

Descripció

Resoldràs alguns problemes de la vida quotidiana aplicant les estructures de dades i control en Python.

Un client de l'empresa en la qual treballes demana una llista de programes molt senzills, però que li facilitarien molts processos.

Se't demanarà que programis varies funcions. Per a fer l'entrega, hauràs d'entregar tant el codi de la funció:

```
def funcioexemple(variable1, variable2):  
  
    resultat = variable1 + variable2 # sumem les dues variables  
  
    return resultat
```

Com la seva execució per demostrar que funciona correctament, ensenyant-ne l'output:

```
funcioexemple(3, 8)
```

Intenta que tot el codi que generis sigui el més reproduïble possible.



Nivell 1

1. Calculadora de l'índex de massa corporal

- Escriu una funció que calculi l'IMC ingressat per l'usuari/ària, és a dir, qui ho executi haurà d'ingressar aquestes dades. Pots obtenir més informació del seu càlcul en: [Índice de masa corporal IMC que es y como se calcula](#).
- La funció ha de classificar el resultat en les seves respectives categories.

Consell: Intenta validar les dades prèviament, perquè envii un missatge d'avertència si les dades introduïdes per l'usuari/a no estan en el format adequat o no pren valors raonables.

2. Convertidor de temperatures

Existeixen diverses unitats de temperatura utilitzades en diferents contextos i regions. Les més comunes són Celsius (°C), Fahrenheit (°F) i Kelvin (K). També existeixen altres unitats com Rankine (°Ra) i Réaumur (°Re).

Selecciona almenys 2 conversors, de tal manera que en introduir una temperatura retorni, com a mínim, dues conversions, de manera que es puguin guardar (recorda que un print() no es pot guardar mai).

Consell: Intenta validar les dades prèviament, perquè envii un missatge d'avertència si les dades introduïdes per l'usuari/a no estan en el format adequat.

(EXTRA): Pensa una manera d'emmagatzemar totes les possibles conversions en un sol objecte (Llista? Diccionari? DataFrame?) en comptes d'escriure molts if else en funció de la temperatura d'origen i la temperatura de destí.

3. Comptador de paraules d'un text.

Escriu una funció que donat un text, mostri les vegades que apareix cada paraula. Intenta que es gestionin totes les casuístiques possibles que facin que el programa no funcioni correctament.

(EXTRA): Quina és la llargada mitjana de les paraules del text que has escrit? "Hola com va?" hauria de retornar $(4+3+2) / 3 = 3$

4. Diccionari invers (amb possibilitat de duplicats)

Resulta que el client té una enquesta molt antiga que s'emmagatzema en un diccionari i els resultats els necessita al revés, és a dir, intercanviant les claus i els valors. Els valors i claus en el diccionari original són únics; si aquest no és el cas, la funció hauria d'imprimir un missatge d'avertiment, juntament amb una llista amb els valors associats a la clau repetida.

Fixa't en el següents exemples per veure l'output esperat.

[5]:

```
reverse_dictionary({'a':1, 'b':2, 'c':3})
```

[5]:

```
{1: 'a', 2: 'b', 3: 'c'}
```

[7]:

```
reverse_dictionary({'x':'apple', 'y':'banana', 'z':'banana'})
```

[7]:

```
{'apple': 'x', 'banana': ['y', 'z']}
```



Nivell 2

1. Comptador i endreçador de paraules d'un text.

El client va quedar content amb el comptador de paraules, però ara vol llegir arxius TXT i que calculi la freqüència de cada paraula ordenades dins de les entrades habituals del diccionari segons la lletra amb la qual comencen, és a dir, les claus han d'anar de la A a la Z i dins de la A hem d'anar de la A la Z. Per exemple, per a l'arxiu "tu_me_quieres_blanca.txt" la sortida esperada seria:

```
{ 'a': { 'a': 3,
        'agua': 1,
        'al': 2,
        'alba': 4,
        'alcobas': 1,
        'alimenta': 1,
        'alma': 1,
        'amarga': 1,
        'azucena': 1},
  'b': { 'baco': 1,
        'banquete': 1,
        'bebe': 1,
        'blanca': 3,
        'boca': 1,
        'bosques': 1,
        'buen': 1},
  'c': { 'cabañas': 1,
        'carnes': 2,
        'casta': 3,
        'cerrada': 1,
        'con': 4,
        'conservas': 1,
        'copas': 1,
        'corola': 1,
        'corriste': 1,
        ...
        'tornadas': 1,
        'tú': 8},
  'u': { 'un': 1, 'una': 1},
  'v': { 'vestido': 1, 'vete': 1, 'vive': 1},
  'y': { 'y': 5}}
```

