

**ÉRETTSÉGI VIZSGA • 2024. május 15.**

**INFORMATIKAI  
ISMERETEK**

**KÖZÉPSZINTŰ  
GYAKORLATI VIZSGA**

**2024. május 15. 9:00**

Időtartam: 180 perc

Pótlapok száma	
Tisztázati	
Piszkozati	

Jelölje be az Ön által választott  
programozási nyelvet!  
(Csak egy nyelvet jelölhet meg!)

Java ☐

C# ☐

**OKTATÁSI HIVATAL**

## Fontos tudnivalók

A vizsgán használható eszközök: a vizsgázó számára kijelölt számítógép, papír, toll, ceruza, lepecsételt pótlap.

A feladatlap belső oldalain és a pótlapon készíthet jegyzeteket, ezeket a vizsga végén be kell adni, de tartalmukat nem fogják értékelni.

A feladatokat tetszőleges sorrendben oldhatja meg.

Javasoljuk, hogy a feladatokat először olvassa végig, utána egyenként oldja meg az egyes részfeladatokat!

A forrásfájlokat a vizsgakönyvtárban találja.

Felhívjuk a figyelmet a gyakori mentésre, és feltétlenül javasoljuk a mentést minden esetben, mielőtt egy másik feladat megoldásába kezd.

Vizsgadolgozatát a vizsgakönyvtárába kell mentenie. A vizsga végén ellenőrizze, hogy minden megoldás a megadott könyvtárban van-e, mert csak ezek értékelésére van lehetőség! Ellenőrizze, hogy a beadandó állományok olvashatók-e, mert a nem megnyitható állományok értékelése nem lehetséges!

A programozási feladatnál a program csak abban az esetben értékelhető, ha a vizsgázó létrehozta a választott programozási környezetnek megfelelő forrásállomány(oka)t a vizsgakönyvtárában, és az tartalmazza a részfeladatok megoldásához tartozó forráskódot.

Az adatbázis-fejlesztés feladatnál az egyes részfeladatok megoldását adó SQL kódokat kell elmentenie. A feladatban megadott állományba mentett SQL kódok kerülnek csak értékelésre.

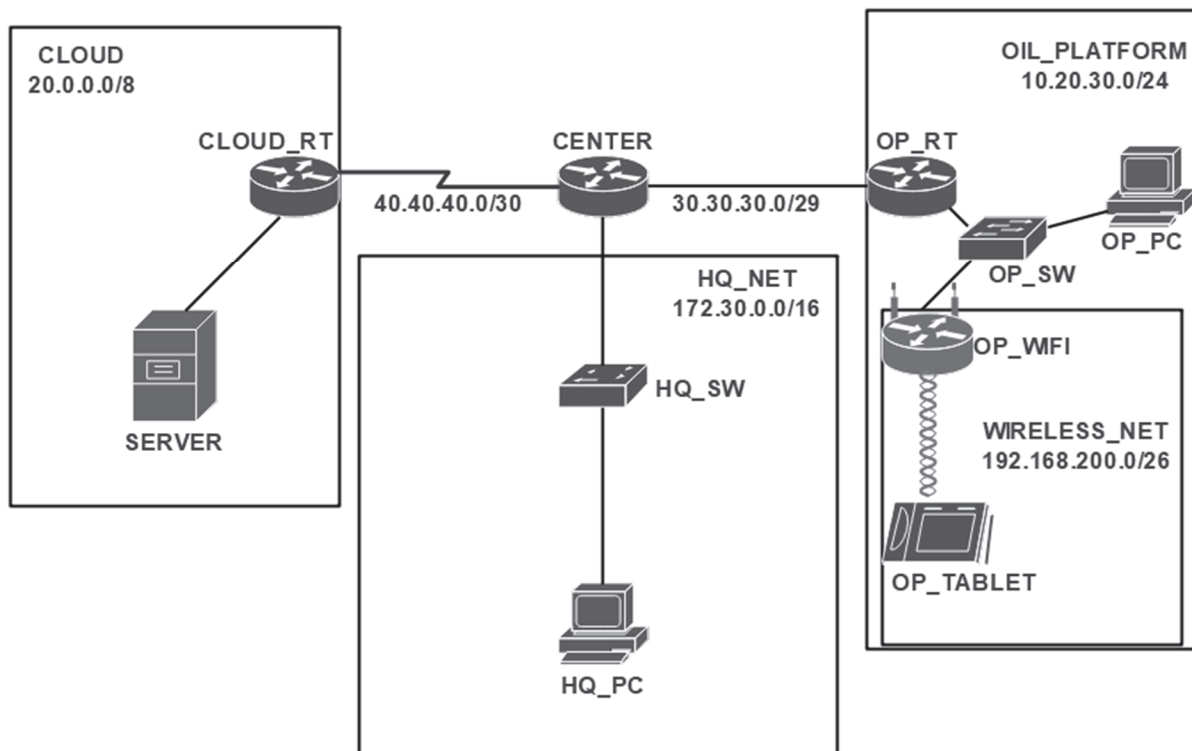
Amennyiben számítógépével műszaki probléma van, jelezze a felügyelő tanárnak! A jelzés ténye és a megállapított hiba jegyzőkönyvezésre kerül. A kiesett idővel a vizsga ideje hosszabb lesz. Amennyiben a hiba mégsem számítógépes eredetű, a javító tanár értékeléskor köteles figyelembe venni a jegyzőkönyv eseteírását. (A rendszergazda nem segítheti a vizsgázót a dolgozat elkészítésében.)

