

## Projektfeladat vizsgatevékenység

### B) Asztali- és webes szoftverfejlesztés, adatbázis-kezelés vizsgarész

Vizsgázó neve:

Szakma megnevezése:

Szoftverfejlesztő és -tesztelő

Szakma száma:

5 0613 12 03

Képző intézmény neve:

Budapesti Műszaki Szakképzési Centrum  
Verebély László Technikum

Vizsgaszervező neve:

Budapesti Műszaki Szakképzési Centrum  
Verebély László Technikum

A vizsgatevékenység időpontja:

2024.05.07. 08:00

A vizsgatevékenység helyszíne:

1139 Budapest Üteg u. 15.

A vizsgatevékenység időtartama:

240 perc

A vizsgafeladat értékelési aránya:

65 pont/ 120 pont

Értékelési tábla:	
0% - 39%	elégtelen (1)
40% - 49%	elégséges (2)
50% - 59%	közepes (3)
60% - 79%	jó (4)
80% - 100%	jeles (5)

A gyakorlati vizsgafeladatot jóváhagyom.

Budapest, 2024. május 7.

.....  
a vizsgabizottság elnöke

## Konzolos és grafikus részt egyaránt tartalmazó asztali alkalmazás fejlesztése

A következő két feladatsorban a Paralimpia nemzetközi sportverseny-sorozat nyári játékein elért magyar dobogós feldolgozó alkalmazásokat kell készítenie. A források az összes napjainkig megrendezett nyári Paralimpia versenyeseemény adatait tartalmazzák, azokat is, amelyeken Magyarország nem képviseltette magát. Ezeknél a rekordoknál minden éremeredmény helyén numerikus nulla érték szerepel.

### Konzolos asztali alkalmazás fejlesztése C# nyelven (15 pont)

A **paralympics.txt** UTF-8-as karakterkódolású, pontosvesszőkkel tagolt állomány, első sora a fejlécelemeket tartalmazza. Minden további sora azonos szerkezetű. Egy sora például:

**1996;Egyesült Államok;Atlanta;5;2;3**

ahol az adattagok jelentése a következő:

1996	az év, amikor a sporteseményt rendezték
Egyesült Államok	az adott évben a rendező ország neve
Atlanta	az esemény helyszínéül szolgáló város neve
5, 2, 3	rendre a magyar sportolók által szerzett arany-, ezüst- és bronzérmek száma

- Hozzon létre egy objektumosztályt, ami képes reprezentálni egy versenyeseeményt. Az osztály egyetlen publikus konstruktora az adatfile egy sorát kapja paraméterül!
  - Bírálja felül az osztály **ToString()** metódusát olyan módon, hogy hívásnál a példány minden tulajdonság értékét szövege alakítva adja vissza a, az 5-ös feladat mintájához hasonló módon!
  - Olvassa be az állomány tartalmát egy objektumpéldányokat tartalmazó kollekcióba, ennek felhasználásával oldja meg a következő feladatokat. A feladatok kiírása a mintához hasonló módon történjen, minden feladat eredménye előtt jelenítse meg a feladat sorszámát.
- Jelenítse meg a konzolablakban, hogy hány alkalommal tartottak eddig paralimpiai versenyeket!
  - Határozza meg, és írja ki a konzolra, hogy a magyar sportolók eddig összesen hány érmet szereztek a paralimpián!
  - Jelenítse meg a konzolon, hogy Magyarország a versenysorozat mely helyszínein és években nem képviseltette magát a Paralimpián. Figyelembe veheti, hogy minden alkalommal, amikor voltak magyar nemzetiségű sportolók az eseményen, akkor legalább egy dobogós eredményt szereztek.
  - Jelenítse meg a konzolon azon országok listáját, akik már több alkalommal is rendezői voltak a paralimpiai játékoknak. Az országok mellett az az alkalmak száma is jelenjen meg.
  - Kérjen be a felhasználótól egy évszámot. Keresés előtt ellenőrizze, hogy megfelelő formátumú-e a beírt adat. Ha az adott évben rendeztek paralimpiai játékokat, jelenítse meg a találatnak megfelelő objektum tulajdonságait az osztályban felülbíralt **ToString()** metódus hívásával. Ha a listában nem szerepel olyan elem, amihez a bekért évszám tartozik, akkor írja ki, a konzolra, hogy **„Ebben az évben nem rendeztek paralimpiai játékokat.”**!

```
1. feladat: A Paralimpia eddig 16 alkalommal került megrendezésre
2. feladat: A magyar sportolók összesen 161 érmet szereztek
3. feladat: A következő helyszíneken nem képviseltette magát Magyarország:
    Róma (1960)
    Tokió (1964)
    Tel Aviv (1968)
    Arnheim (1980)
4. feladat: Országok, amik több paralimpiát is rendeztek:
    Japán (2 alkalom)
    Egyesült Államok (2 alkalom)
5. feladat: Írjon be egy évszámot: 1992
    Év: 1992
    Ország: Spanyolország
    Város: Barcelona
    Eredmények: arany: 4 | ezüst: 3 | bronz: 4
```

