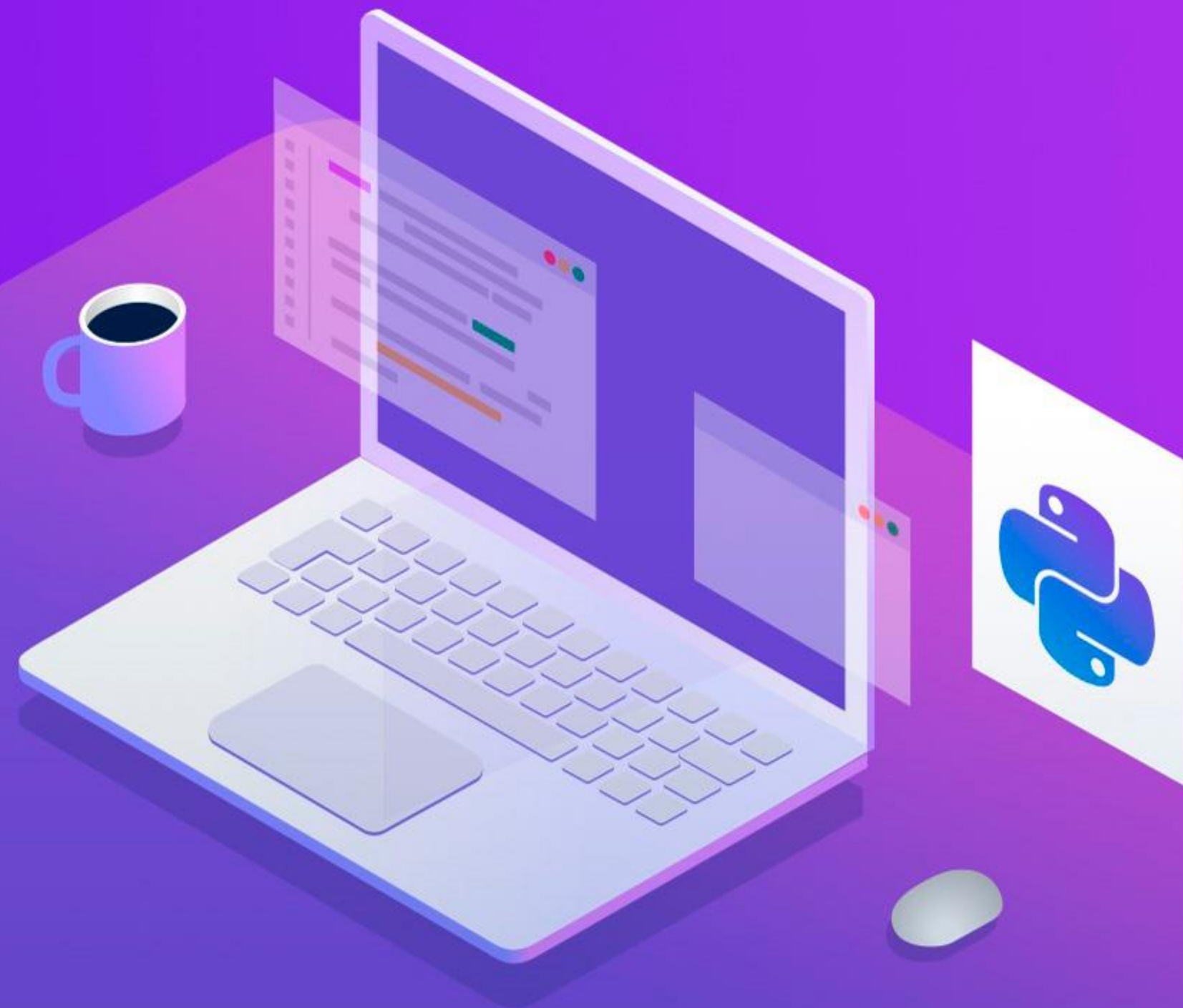




PONTIFICIA
UNIVERSIDAD
CATÓLICA
DE CHILE

DATOS Y LISTAS DE LISTAS



Recordemos

Ya vimos cómo operar con variables, además de la creación de funciones. Además, vimos funciones para las operaciones de `strings` y listas. Estas últimas nos permiten almacenar más de un dato en una variable, y de esa forma “simular” situaciones cotidianas de la vida real.