## 1. feladat

40 pont

## Olajplatform

Az **olajplatform** egy fix vagy úszó tengeri konstrukció, amelyet olajmezők üzemeltetésére használnak, és elsősorban a fúrási és olajkitermelési fázishoz szükséges eszközöket tartalmazza. Ezeken felül néha olyan berendezéseket is támogat, amelyek célja az emberi jelenlét biztosítása a fedélzeten. Egy ismert olajtársaság egy nemrég felfedezett olajlelőhelyének kiaknázására egy ilyen létesítmény kiépítését és üzemeltetését tűzte ki célul. Maga a létesítmény (OIL\_PLATFORM) magában foglal egy vezeték nélküli hálózatot (WIRELESS\_NET), és a szárazfölddel (CENTER) is összeköttetésben van. A vállalat központja (HQ\_NET) a szárazföldön található, ahol a termelési adatok feldolgozása történik. A feldolgozott adatok egy ismert felhőszolgáltatónál bérelt virtuális hálózatban (CLOUD) kerülnek eltárolásra. Az Ön feladata, hogy az alábbi követelményeknek megfelelően bekonfigurálja a topológián található eszközöket. Munkáját Olajplatform néven mentse a használt szimulációs program alapértelmezett formátumában!



*A feladat a következő oldalon folytatódik*

**Hálózati címzés:**

Eszköz	IP-cím	Alhálózati maszk	Alapértelmezett átjáró	Név
CLOUD_RT	20.255.255.254	255.0.0.0	-	CLOUD_RT
	40.40.40.1	255.255.255.252	-	
CENTER	40.40.40.2	255.255.255.252	-	CENTER
	172.30.255.254	255.255.0.0	-	
	30.30.30.6	255.255.255.248	-	
OP_RT	30.30.30.1	255.255.255.248	-	OP_RT
	10.20.30.254	255.255.255.0	-	
HQ_SW	172.30.255.253	255.255.0.0	172.30.255.254	HQ_SW
OP_WIFI	192.168.200.62	255.255.255.192	-	-
	10.20.30.1	255.255.255.0	10.20.30.254	-
SERVER	20.0.0.200	255.0.0.0	20.255.255.254	-
OP_PC	10.20.30.100	255.255.255.0	10.20.30.254	-
HQ_PC	DHCP kliens		172.30.255.254	-
OP_TABLET	DHCP kliens		192.168.200.62	-

**Beállítások:**

- Építse fel a hálózat prototípusát a szimulációs program segítségével! Az eszközök kiválasztásánál vegye figyelembe az alábbiakat:
  - A forgalomirányítók rendelkezzenek legalább két 100 Mbit/s (vagy nagyobb) sebességű interfésszel, valamint, ahol szükséges, legalább egy-egy, soros kapcsolat megvalósításához szükséges interfésszel.
  - A kapcsolók legalább 8 portosak legyenek.
  - A vezeték nélküli hálózathoz használja a szimulációs programban elérhető vezeték nélküli SOHO forgalomirányítók valamelyikét!
  - A vezeték nélküli kliens szimulálására bármely WIFI-képes eszköz megfelelő.
- Az eszközök elhelyezése után létesítsen kapcsolatot az eszközök között! A kapcsolatok kiépítéséhez használja az előző oldalon található ábrát! Az OP\_WIFI vezeték nélküli forgalomirányító az internet porton keresztül csatlakozzon az OP\_SW kapcsoló tetszőleges ethernet portjához!
- A fenti táblázat alapján ossza ki a megadott eszközöknek a megfelelő IP-paramétereket! A forgalomirányítókon (CLOUD\_RT, CENTER, OP\_RT), valamint a kapcsolón (HQ\_SW) állítsa be a táblázatban található nevet! DNS-szervernek a 20.0.0.200 IP-cím legyen beállítva az OP\_PC kliensen, és az OP\_WIFI vezeték nélküli forgalomirányító internet portján is!

