# **Estructura de datos tipo Tupla**

Hemos desarrollado gran cantidad de algoritmos empleado tipos de datos simples como enteros, flotantes, cadenas de caracteres y estructuras de datos tipo lista.

Vamos a ver otra estructura de datos llamada tupla.

Una tupla permite almacenar una colección de datos no necesariamente del mismo tipo. Los datos de la tupla son inmutables a diferencia de las listas que son mutables.

Una vez inicializada la tupla no podemos agregar, borrar o modificar sus elementos.

Utilizamos una tupla para agrupar datos que por su naturaleza están relacionados y que no serán modificados durante la ejecución del programa.

La sintaxis para definir una tupla es indicar entre paréntesis sus valores:

Tupla = (1,2,3)

Fecha=(25, “Diciembre”, 2016)

Punto = (10,2)

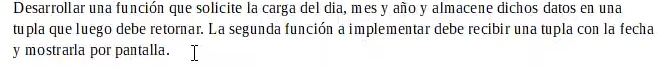
Persona = (“Rodriguez”,”Pablo”,43)

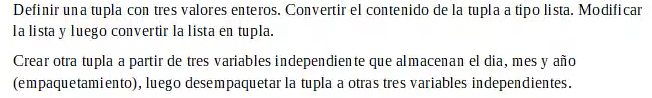
**Problema 1**

****

tupla = (1,2,3)  
fecha = (25, "Diciembre",2016)  
punto = (10,2)  
persona = ("Ridriguez""Pablo",43)  
  
print(tupla)  
print(fecha)  
print(punto)  
print(persona)

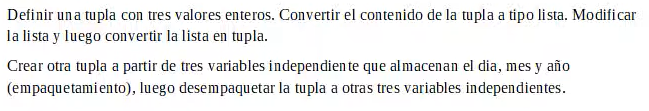
**Problema 2**

****

****

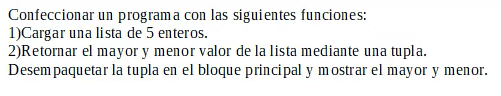
def CargarFecha():  
 dia = int(input("Ingresar dia: "))  
 mes = int(input("Ingresar mes: "))  
 año = int(input("Ingresar año: "))  
 return (dia, mes, año)  
  
def ImprimirFecha(fecha):  
 print(fecha[0],fecha[1],fecha[2],sep="/")  
  
fecha = CargarFecha()  
ImprimirFecha(fecha)  
lista = list(fecha)

**Problema 3:**

****

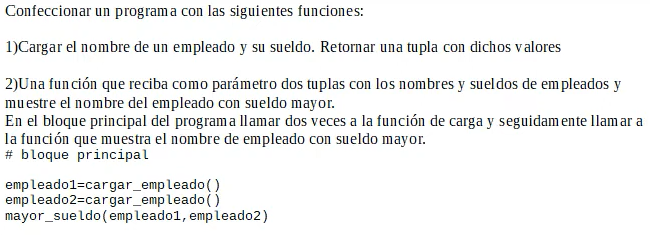
fechatupla1 = (1,5,2020)  
print("Imrprimir la tupla")  
print(fechatupla1)  
fechatupla1 = list(fechatupla1)  
print("Imprimimos la lista que se copip de la tupla")  
print(fechatupla1)  
fechatupla1[0]=25  
print("Imprimimos la lista modificada")  
print(fechatupla1)  
  
fechatupla2 = tuple(fechatupla1)  
print("Imprimimos la segunda tupla que se le copio la lista")  
print(fechatupla2)  
  
dia = 10  
mes = 5  
año = 2020  
#Empaquetar  
fechatupla3 = dia,mes, año  
#Desempaquetar  
dia1, mes1, año1 = fechatupla3  
dia1 = 30  
mes1 = 12  
año1 = 1998  
fechatupla3 = dia1,mes1,año1  
print(fechatupla3)

**Problema 4:**

****

def cargar():  
 lista = []  
 for x in range(5):  
 valor = int(input("Ingrese valor: "))  
 lista.append(valor)  
 return lista  
def MayorMenor(n):  
 mayor = 0  
 menor = n[0]  
 for x, valor in enumerate(n):  
 if valor>mayor:  
 mayor = valor  
 elif valor<menor:  
 menor = valor  
 return (mayor, menor)  
lista = cargar()  
mayor, menor = MayorMenor()  
print("El valor maximo es: ",mayor)  
print("El valor minimo es: "menor)

**Problema 5:**

****

# **Listas y tuplas anidadas**

Hemos visto dos estructuras de datos fundamentales en Python que son las listas y las tuplas.

La lista es una estructura mutable (es decir podemos modificar sus elementos, agregar y borrar) en cambio una tupla es una secuencia de datos inmutable, es decir una vez definida no puede cambiar.