De forma general, estas funcionalidades son la base para poder hacer operaciones más complejas, y lo más importante, acercarnos a manejar herramientas concretas que permitan procesar datos.



>>> **Datos**

Datos

Características

- Rara vez se almacena "un solo tipo" de datos.
- La gran mayoría de los datos en el mundo laboral se almacenan en bases de datos.
- La definición de las bases de datos escapa el alcance de este curso. Sin embargo, es importante notar que estas almacenan datos de distinta naturaleza que están relacionados.

EJEMPLO



Es muy probable que todas las empresas tengan una base de datos "Empleados". Esta base de datos tendrá la siguiente información:

- Nombre
- Edad
- Fecha de Nacimiento
- RUT

Datos

Respecto a la base de datos podemos destacar dos cosas:

1

Se constituye por conjunto de datos. Por ejemplo, la "característica" "Nombre", contendrá los nombres de todos los empleados de la empresa.

2

Las "características" son de distinto tipo. Nombre es de tipo texto, edad es de tipo número, fecha de nacimiento es de tipo fecha.

>>> Listas de listas

Listas de listas

Pensemos, en la siguiente situación:

Si quisiéramos, almacenar una de las características de los empleados de nuestra empresa en Python, podríamos usar una **lista**.

Si quisiéramos, “simular” la base de datos de empleados completa en Python, podríamos tener **varias listas**. Una para los nombres, otra para las edades, otra para las fechas de nacimiento, y otra para los RUT.

Listas de listas

No obstante, esto no es eficiente ¿imagina si tuviéramos 50 características de las personas de nuestra empresa, almacenadas en una base de datos?

Claramente tener 50 listas distintas en nuestro código no es la solución óptima.

Importante

Para esto existen las listas de listas. Así como podemos crear listas con datos de tipo `int`, `float`, `string`, `bool`, u otros, también podemos crear una lista que almacene otras listas.



Listas de listas

Podemos definir una lista de listas de la siguiente manera:

```
matriz = [["a", "b", "c"], ["d", "e", "f"], ["g", "h", "i"]]
```

La variable matriz

El primer elemento, es una lista que contiene los elementos a, b, y c.

Los tres elementos

Cada una de estas listas tiene a su vez tres elementos.

Listas de listas

Podemos definir una lista de listas de la siguiente manera:

```
matriz = [ ["a", "b", "c"], ["d", "e", "f"], ["g", "h", "i"] ]
```

La variable `matriz` es una lista que contiene tres elementos.

Los tres elementos son:

El segundo elemento, es una lista que contiene los elementos d, e, y f.

Cada una de estas listas tiene a su vez tres elementos.

Listas de listas

Podemos definir una lista de listas de la siguiente manera:

```
matriz = [ ["a", "b", "c"], ["d", "e", "f"], ["g", "h", "i"] ]
```

La variable matriz es una

Los tres elementos son l

Cada una de estas listas tiene a su vez tres elementos.

El tercer elemento, es una lista que contiene los elementos g, h, e i.

Listas de listas

Detengámonos para analizar la variable matriz

- El nombre de esta variable no es azaroso.
- Se asemeja a una matriz, es decir, tendrá filas y columnas.
- Esta lista de listas puede representarse por la siguiente figura:

a	b	c
d	e	f
g	h	i

Listas de listas

¿Qué significa que una matriz tenga filas y columnas?

Significa, que es posible encontrar elementos de esta matriz, sabiendo en qué fila y en qué columna están. En este caso, podemos enumerar las filas y columnas de 0 al 2.

a	b	c
d	e	f
g	h	i

Se enumeran las filas y columnas del 0 al 2 ya que recordemos que en Python se empieza a contar desde el 0.