4. Az OP\_RT forgalomirányítót készítse fel a távoli eléréshez! Ennek érdekében hozzon létre egy **kezelő** nevű felhasználót az eszközön, és állítsa be hozzá a **kezelő2024** jelszót. Ellenőrizze, hogy az OP\_RT eszközön hány virtuális terminál érhető el maximálisan, majd ezután tegye meg a megfelelő beállítást ahhoz, hogy az eszközt (OP\_RT) a maximális számban rendelkezésre álló terminálok mindegyikén a **kezelő** felhasználóval lehessen elérni.
5. Óvja az OP\_RT forgalomirányítót az illetéktelen felhasználóktól! Ennek érdekében állítsa be a **kezelő2024** jelszót a rendszergazdai mód eléréséhez!
6. A platformon (OIL\_PLATFORM) a kitermelt olaj mennyiségét és minőségét folyamatosan ellenőrzik, és ezeket az adatokat mind a központba (HQ\_NET), mind a felhőbe (CLOUD) el kell juttatni. Ennek érdekében definiáljon OSPF irányító protokollt az alábbiak szerint:
  - Irányító protokoll konfigurálása csak a CLOUD\_RT és a CENTER forgalomirányítókra történjen.
  - Mindkét forgalomirányítón a közvetlenül kapcsolódó hálózatok (az összes) kerüljenek hirdetésre.
  - A konfiguráció során mindegyik eszközön használja a 100-as folyamatazonosítót és a 10-es területazonosítót.
7. Az OP\_RT forgalomirányítón definiáljon alapértelmezett útvonalat, amely az ismeretlen forgalmat a CENTER forgalomirányítónak továbbítja.
8. A HQ\_NET hálózatába a CENTER forgalomirányító nyújt DHCP-szolgáltatást. Állítsa be a szolgáltatást az alábbiak szerint:
  - A HQ\_NET hálózat kiosztható címtartományának utolsó 20 címét zárja ki a dinamikus címkiosztásból!
  - Definiálja a DHCP-szolgáltatáshoz a hálózatot és az alapértelmezett átjárót!
  - DNS-szervernek a 20.0.0.200 cím kerüljön beállításra.
9. Tesztelje a DHCP-szolgáltatást! Ellenőrizze, hogy a HQ\_PC kliens megkapta-e a megfelelő IP-paramétereket!

*A feladat a következő oldalon folytatódik.*

10. A platformon (OIL\_PLATFORM) vezeték nélküli hálózat (WIRELESS\_NET) is működik, a rugalmas kommunikációt elősegítendő. Ehhez konfigurálja be a szükséges eszközöket az alábbiaknak megfelelően:

- A belső hálózat a 192.168.200.0/26 címtartományt használja. Állítsa be a belső hálózat alapértelmezett átjáróját ezen tartomány utolsó kiosztható címére!
- A belső hálózat kliense számára (OP\_TABLET) DHCP-szolgáltatást kell beállítania úgy, hogy a kiosztás a címtartomány első kiosztható címétől induljon, és maximum 60 db kliens egyidejűleg való kiszolgálására korlátozódjon. DNS-szervernek a 20.0.0.200 IP-cím legyen megadva.
- A vezeték nélküli hálózat azonosítója (SSID) **OP\_WIFI** legyen.
- A vezeték nélküli hálózaton továbbított adatok nem publikusak, így gondoskodni kell azok védelméről az esetleges illegális lehallgatások ellen. Ennek érdekében állítson be a vezeték nélküli hálózatra titkosított adatátvitelt hitelesítéssel:
  - A vezeték nélküli protokollnak a **WPA2** legyen kiválasztva.
  - A hitelesítéshez használt szöveg legyen: **oilpt123\***

11. Tesztelje a vezeték nélküli forgalomirányító belső hálózatát az OP\_TABLET eszközzel! Tegye meg a megfelelő beállításokat ahhoz, hogy a csatlakozás létrejöjjön! Az eszközön állítsa be az automatikus IP-cím kérést!

12. Jelenleg a termelési adatok még nem továbbíthatók a távoli szerverre (SERVER). Ehhez további beállítások is szükségesek. Ennek érdekében tegye meg az alábbi beállításokat:

- Az OP\_RT forgalomirányítón állítson be túlterheléses hálózati címfordítást (PAT). Ehhez használja fel az OP\_RT forgalomirányító külső (CENTER forgalomirányító felé menő), ethernet interfészét és a belső (OP\_SW felé menő) ethernet interfészét is. Alkalmazzon a működéshez megfelelő, tetszőleges típusú hozzáférési listát!

13. Mentse a HQ\_SW kapcsoló konfigurációját helyben úgy, hogy az eszköz újraindítása után is megmaradjanak a beállítások!

A hálózat működésének tesztelése:

- A HQ\_PC kliensről elérhető az OP\_RT forgalomirányító TELNET kapcsolattal.
- Az OP\_TABLET-ről elérhető a SERVER kliens a 20.0.0.200 címen keresztül (ping).
- A HQ\_PC kliensről elérhető a SERVER kliens a 20.0.0.200 címen keresztül (ping).

**2. feladat – Szarvasmarhatelep****40 pont**

A Happy-Cows Nyírség Kft. szíven viseli bolygónk jövőjét, és felelős állattartó céggént az állatok jólétét is. Ezért folyamatosan vezetnek be újításokat, amelyek az üvegházhatású gázok mérséklésével vagy a szarvasmarhák kedvezőbb életkörülményeivel vannak összefüggésben. Az egyik telephely egyik istállójában automata fejőrendszert vezettek be, ahol a tejelő szarvasmarhákat teljesen automatikusan fejik, akkor, amikor az állatok azt szükségesnek tartják. Az adatokat egy automata berendezéssel adatbázisba továbbítják. Ennek az adathalmaznak az egyik hétre vonatkozó, már leválogatott és átalakított részével kell ebben a feladatban dolgoznia. Az adatok és a cég is fiktívek. A megoldás során vegye figyelembe a következőket:

