

Compresion

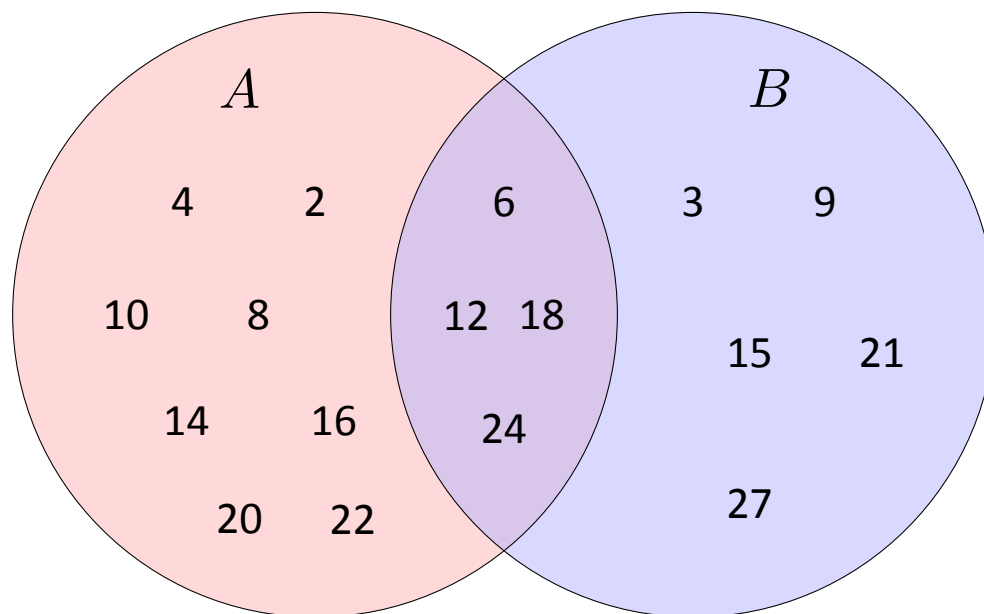
CS1100 - Introducción a Ciencia de la Computación
UTEC

Logro de la Sesión

Al finalizar la unidad, estaras en la capacidad de:

Utilizar adecuadamente los diferentes tipos de comprehensions.

Notaciones matemáticas para definir un conjunto:



Respecto a estos conjuntos:

En matemáticas existen 2 notaciones para definir un conjunto; (a) por extensión, (b) por comprensión.

- 1 ¿Cómo definiría por extensión el conjunto A? y ¿Cómo lo definiría por comprensión?
- 2 ¿Cómo definiría por extensión el conjunto B? y ¿Cómo lo definiría por comprensión?
- 3 ¿Cómo definiría por comprensión el conjunto $(A \cap B)$?

Notaciones matemáticas para definir un conjunto:

Solución:

1 Problema 1:

1 $A = \{2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24\}$

2 $A = \{x | x \in N, x \text{ es par}, x \geq 2 \wedge x \leq 24\}$

2 Problema 2:

1 $B = \{3, 6, 9, 12, 15, 18, 21, 24, 27\}$

2 $B = \{x | x \in N, x \text{ es multiplo de 3}, x \geq 3 \wedge x \leq 27\}$

3 Problema 3:

1 $(A \cap B) = \{x | x \in N, x \text{ es par}, x \text{ es multiplo de 3}, x \geq 2 \wedge x \leq 27\}$

2 $(A \cap B) = \{x | x \in A \wedge x \in B\}$

¿Qué es Comprensión en Python?

Es la instrucción que permite de forma concisa la construir de una colección secuencial.

***Comprension** se basa en la notación por comprensión de conjuntos. Desde la perspectiva computacional **Comprension** utiliza un estilo declarativo y funcional. Python soporta 4 tipos de comprension:*

- Comprension de listas
- Comprension de diccionarios
- Comprension de conjuntos
- Comprension de generadores

Sintaxis de List Comprehesion

```
1 lista = [expresion for item in iterable]
2 lista = [expresion for item in iterable if condition]
```

Ejemplo:

```
1 A = [x for x in range(2,25,2)]
2 B = [x for x in range(3,28,3)]
3 print(A)
4 print(B)
```

Ejemplos

- 1 Escribir un programa que genere la lista de numeros entre 0 y 50 que no sean pares y que no sean multiplos de 3
- 2 Escribir un programa que genere la lista de 20 numeros aleatorios entre 1 y 15
- 3 Dado el siguiente texto:

```
1 texto = "10,20, 33, 40, 11, 90"
```

Escribir un programa que genere una lista con 6 valores numericos incluidos en el texto y luego calcular la suma de aquellos valores multiplos de 10.

Soluciones

1 Problema 1:

```
1 lista1 = [x for x in range(0, 50) if x % 2 != 0 and x % 3 != 0]
2 print(lista1)
```

2 Problema 2:

```
1 from random import randint
2
3 lista3 = [randint(1, 15) for _ in range(20)]
4 print(lista3)
```

3 Problema 3:

```
1 texto = "10,20, 33, 40, 11, 90"
2
3 lista = [int(x) for x in texto.split(',')]
4 print(lista)
5
6 suma = sum([x for x in lista if x % 10 == 0])
7 print(suma)
```


List Comprension Anidado

```
1 lista = [expresion for item1 in iterable for item2 in iterable]
```

Dados el conjunto C, definir una lista que sea la suma de las filas y calcular el total de todos los valores.

```
1 # Matriz: Lista de Lista
2 C = [ [2, 3, 4],
3       [40, 50, 60],
4       [100, 200, 300] ]
```

Solución de List Comprehension

```
1  # suma por filas
2  sum_row1 = [sum(x) for x in C]
3  print(sum_row1)
4  # equivalente usando for
5  sum_row2 = []
6  for row in C:
7      subtotal = sum(row)
8      sum_row2.append(subtotal)
9  print(sum_row2)
10 # suma total
11 total1 = sum([x for row in C for x in row])
12 print(total1)
13 # equivalente usando for
14 total2 = 0
15 for row in C:
16     for x in row:
17         total2 += x
18 print(total2)
```

List Comprehension para el ingreso de datos

Escribir un programa que permita ingresar n números y genere una lista:

```
1  # llamada usando for
2  n = int(input("cantidad # 1: "))
3  lista1 = []
4  for i in range(n):
5      value = int(input("ingrese {0}:".format(i+1)))
6      lista1.append(value)
7  # llamada concisa usando comprehesion
8  n = int(input("cantidad # 2: "))
9  lista2 = [int(input("ingrese {0}:".format(i+1))) for i in range(n)]
10 # llamada mas concisa
11 lista3 = [int(input("ingrese {0}:".format(i+1))) for i in range(int(
12     input("cantidad # 3: ")))
13 print(lista3)
14 # mostrando listas
15 print("Lista 1 :", lista1)
16 print("Lista 2 :", lista2)
17 print("Lista 3 :", lista3)
```

Evaluación

Responda las siguientes preguntas:

- ¿Qué es Comprehension?
- ¿Cual es la ventaja de comprehension?
- Respecto al tipo de colección ¿Existe comprehension para tuplas?

Cierre

En esta sesión se ha aprendido y concluye:

- Una forma concisa de declarar listas por medio de List Comprehension.
- Comprehension se basa en la notación por comprensión de conjuntos.
- Un programa conciso y claro permite reducir los errores de codificación.