## Grafikus asztali alkalmazás fejlesztése C# nyelven (10 pont)

- Hozzon létre a helyi adatbázisszerveren **paralimpia** néven **utf8\_hungarian\_ci** karakterkészletű adatbázist, majd importálja **nyari\_magyar\_ermetablazat.sql** file tartalmát. A file a következő szerkezetű adattáblát tartalmazza:

nyari\_magyar\_ermetablazat

id	INT AI	PK
orszag	VARCHAR(20)	
varos	VARCHAR(20)	
ev	INT	
arany	INT	NULL
ezust	INT	NULL
bronz	INT	NULL

**id:** a versny sorszáma és egyben azonosítója, automatikus, ez a tábla elsődleges kulcsa

**orszag, varos:** a rendező ország, és a sporteseménynek helyszínt adó város neve

**ev:** az év, amikor az adott sporteseményt rendezték, egyedi érték a táblában

**arany, ezust, bronz:** a magyar sportolók által szerzett érmek száma

- Készítse el a grafikus alkalmazás statikus részeit a minta alapján!

Magyar paralimpiai eredmények

szűrés ország vagy város alapján:

#	rendező ország	helyszín	év	arany	ezüst	bronz
---	----------------	----------	----	-------	-------	-------

kijelölt év: ####

arany: 0    ezüst: 0    bronz: 0

érmek számának módosítása

- Telepítse a szükséges csomagokat, kapcsolódjon az adatbázishoz. Az alkalmazás betöltésekor jelenjen meg az adattábla összes rekordja az alkalmazásablak adatrácsában!
- Ha a mintán fent látható szöveges beviteli mezőbe gépelünk, az adatrácsban csak azok a rekordok jelenjenek meg, ahol az **ország** vagy **varos** attribútumok valamelyike a szöveges beviteli mező tartalmával kezdődik!

Magyar paralimpiai eredmények

szűrés ország vagy város alapján: Egyel

#	rendező ország	helyszín	év	arany	ezüst	bronz
7	Egyesült Államok	New York	1984	13	11	5
10	Egyesült Államok	Atlanta	1996	5	2	3
14	Egyesült Királyság	London	2012	2	6	6

kijelölt év: 1996

arany: 5 ezüst: 2 bronz: 3

érmek számának módosítása

- Ha egérekattintással kijelöljük az adatrácsban valamelyik sort, a hozzá tartozó rekord **arany**, **ezüst** és **bronz** mezőinek értéke jelenjen meg az alkalmazásablak alján lévő megfelelő numerikus beviteli mezőkben!
- Ha van kijelölve sor, az alkalmazásablak jobb alsó részén lévő parancsgomb legyen aktív. A gomb megnyomását követően jelenjen meg egy üzenetablak, amiben a felhasználó meg tudja erősíteni a módosítás szándékát. Pozitív megerősítés után módosítsa az adatbázis megfelelő rekordját a numerikus beviteli mezők tartalma szerint, majd frissítse az adatrács tartalmát!

