

DECISIONES



Introducción a la Programación (11071)
Departamento de Ciencias Básicas
Universidad Nacional de Luján



MATERIAL RELACIONADO

TEORIA



Teórica 4 (A, B, y C) - Decisiones.



Apunte

Cap.

4

-

Decisiones.

PRÁCTICA



TP VI - Estructura alternativa y operadores lógicos.

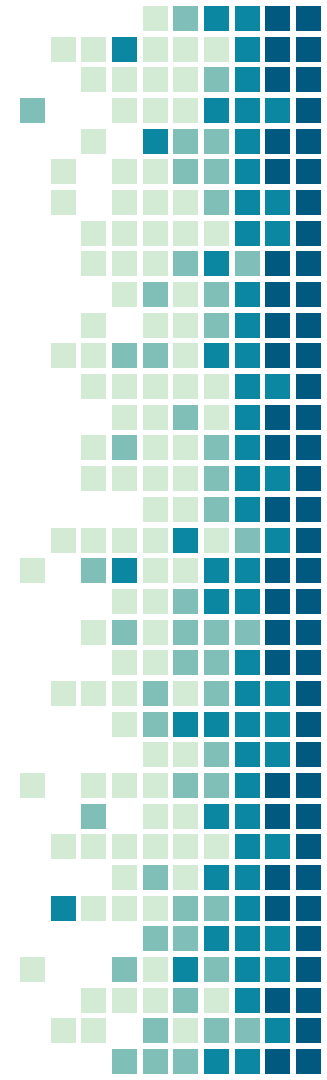


Notebook 3 - Condicionales.

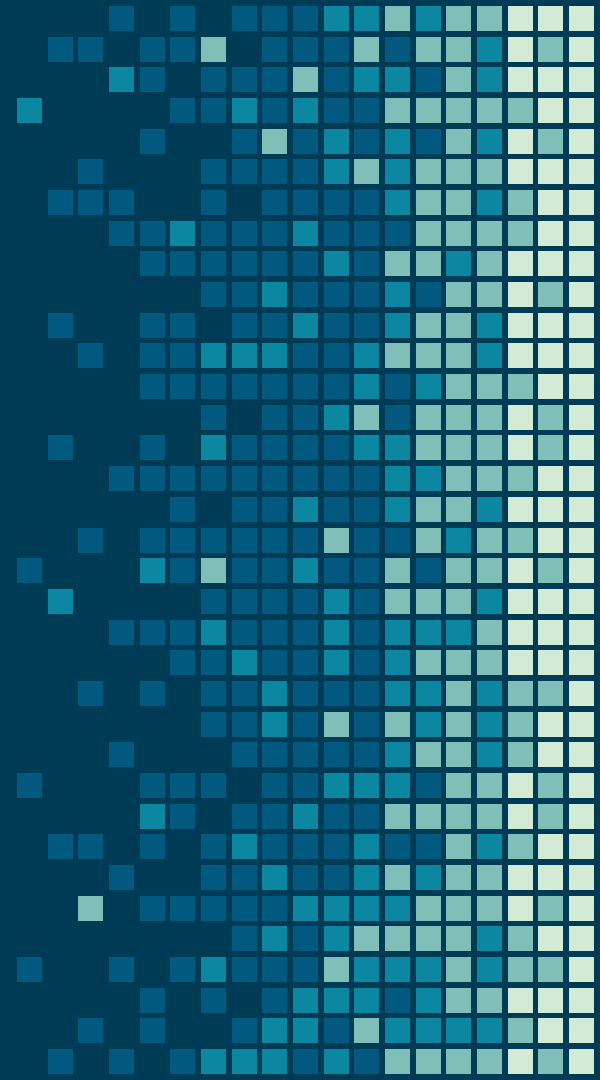
ENTREGABLES



HomeWork 2 - Condicionales.



REPASO DE CONCEPTOS



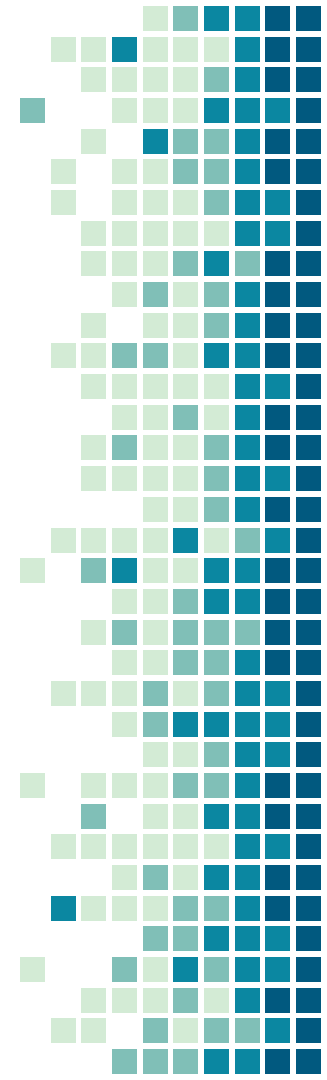
DECISIONES

DEFINICIÓN

En el paradigma procedural, los programas se ejecutan de manera secuencial, pero determinadas situaciones requieren que podamos tomar **decisiones** dentro del código.

