Feladatok

# Kamatszámítás

1. Implementálj egy osztályt, amely az InterestCalculator interfészt valósítja meg!
   1. Az osztálynak legyen egy calculate\_interest metódusa, amely float típusú principal (tőke), rate (kamatláb) és time (időtartam) paramétereket vár. A metódusnak vissza kell adnia a kiszámított kamat összegét.
2. Implementálj egy osztályt, amely a DataValidator interfészt valósítja meg!
   1. Az osztálynak legyen egy validate\_data metódusa, amely float típusú principal, rate és time paramétereket vár. A metódusnak logikai értékkel kell visszatérnie, amely jelzi, hogy az adatok érvényesek-e vagy sem.

# Authenticator

1. Implementálj egy osztályt, amely az IIoHandler interfészt valósítja meg!
   1. Az osztálynak legyen egy read\_input metódusa, ami egy szöveget kér be a felhasználótól, és visszaadja a beolvasott értéket.
   2. Az osztálynak legyen egy write\_output metódusa, ami kiír egy szöveget a képernyőre.
2. Implementálj egy osztályt, amely az IDatabaseHandler interfészt valósítja meg!
   1. Az osztálynak legyen egy get\_user\_credentials metódusa, ami egy felhasználónevet vár paraméterként, és visszaadja a tárolt jelszót.
3. Implementálj egy osztályt, amely az IAuthenticator interfészt valósítja meg!
   1. Az osztálynak legyen egy authenticate metódusa, ami egy felhasználónevet és jelszót vár paraméterként, és bool értékkel tér vissza attól függően, hogy a megadott adatok helyesek-e.

* with statement
* for in file

# Online adathalászat

<https://www.napiarfolyam.hu/betapi>

1. Implementálj egy osztályt, amely az ExcelWriter interfészt valósítja meg!
   1. Az osztálynak legyen egy write\_data metódusa, ami egy listát kap paraméterként, és ezeket a listaelemeket soronként hozzáadja egy Excel táblázathoz.
2. Implementálj egy osztályt, amely az APICaller interfészt valósítja meg!
   1. Az osztálynak legyen egy make\_api\_call metódusa, ami egy URL-t kap paraméterként, és visszaadja az API válasz XML tartalmát szöveges formában.
3. Implementálj egy osztályt, amely az IOHandler interfészt valósítja meg!
   1. Az osztálynak legyen egy read\_input metódusa, ami egy szöveget vár paraméterként, és visszaadja a felhasználó által megadott értéket.
   2. Az osztálynak legyen egy write\_output metódusa, ami egy szöveget kap paraméterként, és kiírja azt a képernyőre.

* requests
* xml.etree.ElementsTree
* openpyxl
* response.status\_code
* element.tag
* child.text for child in element
* selected\_data = [] és selected\_data.append(data\_row)

Kiegészítő lehetőségek:

* Kéne tudni, milyen formátumban lesz az excel a végén
  + megadott formátum - xml tag-ek formalizálása
* ListFromXML data nem szerepel a specifikációban
* corrupted XML, üres lista, corrupted lista hogy legyen kezelve
* bemenetek:
  + API url
  + paraméterek
* hiányzó automata teszteknek az írása