Listas de listas

¿Qué significa que una matriz tenga filas y columnas?

Significa, que es posible encontrar elementos de esta matriz, sabiendo en qué fila y en qué columna están. En este caso, podemos enumerar las filas y columnas del 0 al 2.

	0	1	2
0	a	b	c
1	d	e	f
2	g	h	i

Por ejemplo:

Si quisiéramos obtener el elemento "f", sabemos que está en la fila 1 y la columna 2.

¿Cómo podríamos hacer el mismo ejercicio en Python?

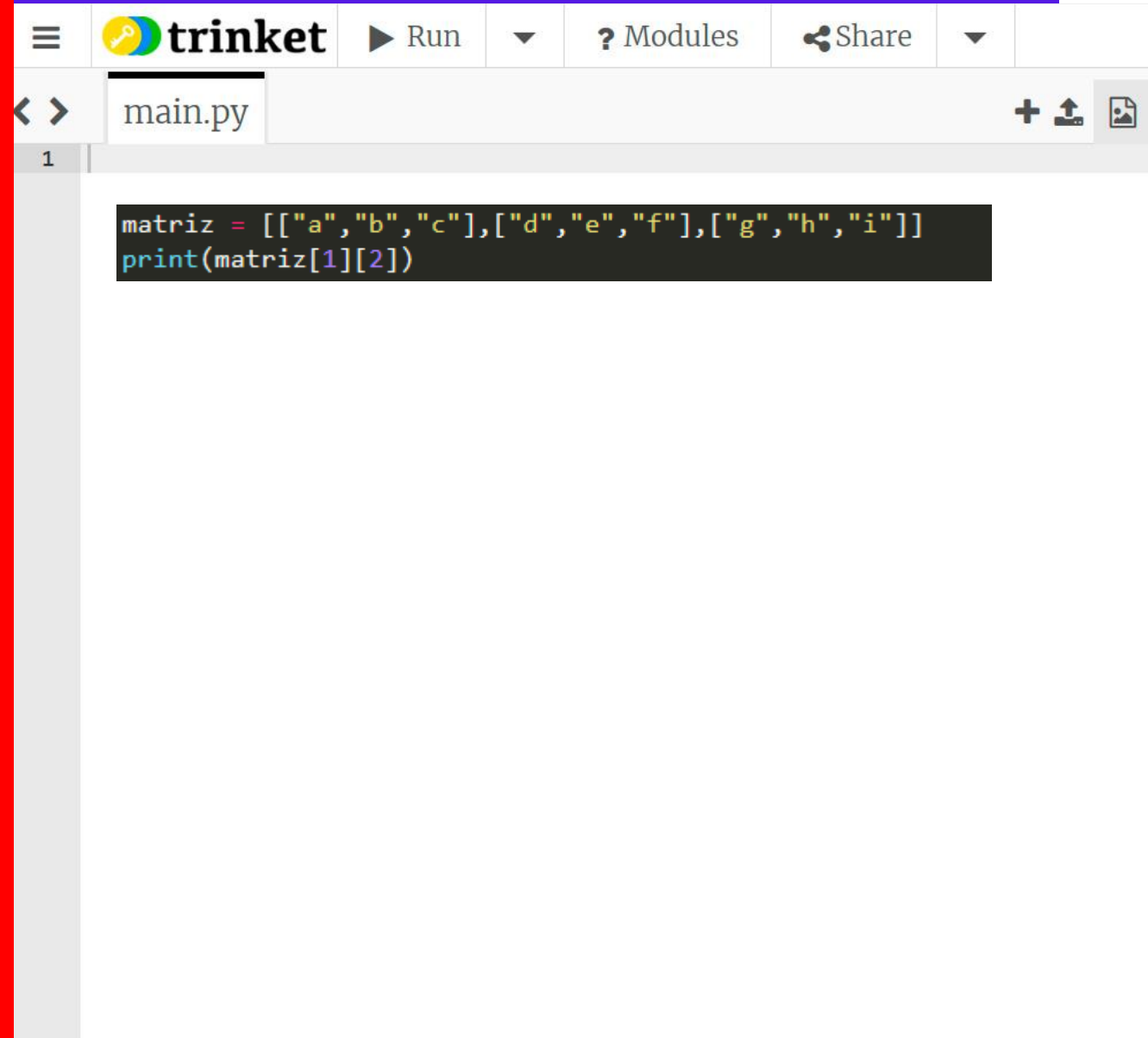
De forma general, podemos obtener un elemento de una variable M que representa a una lista de listas de la siguiente manera:

```
M[i][j]
```