- *A képernyőre írást igénylő feladatok eredményének megjelenítése előtt írja a képernyőre a feladat sorszámát (például: 3. feladat)!*
- *Az egyes feladatokban a kiírásokat pontosan a minta szerint készítse el! (Legfeljebb 1 eltérés-pontossággal, az ékezetmentes kiírástól eltekintve.)*
- *Az ékezetmentes kiírások is elfogadottak.*
- *Az azonosítókat kis- és nagybetűkkel is kezdheti, de igyekezzon betartani az adott programozási nyelv konvencióit.*
- *A program megírásakor az állományban lévő adatok helyes szerkezetét nem kell ellenőriznie, feltételezheti, hogy a rendelkezésre álló adatok a leírtaknak megfelelnek.*
- *A megoldását úgy készítse el, hogy az azonos szerkezetű, de tetszőleges bemeneti adatok mellett is helyes eredményt adjon.*

A forrásállomány neve `hozam.txt`, a felépítése a következő: egy sor egy szarvasmarha adott napon lefejt tejét tartalmazza literben. Az adatokat pontosvessző választja el.

A forrásállomány néhány sora:

```
CAE3;3;21
BAB1F2;6;28
BA;5;16
H7;2;24
BAB2;4;21
D1;0;30
BA;2;27
```

Az egyes adatsorokban rendre a következő adatokat tároltuk:

- *azonosító*, a szarvasmarha egyedi azonosítója, csak az angol ábécé nagybetűit és számjegyeket tartalmazhat.
- *nap sorszáma*, 0-6-ig vehet fel értéket, ahol a 0 a vizsgált hét hétfőjét, a 6 a vasárnapját jelöli.
- *mennyiség*, az adott napon az adott szarvasmarha által leadott tej mennyisége literben, egész típusú. (Az adatok minimuma 15 liter, maximuma 30 az állományban.)

A minta második sora alapján a BAB1F2 azonosítójú tehén a vizsgált hét vasárnapján 28 liter tejet adott.

*A feladat a következő oldalon folytatódik.*

A tehenek nem minden nap adnak tejet ebben a gazdaságban. Ha egy nap többször is ad tejet egy tehén, az nem több sorban, hanem az adott napon lefejt mennyiség összesítve szerepel az állományban. Minden szarvasmarha legalább egyszer adott tejet a héten. A rendszer nem rögzíti, ha egy szarvasmarha 0 liter tejet adott, azt úgy kezelik, mintha oda sem ment volna.

1. Készítsen **konzolos alkalmazást** tehenek néven a következő feladatok megoldására!
2. Projektjében tegye elérhetővé az Ön által használt programozási nyelvnek megfelelő, a forrásoknál megadott `java.txt`, vagy az `csharp.txt` állományokban definiált `Tehen` osztályt. Főprogramjában „happyCows” nevű azonosítóval hozzon létre `Tehen` típusú üres listát! Olvassa be és tárolja el az állomány sorait, és adatsorokként dolgozza fel azokat a következő algoritmus szerint:

```
Szöveges id := adatSor.vág(';')[0];
Szöveges nap := adatSor.vág(';')[1];
Szöveges mennyiseg := adatSor.vág(';')[2];
Tehen aktTehen := new Tehen(id); // aktuális tehén

Ha NEM happyCows.tartalmazza(aktTehen) //Ha még nincs a listában
    akkor: happyCows.veddFelListába(aktTehen); //vedd fel
// Ahol van az aktuális tehén a listában,
Egész index=happyCows.index(aktTehen);
// Az adott naphoz rögzítsd a mennyiseg-et:
happyCows[index].eredmenytRogzit(nap, mennyiseg);
```

*A Java-ban a listák elemeinek elérésére használt metódus a `lista.get(index)`, a `lista[index]` helyett.*

*A megadott osztályt tetszőlegesen bővítheti, átalakíthatja a megoldások érdekében.*

3. Hány szarvasmarha adatait tartalmazza a forrás? A beolvasott adatok alapján az eredményt írja ki a képernyőre!
4. Készítsen jellemzőt/metódust a `Tehen` osztályban `hetiTej` néven, amely az adott tehén heti összesített tejmennyiségét adja meg!
5. Készítsen egész típusú értékkel visszatérő jellemzőt/metódust a `Tehen` osztályban `hetiAtlag` néven, amely egy szarvasmarha átlagos fejési eredményét adja vissza, de csak abban az esetben, ha legalább 3 nap adott tejet. Ellenkező esetben a jellemző/metódus `-1` értéket adjon vissza! A kerekítéshez tetszőleges módszert választhat. Használja fel az előző feladatban készített jellemzőt/metódust a feladat megoldásához!
6. Határozza meg a legalább 3 nap tejet adó szarvasmarhákat! A meghatározott szarvasmarhák adatait soronként írja ki a `joltejelok.txt` állományba, a kimeneti minta alapján. A tehén azonosítóját a heti átlagmennyiség követi, amelyet a `hetiAtlag` metódussal/jellemzővel határozott meg. A két adat között az elválasztó karakter szóköz legyen. Hány tehén adatait írta ki az állományba? Az eredményt a minta szerint jelenítse meg!