Magyar paralimpiai eredmények

szűrés ország vagy város alapján: Ne

#	rendező ország	helyszín	év	arany	ezüst	bronz
4	Németország	Heidelberg	1972	0	0	1
7	Egyesült Államok	New York	1984	13	11	5

MEGERŐSÍTÉS

! Biztosan módosítani kívánja a(z) 1984-as/es paralimpia érmeszámait?

Igen Nem

kijelölt év: 1984

arany: 33 ezüst: 33 bronz: 33

érmek számának módosítása

## Komplex webes és adatbázis-kezelési feladat

40 pont

### Kémiai elemek

Már általános iskolában találkozhattunk a kémiai elemekkel, és azok rendszerezésével.

Feladata egy komplex webalkalmazás megvalósítása, amelynek segítségével a kémiai elemekről kaphatunk információkat. Az alkalmazásnak két jól elkülönülő része REST API segítségével cserél információkat. Az API végpont egy adatbázisból nyeri ki és teszi elérhetővé a lekérhető adatokat, az API kliens a végponttal kommunikál, és a felhasználó egy responszívan működő weboldalon tájékozódhat.

A feladatot a leírások és a minták alapján kell megoldania, az ehhez szükséges forrásokat a vizsga előtt a megadott helyen találja. **Figyelmesen olvassa el a leírást!**

A feladat megoldása során a következő internetes oldalakat használhatja:

- <https://www.w3schools.com/>
- <https://www.php.net/>
- <https://www.javascript.com/>
- <https://getbootstrap.com/>
- <https://ajax.googleapis.com/>
- <https://jquery.com/>
- <https://developer.mozilla.org/>
- <https://info.nytt.hu/>

\* \* \*

### API végpont készítése

#### 1) Készítse elő az adatbázist!

Az adatbázis előkészítéséhez szüksége van a források között megtalálható **elemek.sql** állományra, amely tartalmazza megfelelő adatstruktúrát és a kémiai elemekre vonatkozó információkat.

#### Az elemek tábla mezői

elemek	
rendszám	: int(3)
fem	: tinyint(1)
halmaz	: char(1)
csoport	: varchar(20)
ev	: int(4)
nev	: varchar(20)
vegyjel	: varchar(2)
felfedező	: varchar(100)

rendszám	: egyedi azonosító, az elem rendszáma (egész)
fem	: az elem jellemzője; 0-nemfém, 1-fém, null-egyik sem (logikai)
halmaz	: az elem halmazállapota; S-szilárd, F-folyékony, G-gáz, I-ismeretlen (karakter)
csoport	: az elem besorolása (szöveg)
nev	: az elem megnevezése (szöveg)
ev	: az elem felfedezésének éve (egész)
vegyjel	: az elem vegyjele (szöveg)
felfedező	: az elem felfedezője (szöveg)

- **Csatlakozzon** a lokális adatbázis-kiszolgálójához a phpMyAdmin felületen keresztül!

- **Hozzon létre** új adatbázist „kemia” néven!
- **Importálja** az elemek.sql állományt a létrehozott adatbázisba!
- **Tanulmányozza** az adatokat!
- **Módosítsa** az adatbázist!
  - Az ismeretlen halmazállapotú elem karakter jelölése nem megfelelően szerepel. Módosítsa minden olyan helyen „I”-re a halmazállapotot, ahol „N” szerepel!
  - A 119-es rendszámmal ellátott elem jelenleg még nem bizonyítottan létező elem. Törölje az erre vonatkozó rekordot az adatbázisból!
- Exportálja a „kemia” adatbázist az API végpont mappájába **vizsga.sql** néven!