Donde i y j representan a una fila y a una columna, específica de la lista de listas (recordemos que las filas y columnas se empiezan a contar desde 0).

¿cómo podríamos hacer el mismo ejercicio en Python?

Recuerda revisar la Ruta de ejercicios.
Ejercicio EM3-01 →



```
matrix = ["a","b","c"],["d","e","f"],["g","h","i"]
print(matrix[1][2])
```

'f'

Revisemos desde otra perspectiva el caso

Pensemos en qué podríamos hacer.

¿Si quisiéramos cambiar el elemento que tiene el valor “h” por “z”?

De acuerdo a la tabla:


	0	1	2
0	a	b	c
1	d	e	f
2	g	h	i

“h” está en la fila 2 y la columna 1.

Para cambiar el elemento que tiene "h" por "z",
debemos hacer lo siguiente

Recuerda revisar la
Ruta de ejercicios.

Ejercicio EM3-02 →



```
matriz = [['a','b','c'],['d','e','f'],['g','h','i']]
matriz[2][1] = "z"
print(matriz)
```

```
[['a', 'b', 'c'], ['d', 'e', 'f'], ['g', 'z', 'i']]
```

Podemos notar que en la matriz inicial
el elemento que estaba en la fila 2 y
columna 1 (que era inicialmente "h"),
cambió a "z" luego de asignárselo.
Notemos que "z" sobrescribió a "h".

Listas de listas

De forma general, podemos modificar un elemento de una variable M que representa a una lista de listas de la siguiente manera:

$$M[i][j] = k$$

Donde i y j representa a una fila, y a una columna específica de la lista de listas (recordemos que las filas y columnas se empiezan a contar desde 0).

k representa el nuevo elemento que modificará lo que estaba en la fila i y columna j .

Revisemos otro caso

Imaginemos que recibimos una matriz M ya creada.

No obstante, a priori no sabemos cuántas filas o columnas tiene.

Entonces **¿Cómo lo podríamos resolver?**

Matriz

M

Listas de listas

En forma general, si tenemos una matriz M :

Para poder saber el número de filas:

```
len (M)
```

Para poder saber el número de columnas:

```
len (M[0] )
```

Veamos cómo resolver en Python

CÓDIGO	RESULTADO
<pre>matriz2 = [["a", "b", "c"], ["d", "e", "f"], ["g", "h", "i"], ["j", "k", "l"]] print("Filas:", len(matriz2)) print("Columnas:", len(matriz2[0]))</pre>	<pre>Filas: 4 Columnas: 3</pre>

Importante

Python no controla que las listas que nosotros agregamos a una lista de listas tengan el mismo tamaño. Por lo tanto, podría ocurrir que al crear una lista de listas no cree una matriz (es decir, que cada fila tenga el mismo número de columnas). Hay que tener mucho cuidado al crear listas, ya que nos podríamos encontrar con este problema.

Esto reflejaría errores en nuestro código. En general, queremos evitar listas de listas como esta:

```
lista = [[1],[2,3,4],[5,6,7,8],[9,10]]
```


>>> Cierre

Has finalizado la revisión de los contenidos que corresponden a esta clase.

A continuación, te invitamos a estudiar la siguiente clase del módulo.