7. A Happy-Cows Nyírség Kft.-nél úgy gondolták, hogy az újszülött borjak azonosítása során az anya azonosítóját használják fel, és ezt egészítik ki két új karakterrel jobb oldalon. A forrásállományban a BA jelű tehénnek több leszármazottja szerepel, így közvetlen leszármazottja a BAB2 vagy a BAB1 jelű tehén, míg ennek az utódja a BAB1F2 azonosítójú szarvasmarha. Kérjen be a felhasználótól egy azonosítót! Számolja meg, hogy a bekért azonosítójú tehénnek összesen hány leszármazottja van az állományban! Az eredményt a minta szerint jelenítse meg a képernyőn. (Ha a megadott tehén nem létezne az állományban, annak nyilván 0 leszármazottja van.)
8. Készítsen **grafikus alkalmazást** a tehenészet munkájának megkönnyítésére! A grafikus alkalmazás projektjét `tehenAdatok` néven hozza létre! A konzolos alkalmazásból tetszőlegesen emelhet át kódrészleteket.
- a. Alakítsa ki a felhasználói felületet az alábbi minta alapján! Az elrendezésre és a feliratokra figyeljen, a színeket és a méreteket tetszőlegesen választhatja meg, ha az nincs szabályozva. A beviteli mezők alapértékei a mintán látható értékek legyenek.

The screenshot shows a Java Swing window titled "Happy-Cows Nyírség Kft." with a subtitle "Tejhozam lekérdezés". The window contains a text field for "Szarvasmarha azonosítója:" with the value "BA", a slider for "Tej limit:" ranging from 15 to 25 liter with a current value of 20, a dropdown menu for "Szerda", and two buttons: "Lekérdezés" and "Kilépés".

- b. A „Kilépés” gombra az alkalmazás záródjon be.
- c. A konzolos alkalmazásban használt adatokat kell itt is használnia. A `javaCows.txt`, illetve a `csharpCows.txt` állományok tartalmazznak egy nyelvspecifikus adatszerkezetet a szükséges adatokkal. Másolja be a kódjába a megfelelő helyre az adatszerkezetet az adatokkal, és a továbbiakban ezzel dolgozzon!
- d. A legördülő listából ki lehessen választani egy napot hétfőtől vasárnapig. Alapértelmezett érték legyen a szerda.
- e. A csúszka segítségével be lehessen állítani 15 és 25 közötti egész számot. Az alapértelmezett érték legyen 20.
- f. A csúszkán beállított aktuális érték a minta szerint legyen leolvasható. (Kövesse a kiírás a csúszka változását.)
- g. A szöveges beviteli mezőben egy szarvasmarha kódját kell megadnia. Tesztelésre ajánljuk a „BA” és a „BAB2” azonosítókat.

*A feladat a következő oldalon folytatódik.*

- h. A „Lekérdezés” gombra kattintva a mintának megfelelően jelenítse meg az adott szarvasmarhára vonatkozó adatokat, a következők szerint:
- Az eredmény a minta szerint egy felugró ablakban jelenjen meg.
  - Ha az adott napon nem volt fejés, jelenítse meg a „Nem volt fejés” feliratot a minta szerint. (Hibásan megadott kód esetén is ez a felirat jelenjen meg.)
  - Ha volt fejés, de a csúszkán beállított értéknél kevesebb tejet adott, akkor „Az adott napon a mennyiség nem érte el a limitet!” felirat jelenjen meg.
  - Ha volt fejés és elérte a limitet vagy azon is túl volt, akkor meg kell jeleníteni, hogy hány liter a fejés eredménye.

**Minta a konzolos alkalmazáshoz:**

```
3. feladat
Az állomány 22 tehén adatait tartalmazza.

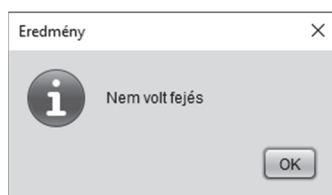
6. feladat
20 darab sort írtam az állományba.

7. feladat
Kérem adja meg egy tehén azonosítóját!
BA
A leszármazottak száma: 6
```

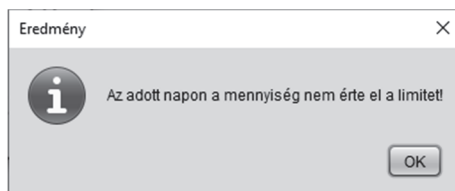
**Minta a kimeneti állományhoz:**

```
BAB1F2 26
BA 23
H7 22
BAB2 23
D1 26
```

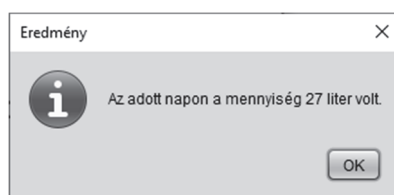
**További minták a grafikus alkalmazáshoz:**



*Eredmény, ha a bemeneti kód „BABAR”*



*Eredmény, ha a szombaton vizsgáljuk a „BA” tehenet, 20 literes limittel*



*Eredmény az alapértékek esetén*

**3. feladat****40 pont****Gitár**

A következő feladatban egy weboldalt kell készítenie a gitár bemutatására a feladatleírás és a minta szerint, valamint gitárművészek adataival kell dolgoznia.