En Python vimos que podemos definir elementos de una lista que sean de tipo lista, en ese caso decimos que tenemos una lista anidada.

Ahora que vimos tuplas también podemos crear tuplas anidadas.

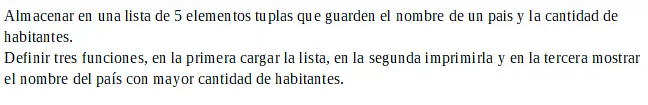
En general podemos crear y combinar tuplas con elementos de tipo lista y viceversa, es decir listas con componente de tipo tupla.

**Problema 1:**

****

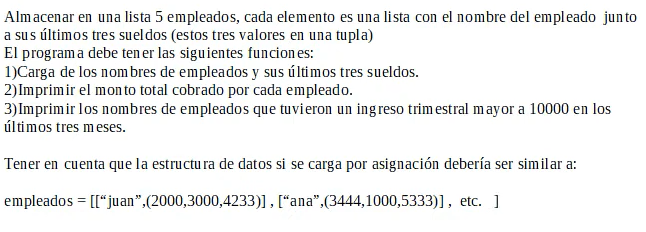
#Lista anidada  
empleado1 = ["Ana",45,(1,1,2020)]  
print(empleado1)  
empleado1.append((1,5,2020))  
print(empleado1)  
#Tupla anidada  
alumno = ("Juan",[8,5])  
print(alumno)  
alumno[1].append(10)  
print(alumno)

**Problema 2:**

****

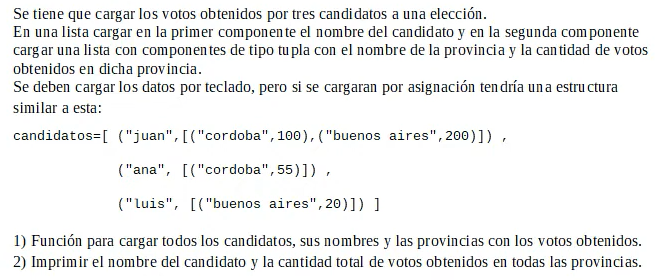
def CargarPais():  
 paises = []  
 for x in range(5):  
 nombre = input("Ingrese el nombre del pais: ")  
 cantidad = int(input("Ingrese la cantidad: "))  
 paises.append((nombre,cantidad))  
 return paises  
def Imprimir(paises):  
 print("Paises y su poblacion")  
 for x, valor in enumerate(paises):  
 print(valor[0],valor[1])  
def MayorMenor(paises):  
 mayor = paises[0][1]  
 nombremayor = paises[0][0]  
 for i, valor in enumerate(paises):  
 if valor[1]>mayor:  
 mayor = valor[1]  
 nombremayor = valor[0]  
 print(f"El pais con mas habitantes es {nombremayor} que son {mayor}")  
  
  
paises = CargarPais()  
Imprimir(paises)  
MayorMenor(paises)

**Problema 3:**

****

def CargarEmpleados():  
 empleados = []  
 for x in range(5):  
 nombre = input("Ingrese el nombre del empleado: ")  
 sueldo1 = int(input("Ingrese el primer sueldo: "))  
 sueldo2 = int(input("Ingrese el segundo sueldo: "))  
 sueldo3 = int(input("Ingrese el tercer sueldo: "))  
 empleados.append([nombre,(sueldo1,sueldo2,sueldo3)])  
 return empleados  
def Ganancias(empleados):  
 print("Monto total ganado por empleado en los ultimos 3 meses")  
 for i in range(5):  
 total = empleados[i][1][0]+empleados[i][1][1]+empleados[i][1][2]  
 print(empleados[i][0],total)  
def GananciaSuperior(empleados):  
 print("Empleados con ingresos superiores a 10000 en los ultimos 3 meses")  
 for i in range(5):  
 total = empleados[i][1][0]+empleados[i][1][1]+empleados[i][1][2]  
 if total >10000:  
 print(empleados[x][0],total)  
  
empleados = CargarEmpleados()  
Ganancias(empleados)  
GananciaSuperior(empleados)

**Problema 4:**

****

def CargarDatos():  
 candidatos = []  
 for i in range(3):  
 nombre = input("Ingrese nombre: ")  
 cant = int(input("Los votos de cuantas provincias tiene que cargar: "))  
 provincias = []  
 for x in range(cant):  
 prov = input("Nombre de la provincia: ")  
 votos = int(input("Cantidad de votos: "))  
 provincias.append((prov,votos))  
 candidatos.append((nombre,provincias))  
 return candidatos  
  
def TotalVotos(n):  
 for i in range(len(n)):  
 suma = 0  
 for x in range(len(n[i][1])):  
 suma = suma + n[i][1][x][1]  
 print(n[x][0],suma)  
  
candidatos = CargarDatos()  
votos = TotalVotos(candidatos)