2) Hozzon létre a lokális szerveren egy API végpontot, amely a következő címen legyen elérhető:

`http://localhost/kemia/api`

3) A végpontot úgy alakítsa ki, hogy az paraméterezhető legyen az elemek szűrésére az alábbi módokon:

```
http://localhost/kemia/api/?elem=mind
http://localhost/kemia/api/?elem=hélium
http://localhost/kemia/api/?elem=oxigén
...
```

A hívások esetében az API válaszként JSON formátumban adja vissza a választott elem (vagy minden elem) összes adatát az alábbi módon.

```
▼ 0:
  rendszam: "1"
  fem:      "0"
  halmaz:   "G"
  csoport:  "egyéb nemfém"
  ev:       "1766"
  nev:      "Hidrogén"
  vegyjel:  "H"
  felfedezo: "Henry Cavendish"

▼ 1:
  rendszam: "2"

.....
  felfedezo: "- "

▼ 117:
  rendszam: "118"
  fem:      "0"
  halmaz:   "I"
  csoport:  "nemesgáz"
  ev:       "2006"
  nev:      "Oganeszon"
  vegyjel:  "Og"
  felfedezo: "Egyesített Atomkutató Intézet, Dubna és L
```

- 4) Bővítse az API paraméterezési lehetőségét egy halmazállapot szerinti keresési funkcióval az alábbiak szerint:

`http://localhost/kemia/api/?halmaz=gáz`

A fenti hívás esetén az API a megadott halmazállapotú elemek összes adatát visszaadja **JSON** formátumban, – a 3. pontban megadott minta szerint. Ügyeljen arra, hogy az adatbázisban a halmazállapot egyetlen karakterrel van megadva!

- 5) Az API paraméter hiányában vagy nem megfelelő paraméterezés esetén adjon az alábbiak szerinti **JSON** formátumú választ, amelyben mintákat mutat arra vonatkozóan, hogy milyen paraméterezéssel lehet használni az API-t:

```
minta1:    "?elem=mind"
minta2:    "?elem=Hélium"
minta3:    "?halmaz=gáz"
```

- 6) Az API találat hiányában szintén **JSON** formátumban adjon hibaüzenetet az alábbiak szerint:

```
üzenet:    "nincs találat"
```

- 7) Az API válaszaihoz állítsa be a megfelelő fejléct és response kódokat!

Az 5. pontban leírtak esetén **404**, a többi esetben **200** legyen!

\* \* \*

### ***API kliens megvalósítása reszponzív webfelületen***

Olyan reszponzív weboldalt kell készítenie, amely igazodik a különböző eszközökhöz, stílusát tekintve a projekt egészére nézve egységes és áttekinthető, valamint a mintának és leírásoknak megfelel. A weboldal a következő címen legyen elérhető:

`http://localhost/kemia`

A weboldal tartalmaz statikus és dinamikus elemeket. A dinamikus tartalom az API-tól kapott adatok alapján jelenik meg. Amennyiben az előző feladatban az API végpontot nem tudta teljesen, vagy részben tudta elkészíteni, úgy ebben a feladatban a saját változata helyett a következő API-t használhatja:

`https://info.nytt.hu/web/api/kemia/`

- 1) Alakítsa ki a megfelelő fájlstruktúrát, amely tartalmazza a weboldalhoz szükséges forrásokat és saját fájlokat! A szükséges forrásokat a **[forrasok]** mappában, a felhasználható sablon elemeket a **[sablonok]** mappában találja.
- 2) Alakítsa ki az oldalak egységes szerkezetét a leírás és a minták alapján! Ügyeljen a magyar nyelvi és karakterkódolási beállításokra! Adjon a weboldalnak témájához kapcsolódó címet! Favikonként használja az **atomszerkezet.png** fájlt!
- 3) Amennyiben külső scriptet használ, helyezze el a szükséges fájlokra mutató hivatkozásokat!
- 4) Használjon saját készítésű belső és/vagy külső **CSS** stíluslapo(ka)t és/vagy standard stíluslapo(ka)t (**bootstrap, w3**) és/vagy **JavaScript** kódokat az ízléses megjelenés kialakítására! A szövegek tördelése és a képek igazítása a minta szerint történjen! Stílusban nem kell minden tekintetben a minta szerinti megjelenést megvalósítani, de ügyeljen arra, hogy a weboldal minden egyes eleme egyértelmű és átlátható legyen, valamint megfeleljen a leírásnak és reszponzívan viselkedjen! A mintában található színek: fehér (*White*), világosszürke (*LightGray*), világoskék (*#00c9b9*), lila (*#2a0054*), rózsaszín (*#ff5c5d*).
- 5) Hozza létre a statikus tartalmakat!
  - Fejléc
    - Kép: **atomszerkezet.png**
    - Cím: „*Kémiai elemek*” – egyes szintű címsor
  - Menüpontok
    - Hivatkozás: „*Kezdőlap*” – kezdő oldalra mutat
    - Hivatkozás: „*Elemek*” – kémiai elemeket felsoroló oldalra mutat
    - Hivatkozás: „*Mengyelejev-táblázat*” – külön ablakban megjeleníti a **ptable.jpg** képetA menükezelést paraméterezéssel oldja meg!  
Mobil nézetben a menü hamburger menüként jelenjen meg!
  - Tartalmi elemek: a cím minden oldalon kettes szintű címsor
    - Kezdő oldal
      - Cím: „*Kémiai elemek periódusos rendszere*”
      - Szöveg: **szoveg.txt**
      - Kép: **period.jpg**
    - Kémiai elemeket felsoroló oldal
      - Legördülő menü: szilárd, folyékony, gáz, ismeretlen, összes
      - Cím: „*Kémiai elemek listája*”
      - Táblázat fejléce (mobil nézetben csak az első három látszik)
        - „*Rendszám*”
        - „*Név*”
        - „*Vegyjel*”
        - „*Felfedezve*”
        - „*Felfedező*”



