

AWS1. M3. UF2. Pràctica Mòduls Python:

Exercici 1:

Crearem un paquet anomenat **matemàtiques** i a dintre un mòdul anomenat **operacions_basiques.py** que contindrà les funcions:

- sumar -> Funció que suma els dos paràmetres i retorna el resultat.
- restar -> Funció que resta dels dos paràmetres i retorna el resultat.
- multiplicar -> Funció que multiplica els paràmetres i retorna el resultat.
- dividir -> Funció que divideix els paràmetres (numerador, denominador) i retorna el resultat. Llança una excepció en cas de divisió per zero. El codi d'aquesta excepció mostrarà el missatge 'ERROR: No es pot dividir per zero' i llançarà una excepció del tipus **ZeroDivisionError**.

NOTA: Cada funció rebrà dos paràmetres (els operands). Cada funció retornarà el resultat de l'operació. Controli l'excepció de la divisió per zero en la funció dividir.

Crea un mòdul anomenat prova.py que està guardat a fora de la carpeta **matemàtiques** i que accedeixi al paquet **matemàtiques** i utilitzi totes les funcions.

Exercici 2:

- a) Importeu el mòdul del exercici1 en el següent codi, de manera que les instruccions següents s'executin sense produir errors:

```
a, b = 13, 3

print ('operands =', a, b)

print ('sumar =', operacions_basiques.sumar (a, b))

print ('restar =', operacions_basiques.restar (a, b))

print ('multiplicar =', operacions_basiques.multiplicar (a, b))

print ('dividir =', operacions_basiques.dividir (a, b))
```

- b) Com hauries d'importar el mòdul perquè no hi hagués errors en executar?

```
a, b = 13, 3

print ('operands =', a, b)

print ('sumar =', ob.sumar (a, b))

print ('restar =', ob.restar (a, b))

print ('multiplicar =', ob.multiplicar (a, b))

print ('dividir =', ob.dividir (a, b))
```

- c) I com faries que funcionés el següent codi?

```
a, b = 13, 0

print ('operands =', a, b)

print ('sumar =', sumar (a, b))

print ('restar =', restar (a, b))

print ('multiplicar =', multiplicar (a, b))

print ('dividir =', dividir (a, b))
```

Exercici 3:

Crear un mòdul per validació de noms d'usuaris. L'esmentat mòdul, haurà de complir amb els següents criteris d'acceptació:

- El nom d'usuari ha de contenir un mínim de 6 i un màxim de 12.
- El nom d'usuari ha de ser alfanumèric.
- Nom d'usuari amb menys de 6 caràcters, retorna el missatge "El nom d'usuari ha de contenir almenys 6 caràcters".
- Nom d'usuari amb més de 12 caràcters, retorna el missatge "El nom d'usuari no pot contenir més de 12 caràcters".
- Nom d'usuari amb caràcters diferents als alfanumèrics, retorna el missatge "El nom d'usuari pot contenir només lletres i números".
- Nom d'usuari vàlid, retorna True.

Exercici 4:

Crear un mòdul per validació de contrasenyes. L'esmentat mòdul, haurà de complir amb els següents criteris d'acceptació:

- La contrasenya ha de tenir un mínim de 8 caràcters.
- Una contrasenya ha de tenir lletres minúscules, majúscules, números i almenys 1 caràcter no alfanumèric.
- La contrasenya no pot contenir espais en blanc.
- Contrasenya vàlida, retorna True.
- Contrasenya no vàlida, retorna el missatge "La contrasenya triada no és segura".

Exercici 5:

Crear un mòdul que sol·liciti a l'usuari l'ingrés d'un nom d'usuari i contrasenya i que els validi utilitzant els mòduls generats en els dos exercicis anteriors.

Ajuda: per comptar la quantitat de caràcters d'una cadena, en Python s'utilitza la funció incorporada: `len (cadena)`