(A két feladatrészt egymástól **függetlenül**, tetszőleges sorrendben megoldható.)

Az első feladatrészben a forrásként kiadott weboldalon kell módosításokat végeznie, a leírás és a minta alapján.

Nyissa meg a `gitar.html` állományt, és szerkessze annak tartalmát az alábbiak szerint:

1. A weboldal karakterkódolása UTF-8, nyelve magyar, a böngésző címsorában megjelenő cím „*Gitárok*” legyen.
2. A weboldal fejrészében helyezzen el hivatkozásokat a CSS/ mappából a `gitar.css` stíluslapra, valamint a JS/ mappából a `gitar.js` állományra a meglévő hivatkozások után.
3. A dokumentum `jumbotron` osztálykijelölővel formázott fejrész keretét formázza a meglévőkön túl a `fej` osztálykijelölővel, és helyezze el benne az `img/` mappában található `fej_kep.png` képet! A képhez sem címkét, sem alternatív szöveget nem kell megadnia.
4. A menüben helyezzen el egy új menüpontot a „*Története*” és „*Felépítése*” menüpontok közé! Az új menüpont szövege „*Hangja*” legyen, és a dokumentum `hangja` azonosítójú elemére mutasson a hivatkozása.
5. Az oldalon a tartalmi blokkok 6:6 arányú megjelenését javítsa 5:7 arányúra!
6. A „*Történet*” címsorhoz tartozó blokkban végezze el a következő módosításokat:
  - a. A címsor utáni bekezdést vágja ketté, a második bekezdés kezdetét egy `*` karakter jelöli! A `*` karaktert törölje!
  - b. A számozott listát alakítsa számozatlan listává, és formázza a `listaKepek` azonosító kijelölővel!
  - c. A blokk végén található `kronika.jpg` kép ne jelenjen meg a weblapon, de a forráskódban továbbra is maradjon ott a jelenlegi kódja.
7. A „*Felépítés*” címsorhoz tartozó blokkban végezze el a következő módosításokat:
  - a. A gombon elhelyezett kép esetén, ha a kép valamiért nem tölthető be, akkor ugyanaz a helyettesítő szöveg jelenjen meg, mint amikor a felhasználó a kép fölé viszi az egeret.
  - b. A blokk hármas szintű címsorait lássa el rendre a `fej`, a `nyak` és a `test` azonosítókkal!
8. Nyissa meg a `gitar.css` állományt, majd módosítsa a következők szerint:
  - a. A `fej` osztályba sorolt elemek belső margója (bélése) legyen 0.
  - b. A `fej` osztályba sorolt elemekben elhelyezett kép magassága 125px legyen.
  - c. Ha a `listaKepek` azonosítójú elemben egy listaelem fölé visszük az egeret, akkor az egérkurzor legyen `pointer` típusú, a háttérszíne pedig `rgb(241, 241, 241)`.
9. Nyissa meg a `gitar.js` állományt, majd módosítsa a `kepcseres()` függvényt a következők szerint:
  - a. Lássa el a függvényt egy `tipus` nevű paraméterrel!
  - b. A függvény első sorában javítsa a betöltendő kép kiterjesztését `.jpg-re`!

**Minta:** (A megoldás szövegének tagolása felbontástól függően eltérhet a képen láthatótól.)




Története Hangja Felépítése Fej Nyak Test

## Történet

A gitár (nemzetközi szó a spanyol guitarra-ból, ez pedig – arab közvetítéssel – a görög kitharából) a húros hangszerek, ezen belül a pengetős hangszerek csoportjába tartozó hangszer. Megszólaltatása pengetővel, illetve ujjal történik. Általában hat húrja van.

A gitár kivitele szerint lehet klasszikus, akusztikus, elektromos vagy elektro-akusztikus. A gitár napjainkra hihetetlen népszerűsége tett szert. Szinte minden zenei stílusban képviselteti magát valamilyen formában. Ez leginkább sokoldalúságának köszönhető, hiszen az egyes gitárok típusuktól és zenei stílustól függően betölthetnek szólista szerepet, de megállják helyüket a ritmusszekcióban is. Gitáron éppúgy előadhatók klasszikus darabok, mint napjaink modern popzenéi. Sőt a rock például elképzelhetetlen lenne e hangszer nélkül.




Népszerű típusai:

- Akusztikus gitár
- Elektromos gitár
- Akusztikus basszusgitár
- Elektromos basszusgitár

A szó szoros értelmében vett gitárok valamikor a 15. század után jelentek meg, de természetesen jóval korábban – már az ókorban is – léteztek a gitárhoz hasonló, vagy arra emlékeztető húros, pengetős hangszerek. A zavaros hangszerelnevezések és hiányos történelmi adatok azonban nehezítik az ennél régebbi ismeretek rendszerezését. A legrégebbi fennmaradt gitár egy 1590 tájáról származó öt hangra hangolt, hangonként dupla húrozással ellátott hangszer.

