

L U M B R E

```
twentyfifteen_setup
register_setup_theme', 'twentyfifteen_setup' );

// Create the widget area.
function twentyfifteen_widgets_init() {
    register_sidebar( array(
        array(
            'name' => __( 'Widget Area', 'twentyfifteen' ),
            'id' => 'sidebar-1',
            'description' => __( 'Add widgets here to appear in the sidebar.', 'twentyfifteen' ),
            'before_widget' => '<aside id="%1$s" class="widget %2$s">',
            'after_widget' => '</aside>',
            'before_title' => '<h2 class="widget-title">',
            'after_title' => '</h2>',
        ),
    ) );
}
```

Indexación avanzada y máscaras booleanas

Grado: 2

Materia: Programación con Python para Datos · Programación

11/10/2026

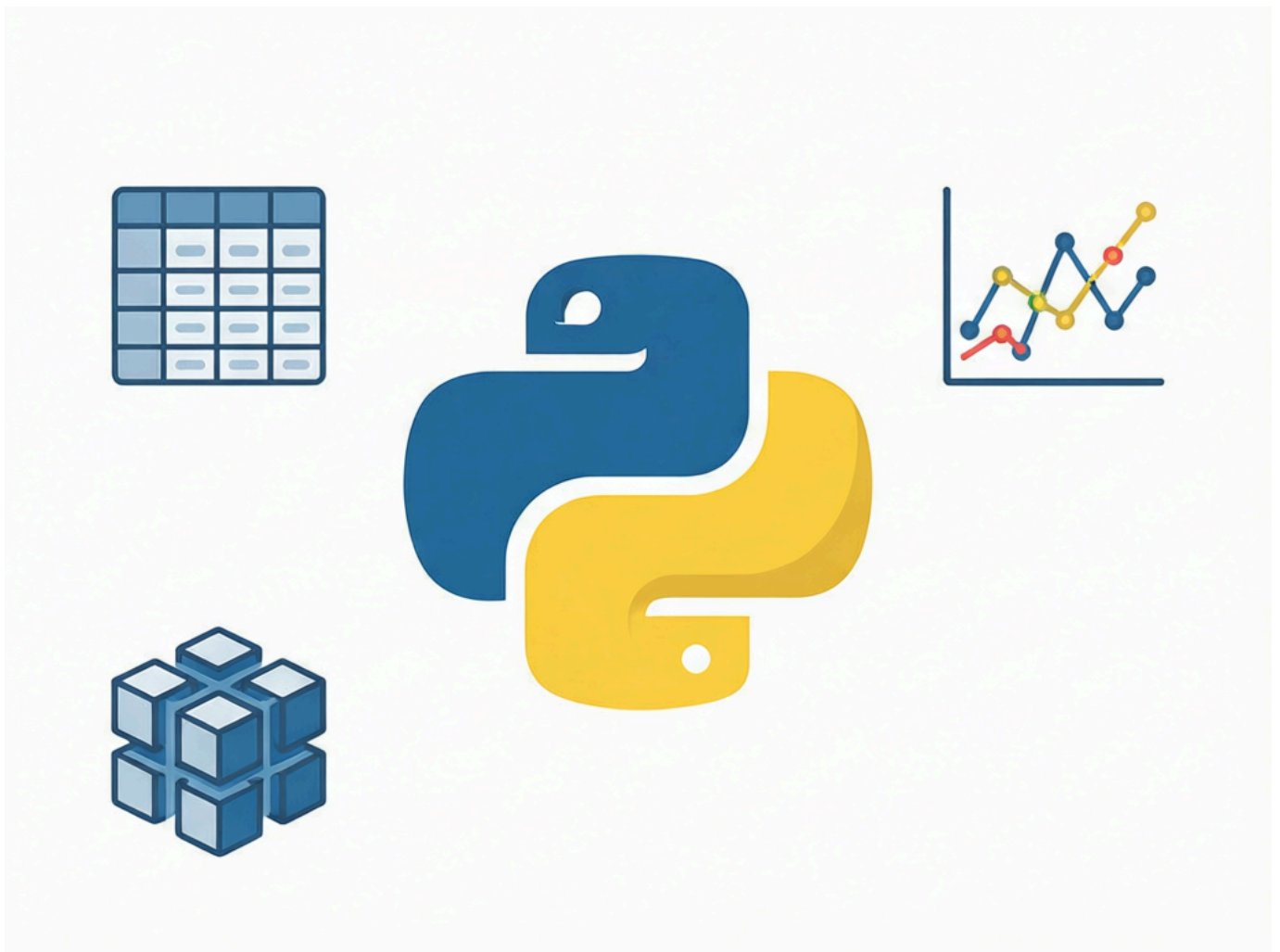
Antes de empezar

Más allá del `a[0]`, NumPy permite seleccionar subconjuntos con condiciones: una máscara booleana marca qué elementos conservar. Dominar esto es la base del filtrado en Pandas.

QUÉ APRENDERÁS

- Seleccionar y filtrar elementos de arrays con índices booleanos, índices de posición y máscaras compuestas.

Indexación avanzada y máscaras booleanas



Python para ciencia de datos: NumPy, pandas y manipulación de datos.