# **Variantes de la estructura repetitiva for para recorrer tuplas y listas**

Hasta ahora siempre que recorremos una lista o una tupla utilizando un for procedemos de la siguiente manera:

Lista = [2,3,50,7,9]  
for x in range(len(lista)):  
 print(lista[x])

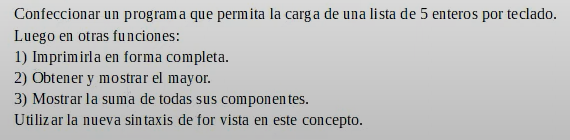
Esta forma de recorrer la lista es utilizada obligatoriamente cuando queremos modificar sus elementos como podría ser:

Lista = [2,3,50,7,9]  
for x in range(len(lista)):  
 if lista[x]<10:  
 lista[x]=0

Ahora veremos una segunda forma de acceder a los elementos de una lista con la estructura repetitiva for sin indicar subíndices.  
lista = [2,3,50,7,9]  
for elementos in lista:  
 print(elemento)

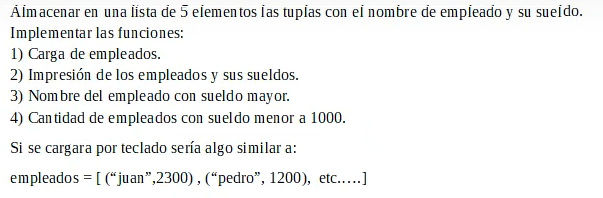
Como podemos ver la instrucción for requiere una variable (en este ejemplo llamada elemento), luego la palabra clave in y por ultimo el nombre de la lista. El bloque del for se ejecuta tantas veces como elementos tenga la lista, y en cada vuelta del for la variable elemento almacena un valor de la lista.

**Problema 1:**

****

def Carga():  
 lista = []  
 for i in range(5):  
 valor = int(input("Ingresa: "))  
 lista.append(valor)  
 return lista  
def Imprimir(numeros):  
 for i in numeros:  
 print(i)  
def mayordef(numeros):  
 mayor = numeros[0]  
 for i in numeros:  
 if i>mayor:  
 mayor = i  
 return mayor  
def Sumadef(numeros):  
 suma = 0  
 for i in numeros:  
 suma+=i  
 return suma  
numeros = Carga()  
mayor = mayordef(numeros)  
suma = Sumadef(numeros)  
print(f"El mayor es: {mayor}")  
print(f"La suma de todos los elementos es {suma}")

**Problema 2**

****

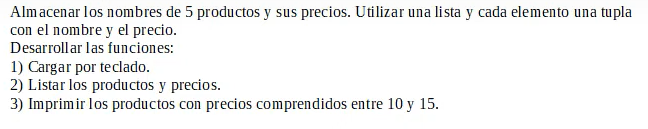
def Cargar():  
 empleado = []  
 for i in range(5):  
 nombre = input("Ingrese el nombre del empleado: ")  
 sueldo = int(input("Ingrese el sueldo del empleado: "))  
 empleado.append((nombre,sueldo))  
 return empleado  
  
def Imprimir(empleados):  
 print("Listado de los nombres de empleados y sus sueldos")  
 for nombre, sueldo in empleados:  
 print(nombre,sueldo)  
def MayorSueldo(empleados):  
 empleado = empleados[0]  
 for emp in empleados:  
 if emp[1]>empleado[1]:  
 empleado = emp  
 print(f"Empleado con mayor sueldo es {empleado[0]} que corresponde a {empleado[1]}")  
def SueldoMenor10000(empleados):  
 cant = 0  
 for empleado in empleados:  
 if empleado[1]<10000:  
 cant+=1  
 print(f"Cantidad de empleados con sueldo menor a 1000 son: {cant}")

**Problema 3**

****

def Cargar():  
 palabras = []  
 cant = int(input("Cuantas palabras quiere cargar?: "))  
 for x in range(cant):  
 pal = input("Ingrese la palabra: ")  
 palabras.append(pal)  
 return palabras  
def palabras\_mas5(palabras):  
 print("Palabras ingresadas con mas de 5 caracteres")  
 for palabra in palabras:  
 if len(palabra)>5:  
 print(palabra)  
palabras = Cargar()  
palabras\_mas5(palabras)

**Problema 4:**

****

def Cargar():  
 productos = []  
 for x in range(5):  
 nombre = input("Ingrese el nombre del producto: ")  
 precio = int(input("Ingrese el precio del producto: "))  
 productos.append((nombre,precio))  
 return productos  
def Imprimir(productos):  
 print("Listado de productos y precios")  
 for nombre, precio in productos:  
 print(nombre, precio)  
def Rango(productos):  
 for nombre, precio in productos:  
 if precio>=10 and precio<=15:  
 print(nombre,precio)