## Felépítés

Sokféle formájú, húrozású, működési elvű gitár létezik, de változatosságuk ellenére bizonyos vonásaikban megegyeznek. A gitár minden esetben rendelkezik egy hangszertesttel, valamint egy ebből kiinduló nyakkal aminek végén a hangszer feje található a hangolókulcsokkal.



## Fej

A gitár feje a nyak végén található, a hangolókulcsoknak és a húrvezető nyeregnek ad helyet. A húrok a nyergen áthaladva csavarodnak rá a kulcsokra, melyek segítségével szabályozható az egyes húrok feszessége, így a megpengetett hang magassága. A hangolókulcsok elrendezését illetően beszélhetünk szimmetrikus, és aszimmetrikus elrendezésről, az általános elrendezés a 3+3-as, ahol mindkét oldalon 3-3 kulcs található.

## Nyak

A gitár nyakának hangszerjátékos szempontjából legfontosabb funkciója, hogy rajta található a fogólap, aminek segítségével az egyes hangok megszólaltathatóak.

Gitáryak készítésére olyan faanyagot használnak, melynek jó a használat közbeni stabilitása, tehát sem a húrok feszítőereje, sem az idő múlása, az anyag öregedése, a klimatikus változások nem okoznak benne maradandó alakváltozást. Legalkalmasabb anyagok a cedrela, a mahagónifélék, a jávorfa.

A gitár húrjainak feszítettsége miatt a viszonylag vékony és hosszú nyak komoly statikai igénybevételnek van kitéve. A klasszikus gitár húrjai kb. 40 kilogramm súlynak megfelelők, a fémhúros gitároké majdnem kétszer ennyi terhelést képviselnek. Ez az erő a nyakat arra készíti, hogy íjszerűen meggörbüljön. A klasszikus gitár nyaka viszonylag vastag, a műanyag húrok húzóereje kisebb, így rendszerint a faanyag önmagában, megerősítés nélkül képes ezt az igénybevételt elviselni. Ezzel szemben a fémhúros gitárok feszítettsége nagyobb, ugyanakkor hagyományosan szűkebb húrkiosztással, karcsúbb nyakkal készülnek, emiatt a húrok terhelését itt valahogyan ellensúlyozni kell. Erre szolgál a nyakmerevítő pálca, egy enyhén íves fémrúd, mely a hangszer nyakának belsejében fut végig.

## Test

A leghagyományosabb gitártest a klasszikus, akusztikus gitárokra jellemző. Ez a rezgőtest, mely a tető-, hát- és az oldallapokból áll össze. Az ezeken a felületeken, elsősorban a tetőn létrejövő hajlítási hullámok sugározzák a térbe a gitár hangját. Belső térfogata üregrezonátorként működik, melynek nyílását a tetőn lévő kerek hanglyuk alkotja, ez szintén részt vesz a gitárhang megformálásában.

Egy másik megoldás fémből készült rezonátor alkalmazása, amely a nevét viselő rezonátoros gitároknál használatos. A lényege, hogy a húrok egy vagy több, fémből készült tányért hoznak rezgésbe, melyek rendkívül érdekes, fémes hangszínt adnak a hangszernek.

Az elektromos hangszeredővel ellátott gitárok teste lehet tömör (solidbody) vagy féltömör (semi-solidbody).

▶ 0:00 / 0:11 — 🔊 ⋮

*A feladat a következő oldalon folytatódik.*

A második feladatrészben gitárművészek adatait tartalmazó adatbázissal kell dolgoznia.

Rendelkezésre állnak – *a teljesség igénye nélkül* – gitárművészek nevei, nemzetiségi adatai, a munkásságaikat meghatározó zenei stílusuk megnevezése és a zenekarok neve, ahol gitárosként játszottak.

Az adatbázis a következő táblákat tartalmazza:

muvesz

id	Egész szám, a gitárművész azonosítója, PK
muveszNev	Szöveg, a gitárművész neve ( <i>egyedi érték</i> )
stilusID	Egész szám, a zenei stílus azonosítója, FK
nemzetiseg	Szöveg, a gitárművész nemzetisége
	Értéke NULL, ha a gitárművész nemzetisége nincs eltárolva

stilus

id	Egész szám, a zenei stílus azonosítója, PK
stilusNev	Szöveg, a zenei stílus neve ( <i>egyedi érték</i> )

zenekar

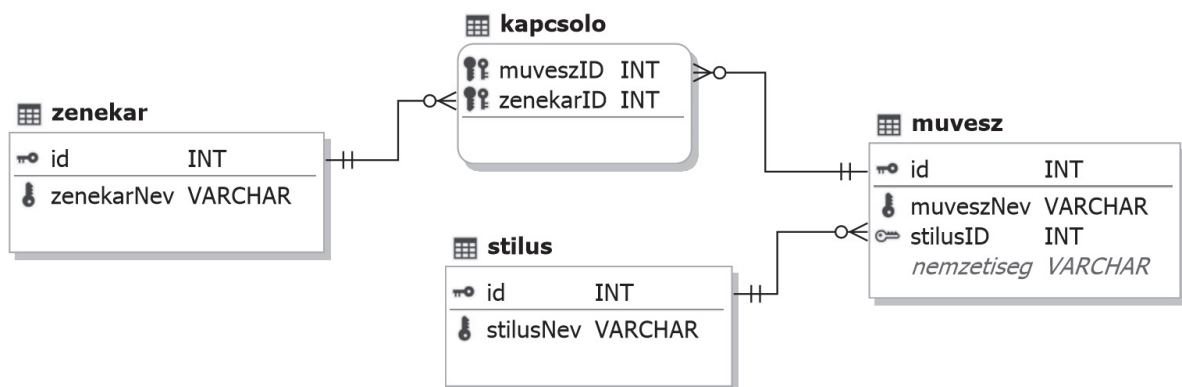
id	Egész szám, a zenekar azonosítója, PK
zenekarNev	Szöveg, a zenekar neve ( <i>egyedi érték</i> )