Mapa conceptual

- **Python para Datos**
 - Entorno
 - pip

- venv
- Reproducibilidad
- Funcional
 - Comprensiones
 - Lambda
 - map filter
- Calidad
 - try except
 - pytest
 - Asserts
- NumPy
 - Arrays
 - Broadcasting
 - Indexacion
- Pandas
 - DataFrame
 - Groupby
 - Merge Join

El mapa mental interactivo es interactivo en la versión online.



Seleccionar con condiciones

Filtrar «todos los valores mayores que cero» sin un bucle es una de las operaciones más usadas en datos. La herramienta es la máscara booleana: un array de verdaderos y falsos que decide qué pasa.

Indexación básica

Un array se indexa por posición como las listas: `a[2]` toma el tercer elemento; una multidimensional usa coma, `m[0, 1]`. Los rebanados (slices) con dos puntos devuelven vistas, no copias, así que un cambio en la vista afecta al original si no lo deseas.

Array booleano como máscara

Una comparación como `a > 0` no devuelve un sí o no, sino un array de booleanos del mismo tamaño. Al usar ese array para indexar, `a[a > 0]`, NumPy devuelve solo los

elementos donde la condición era verdadera. Esto es el filtrado puro.

Combinar condiciones

- Usa operadores de bits y paréntesis: (a 0) y (a 10) para un intervalo.
- y es Y, es O, ~ es NO. Los símbolos and/or de Python no funcionan sobre arrays.
- Los paréntesis son obligatorios: la precedencia sin ellos rompe la expresión.

np.where

where(condicion, valor_si, valor_no) construye un nuevo array eligiendo elemento a elemento según la máscara, sin bucles. Es la versión vectorizada del «si esto, aquello, de lo contrario esto otro», perfecta para crear columnas derivadas.

Índices de posición (fancy indexing)

En lugar de una condición, pasas un array de enteros con las posiciones que quieres: a[[0, 2, 5]] toma esos tres elementos, en ese orden y permitiendo repetir índices. Sirve para reordenar o remuestrear.

Cada forma de indexar responde a una pregunta distinta sobre los datos.

Modo	Sintaxis (idea)	Devuelve
Por posición	a[0], m[1, 2]	un elemento escalar
Rebanado	a[1:4]	una vista del segmento
Máscara booleana	a[a 0]	copia de los que cumplen
Fancy index	a[[0, 2, 5]]	copia en el orden dado

1. ¿Qué devuelve la expresión a[a 0] en un array NumPy?

- ☐ Un único booleano
- ☐ Una copia con los elementos que cumplen la condición
- ☐ Los índices de los que cumplen
- ☐ El array ordenado

Asignar con máscara

Las máscaras también escriben: a[a 0] = 0 pone a cero todos los negativos de golpe. Este patrón de «limpiar» valores inválidos es constantísimo al preparar datos antes de un análisis.

Contar y resumir con máscaras

Como los booleanos son 0 y 1 al sumarlos, a 0 seguido de `.sum()` cuenta cuántos cumplen la condición, y `.mean()` da la proporción. Sin bucles, obtienes la frecuencia de un evento sobre todo el array.

Cuentas de verdad

`np.count_nonzero` y `np.any / np.all` responden a preguntas como «¿hay algún valor ausente?» o «¿están todos ordenados?». Combinadas con máscaras cubren la mayoría de validaciones rápidas sobre un array.

Errores frecuentes

- Usar `and/or` en lugar de `y sobre` arrays: lanza un error de ambigüedad.
- Olvidar los paréntesis al combinar condiciones y obtener resultados inesperados.
- Esperar que un rebanado sea una copia y terminar modificando el original por accidente.

2. Para unir dos condiciones sobre arrays se emplean los operadores de Python `and` y `or`.

☐ Verdadero

☐ Falso

Del array al DataFrame

Este mismo mecanismo de filtrado es el que usarás en Pandas cada vez que escribas `df[df.ventas > 100]`. Entender la máscara booleana sobre arrays te da la base de todo el filtrado de tablas.

3. Clasifica cada expresión:

Elementos:

- `a[a 0]`
- `a[[3, 0, 7]]`
- `a[1:5]`
- `a[0, 1]`

Mascara	. . .
Fancy	. . .
Rebanado	. . .
Basica	. . .

NumPy Docs <https://numpy.org/doc/stable/user/basics.indexing.html> Guía oficial de indexación, rebanado e indexación avanzada.

VIDEO

Video complementario

Recurso audiovisual para reforzar los conceptos.



Escanea para verlo en Lumbre

<https://www.youtube.com/watch?v=rfscVS0vtbw>

4. Dado un array de calificaciones, describe con máscaras cómo obtendrías: (1) solo los aprobados, (2) cuántos suspensos hay y (3) poner los negativos a cero. ¿Por qué evitas un bucle?

Documentación pandas <https://pandas.pydata.org/docs/> Guía de referencia para manipulación de datos en Python.

Referencia curricular

Grado:	2
Materia:	Programación con Python para Datos · Programación

Versión interactiva completa (con actividades, videos y autoevaluación):



<https://univ.lumbre.lat/object/cdia-pyth-005-indexacion-avanzada-y-mascaras-booleanas>

Material educativo generado desde Lumbre · lumbre.lat