2. Conversió de tipus de dades.

El client rep una llista de dades i necessita generar dues llistes, la primera on estaran tots els elements que es van poder convertir en flotants i l'altra on estan els elements que no es van poder convertir. Exemple de la llista que rep el client:

```
[ '1.3', 'one', '1e10', 'seven', '3-1/2', ('2',1,1.4,'not-a-number'), [1,2,'3','3.4']]
```

```
conversion([ '1.3', 'one', '1e10', 'seven', '3-1/2', ('2',1,1.4,'not-a-number'), [1,2,'3','3.4']])
```

```
([1.3, 10000000000.0, 2.0, 1.0, 1.4, 1.0, 2.0, 3.0, 3.4],
 ['one', 'seven', '3-1/2', 'not-a-number'])
```



Nivell 3

1. Generador de Contrasenyes

Explora el funcionament del mòdul random de la llibreria numpy. [Random module in NumPy](#)

En aquest punt, el client ha detectat un problema amb les contrasenyes que utilitzen els seus treballadors. Asdf1234, dates d'aniversari o similars. Per a solucionar-ho ens ha encarregat una funció de Python que generi contrasenyes més segures. La funció ha de dependre dels següents paràmetres:

- longitud (int): Longitud de la contrasenya
- majuscles (bool = True): Si hi ha d'aparèixer majúscules
- minuscules (bool = True): Si hi ha d'aparèixer minuscules
- numeros (bool = True): Si hi ha d'aparèixer números
- signes (bool = False): Si hi ha d'aparèixer caràcters especials (,-\$? o similars)

Així doncs, si executem la funció de la següent manera:

```
crear_contrasenya(10, True, True, True, True)
```

Hauríem d'obtenir un output (que hem de poder guardar) de l'estil:

```
9Er,5Vn8P$
```

Assegura't que es compleixin tots els criteris, i que aquestes contrasenyes siguin realment aleatòries.

(EXTRA) Explora com podríem fer que la funció copiés la contrasenya automàticament al porta papers de l'ordinador (com si l'haguéssim seleccionat i fet ctrl+copy).

2. Processament de dades simple

Una companya de feina ens ha demanat un favor, aprofitant que sap que estem aprenent a programar. Té un històric de partits de futbol català en un fitxer, on hi ha emmagatzemat els noms dels equips i els resultats. Necessita que processem les dades de manera automàtica, per extreure'n els resultats que necessita.

Utilitza el fitxer "historic_partits.txt"

Necessita un programa que retorni:

- El nombre total de gols que ha fet cada equip.
- El nom de l'equip més golejador.
- El nom de l'equip més golejat
- La classificació global (cada victòria: 3 pts, empat 1 pts, derrota 0 pts)

Com a referència, els equips amb més punts serien els següents:

	partits guanyats	partits empatats	partits perduts	gols	gols encaixats	punts
Girona FC	31	3	13	139	94	96
Llagostera	29	7	20	159	142	94
Sabadell	26	7	15	141	121	85
Cornellà	25	7	22	147	146	82
RCD Espanyol	23	11	21	131	144	80
Figueres	23	10	23	161	154	79
Lleida Esportiu	23	6	23	129	133	75
Terrassa	20	14	23	147	164	74
FC Barcelona	22	7	15	125	115	73

Pots utilitzar el format que et sigui més convenient.



Objectius

- L'objectiu d'aquest sprint és consolidar els fonaments de sintaxi de Python i conceptes generals de la programació, com la creació de funcions reutilitzables, l'ús de diferents estructures de dades i el control de flux i automatització.



Durada:

10 dies.





Lliurament:

❗ Important

Emmagatzema en un repositori del teu GitHub una carpeta que contingui:

- Un arxiu .ipynb de Jupyter Notebook **executat que amb tots els enunciats**, les solucions i les interpretacions dels resultats.
- No incloguis print() d'informació que no sigui rellevant per a la resolució del problema.



[historic.partits.txt](#)

26 de maig 2025, 9:56 PM



[tu me quieres blanca.txt](#)

11 de juny 2025, 12:23 PM

Resum de la qualificació

Grups separats: Data Analytics GIII - Presencial

Ocultat per a l'alumnat	No
Participants	11
S'han tramès	0
Necessiten qualificació	0



 Documentació d'aquesta pàgina

 Contacteu amb el suport del lloc

Has iniciat sessió com a [Joan Gasull Jolis](#) ([Tanca la sessió](#))

[Política de Cookies](#) - [Política de Privacitat](#) - [Condicions Generals d'Us](#)

/ dark mode */ /**evitar que es colapsi el contingut de la descripció dels cursos***/ /**** canvi colors iconas***/*

Funciona amb [Moodle](#)

[Privacy Policy](#)