kapcsolo

muveszID	Egész szám, a gitárművész azonosítója, FK
zenekarID	Egész szám, a zenekar azonosítója, FK

Az elsődleges kulcsokat PK-val, az idegen kulcsokat FK-val jelöltük!

Az adattáblák közti kapcsolatokat az alábbi ábra mutatja:



A feladatok megoldására elkészített SQL parancsokat a megoldasok.sql állományba illessze be, a feladatok végén zárójelben jelölt sor alá!

**A javítás során csak ennek az állománynak a tartalma lesz értékelve.**

Ügyeljen arra, hogy a lekérdezésben pontosan a kívánt mezők és mezőnevek szerepeljenek, és felesleges mezőt ne jelenítsen meg.

10. Hozzon létre a lokális SQL serveren gitarmuveszek néven adatbázist! Állítsa be az UTF-8 kódolást alapértelmezettnek az adatbázis létrehozásánál! Az adatbázis alapértelmezett rendezési sorrendje a magyar szabályok szerinti legyen. (10. feladat)

11. Az `adatbazis.sql` állomány tartalmazza a táblákat létrehozó és az adatokat a táblába beszűrő SQL parancsokat! Futtassa az `adatbazis.sql` parancsfájlt a gitarmuveszek adatbázisban!
12. Törölje a stílusok táblájából a „swing” rekordját! A törlést a stílus azonosítójának meghatározása nélkül végezze el! (12. feladat)
13. Határozza meg, hogy összesen hány zenekar szerepelt az adatbázisban! Állítsa be a számított mező címkéjét a minta szerint! (13. feladat)

darab
84

14. Készítsen lekérdezést, mely megjeleníti az adatbázisban fellelhető különböző nemzetiségeket – ismétlődés nélkül – ábécérendben! *Ügyeljen rá, hogy csak az ismert nemzetiségek jelenjenek meg a listában!* (14. feladat)

nemzetiseg
amerikai
amerikai-olasz
...
svéd

15. Készítsen lekérdezést, mely megjeleníti stílusonként a rögzített gitárművészek számát! A rekordokat rendezze a számított mező alapján a minta szerint! *A lekérdezésben kihasználhatja, hogy nincs két azonos nevű stílus.* (15. feladat)

stílus	muveszekSzama
rock	71
...	
punk	4

16. Készítsen lekérdezést, amely megjeleníti ábécérendben azon zenekarok nevét, melyek legalább 3 szóból állnak, vagyis a zenekar nevében legalább 2 szóköz található! Jelenítse meg a gitárosok nevét is! (16. feladat)

zenekarNev	muveszNev
...	
Guns N' Roses	Robin Finck
Guns N' Roses	Slash
Hobo Blues Band	Fehér Géza
...	

#### Források:

- [https://cdn.fuelrocks.com/does\\_large\\_headstock\\_effect\\_guitar\\_sound.jpg](https://cdn.fuelrocks.com/does_large_headstock_effect_guitar_sound.jpg) (Utolsó letöltés dátuma: 2024.01.15.)
- [https://cdn.icon-icons.com/icons2/1660/PNG/512/38444466-in-magnifier-plus-search-zoom\\_110347.png](https://cdn.icon-icons.com/icons2/1660/PNG/512/38444466-in-magnifier-plus-search-zoom_110347.png) (Utolsó letöltés dátuma: 2024.01.15.)
- <https://hu.wikipedia.org/wiki/Gitár> (Utolsó letöltés dátuma: 2024.01.15.)
- [https://hu.wikipedia.org/wiki/Gitárosok\\_listája\\_stílusuk\\_szerint](https://hu.wikipedia.org/wiki/Gitárosok_listája_stílusuk_szerint) (Utolsó letöltés dátuma: 2024.01.15.)

témakörök	a feladat sorszáma	pontszám	
		maximális	elért
Hálózati ismeretek	1.	<b>40</b>	
Programozás	2.	<b>40</b>	
Weboldalak kódolása és adatbázis-kezelés	3.	<b>40</b>	
<b>A gyakorlati vizsgarész pontszáma</b>		<b>120</b>	

\_\_\_\_\_

dátum

\_\_\_\_\_

javító tanár

	pontszáma <b>egész</b> <b>sámra</b> kerekítve	
	elért	programba beírt
Számítógépen megoldott gyakorlati feladatok		

\_\_\_\_\_

dátum

\_\_\_\_\_

dátum

\_\_\_\_\_

javító tanár

\_\_\_\_\_

jegyző