes parámetros: El código del estudiante, el nombre del estudiante, la edad del estudiante, el promedio (Este es calificable de 0-5) y por último a colegiatura. Esta función debe retornar un diccionario con la información previamente mencionada.

Segunda Función: Esta función recibe como parámetro único un diccionario, este puede ser el creado por la Primera función, debe retornar una cadena "Codigo: {Diccionarios['Codigo']}, Nombre: {Diccionarios['Nombre']}, Edad: {Diccionarios['Edad']}, Promedio: {Diccionarios['Promedio']}, Colegiatura: {Diccionarios['Colegiatura']}", ya con el descuento o aumento en la colegiatura.



Ejemplos:

Primera Función:

Código	Nombre	edad	Promedio	Colegiatura	Return
3001245	Santiago	20	4.0	10000000	{'Codigo': 3001245, 'Nombre': 'Santiago', 'Edad': 20, 'Promedio': 4.0, 'Colegiatura': 10000000}
3599392	Andres	17	4.8	23488000	{'Codigo': 3599392, 'Nombre': 'Andres', 'Edad': 17, 'Promedio': 4.8, 'Colegiatura': 23488000}
3000045	Carol	19	3.2	30000000445	{'Codigo': 3000045, 'Nombre': 'Carol', 'Edad': 19, 'Promedio': 3.2, 'Colegiatura': 30000000445}

Entrada:

Nombre	Tipo	Descripción
Codigo	int	El identificador del estudiante
Nombre	Str	Nombre del estudiante
edad	int	La edad del estudiante
promedio	float	El promedio del estudiante
Colegiatura	float	La colegiatura del estudiante al que se le aplicará descuento o suma de IVA.

Salida:

Tipo de retorno	Descripción
dic	{'Codigo': Codigo, 'Nombre': Nombre, 'Edad': edad, 'Promedio': promedio, 'Colegiatura': Colegiatura }

Esqueleto:

```
def CrearDiccionario(Codigo: int, Nombre: str, Edad : int, promedio: float, Colegiatura: float) -> dict:
```



Segunda Función:

Diccionarios	Return
{'Codigo': 3001245, 'Nombre': 'Santiago', 'Edad': 20, 'Promedio': 4.0, 'Colegiatura': 10000000}	Código: 3001245, Nombre: Santiago, Edad: 20, Promedio: 4.0, Colegiatura: 11000000.0
{'Codigo': 3599392, 'Nombre': 'Andres', 'Edad': 17, 'Promedio': 4.8, 'Colegiatura': 23488000}	Código: 3599392, Nombre: Andres, Edad: 17, Promedio: 4.8, Colegiatura: 18790400.0
{'Codigo': 3000045, 'Nombre': 'Carol', 'Edad': 19, 'Promedio': 3.2, 'Colegiatura': 30000000445}	Código: 3000045, Nombre: Carol, Edad: 19, Promedio: 3.2, Colegiatura: 330000000489.5

Entrada:

Nombre	Tipo	Descripción
Diccionario	vect	Los diccionarios previamente retornados por la Primera Función

Salida:

Tipo de retorno	Descripción
dict	"Codigo: {Diccionarios['Codigo']}, Nombre: {Diccionarios['Nombre']}, Edad: {Diccionarios['Edad']}, Promedio: {Diccionarios['Promedio']}, Colegiatura: {Diccionarios['Colegiatura']}"

Esqueleto:

```
def AplicarColegiatura(Diccionarios: dict)->str:
```