- Lábléc:
  - Vizsgáló teljes neve és a vizsga dátuma
  - Források megjelölései: „Periódusos rendszer”, „Elemek”, „Felfedezők”.  
A hivatkozások címei a **szoveg.txt** állományban találhatók.  
A hivatkozások külön ablakban nyílnak meg!
  - Hivatkozás: „fel” / nyíl / stb. – oldal tetejére mutat.  
A hivatkozás alapértelmezetten ne látszódjon, csak mobil nézetben!

6) Hozza létre az API-val való összeköttetést és végezze el a programozási feladatokat!

- A feladatok megoldásához használja tanult programozási nyelve(ke)t!
- Kérje le az adatokat a megfelelő paraméterezésekkel az API végponttól!
- Dolgozza fel az API-tól kapott választ, és állítsa össze a megjeleníteni kívánt tartalmat!

7) Jelenítse meg a dinamikus tartalmakat!

- Kémiai elemeket felsoroló oldal
  - Jelenjen meg az aktuálisan kiválasztott halmazállapotról vonatkozóan, hogy hány érintett elem van! Az oldal első betöltésekor az összes elem jelenjen meg!
  - Listázza ki táblázatos formában a kémiai elemekre vonatkozóan öt adatot (rendszer, név, vegyjel, felfedezve, felfedező)!  
Az adatok alapértelmezésben rendszer szerint növekvő sorrendben jelenjenek meg!  
Ügyeljen arra, hogy vannak olyan elemek, ahol az adatbázisban a felfedezés évszáma 0.  
Ezekben az esetekben az „Ókor” szöveg jelenjen meg!  
A halmazállapotok megjelenítésénél ügyeljen arra, hogy az adatbázisban egy-egy karakter jelöli a különböző halmazállapotokat, itt szövegesen jelenjenek meg!
  - Minden kémiai elem neve egy hivatkozás legyen, amely az adott elemhez tartozó részletes adatokat mutatja meg. A hivatkozást a rendszer vagy név szerint paraméterezze, és ez alapján azonosítsa be az adott elemet!
- Egy elemet bemutató oldal
  - Az oldalt semmilyen menüpontból nem lehet elérni. A felhasználó úgy juthat el ide, hogy az elemeket felsoroló oldalon az elem nevére kattint.
  - Az oldal tartalma:
    - Keresőmező: A felhasználónak legyen lehetősége elem nevére keresni!  
Ha nem létezik a keresett elem, akkor jelenítse meg a felhasználónak, hogy a keresése eredménytelen (pl. „Oxigén” elem nem található)
    - Cím: A kiválasztott vagy keresett elem neve
    - A kiválasztott vagy keresett elem összes adatának megjelenítése a mintához hasonlóan