ESTRUCTURAS/SENTENCIAS CONDICIONALES

- Los lenguajes de programación nos proveen con **estructuras o sentencias condicionales** para tomar decisiones dentro del código.
- Los condicionales se basan en la lógica proposicional (análisis de predicados con resultado *verdadero* o *falso*).



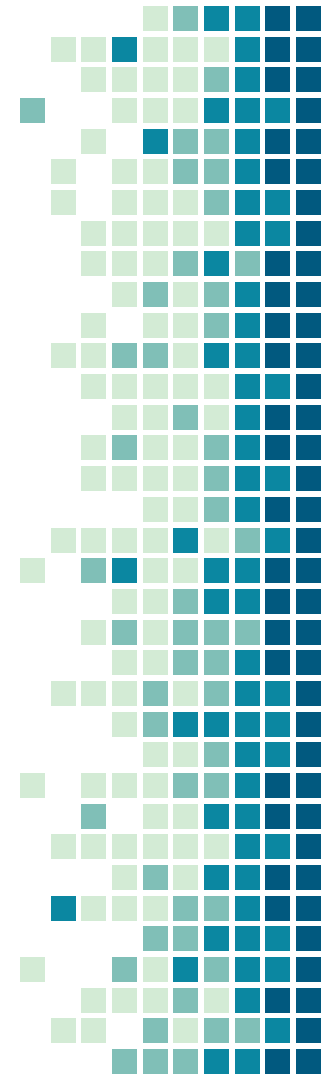
TIPO BOOLEANO

DEFINICIÓN

La mayoría de los lenguajes de programación soporta el tipo de dato **booleano**, el cual admite sólo dos valores: **verdadero** (*true*) o **falso** (*false*).

EN PYTHON

Se representa con el tipo **bool**, y los valores posibles son **True** o **False** (respetando las mayúsculas -*case sensitive*-).



OPERADORES RELACIONALES

DEFINICIÓN

Operaciones de análisis de condiciones (predicados) que tienen como resultado un valor booleano.

EN PYTHON

Existen varios operadores relacionales implementados (todos con el mismo orden de precedencia). Los más comunes son:

Sintaxis	Operación relacional
==	Igual que...
!=	Distinto de...
<	Menor estricto que...
>	Mayor estricto que...
<=	Menor o igual que...
>=	Mayor o igual que...

OPERADORES LÓGICOS

DEFINICIÓN

Operaciones de análisis de condiciones que tienen como resultado un valor booleano, pero que a diferencia de los operadores relacionales, **sólo funciona entre valores booleanos.**

LOS MÁS COMUNES

- **Conjunción** (**y** lógico, o **and**). Verdadero si y sólo si todos los operandos son *verdaderos*.
- **Disyunción** (**o** lógico, o **or**). Verdadero si al menos uno de los operandos es *verdadero*.
- **Negación** (**no** lógico, o **not**). Invierte el valor del operando.

EN PYTHON

CONJUNCIÓN

Operador **and**.

`condición1 and condición2`

DISYUNCIÓN

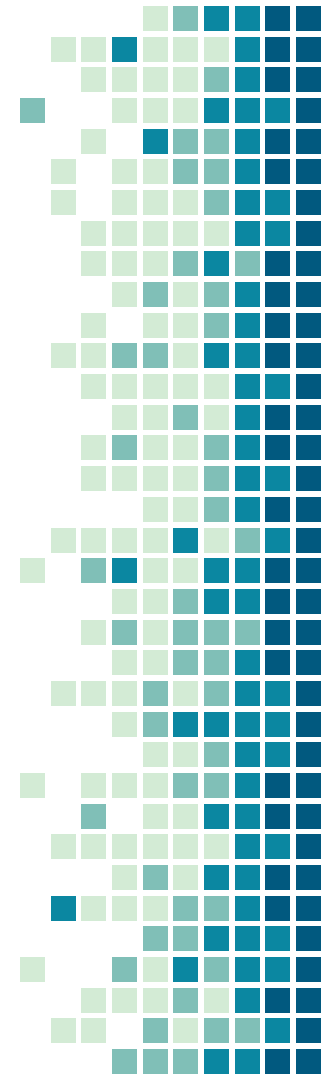
Operador **or**.

`condición1 or condición2`

Negación

Operador **not**.

`not condición`



EJERCICIOS GUIADOS

Para resolución de TP VI.



EJERCICIO MODELO (TP VI)

OBJETIVO DEL EJERCICIO

Comprender y ejercitar los condicionales y su aplicación en Python.

ENUNCIADO

Codifique un programa que solicite al usuario ingresar dos números enteros, y luego informe si la suma de ambos es mayor a 100 o no.

CONTINUACIÓN

Agregue una función que reciba dos enteros como parámetro, e informe (mostrando en pantalla) **si al menos uno** de los valores es mayor a 50.

CONTINUACIÓN (2)

Agregue una función que reciba dos enteros como parámetro, e informe (mostrando en pantalla) **si ambos** valores son menores que 100.

