

Azonosító
jel:

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

ÉRETTSÉGI VIZSGA • 2020. május 14.

**INFORMATIKAI
ISMERETEK**

**EMELT SZINTŰ
GYAKORLATI VIZSGA**

2020. május 14. 8:00

Időtartam: 240 perc

Pótlapok száma	
Tisztázati	
Piszkozati	

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

Java ☐

C# ☐

EMBERI ERŐFORRÁSOK MINISZTERIUMA

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Fontos tudnivalók

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

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 feladatlapon található azonosítóval megegyező nevű vizsgakönyvtárba 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 vizsgakönyvtárban, 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.

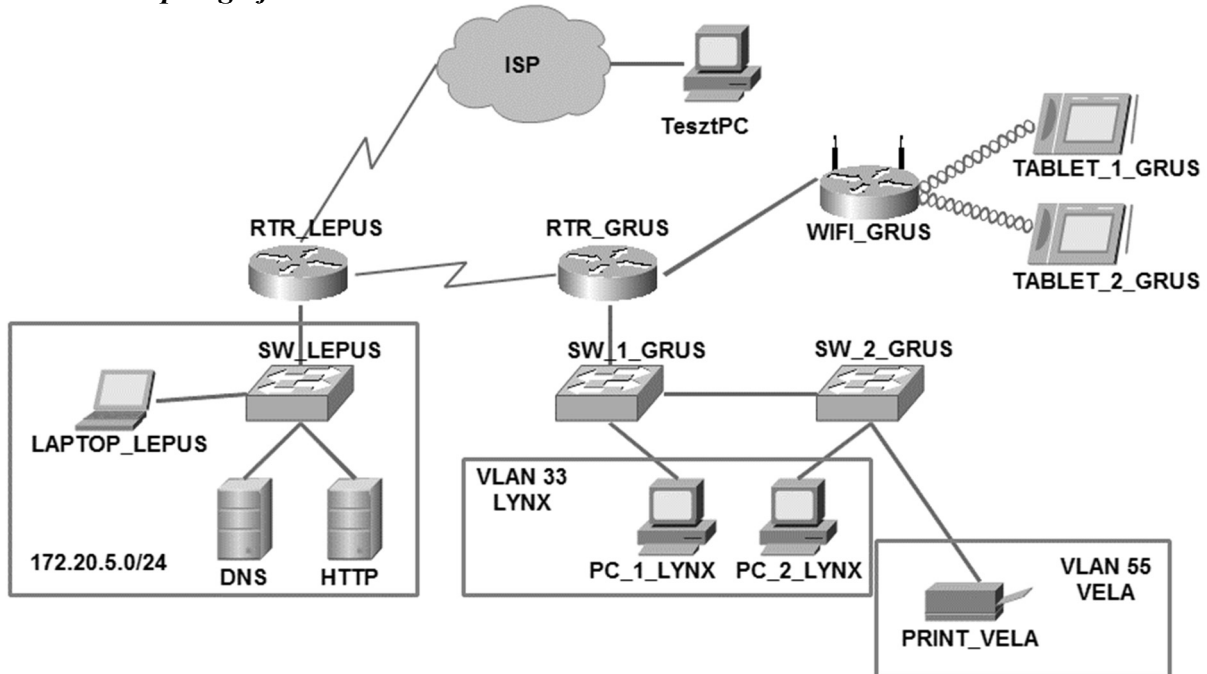
Amennyiben számítógépével műszaki probléma van, jelezze a felügyelő tanárnak! A jelzés ténye és a megállapított hiba jegyzőkönyvezésre kerül. A kiesett idővel a vizsga ideje hosszabb lesz. Amennyiben a hiba mégsem számítógépes eredetű, a javító tanár értékeléskor köteles figyelembe venni a jegyzőkönyv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

1. GRUS-LAN

40 pont

Egy terjeszkedés alatt álló vállalat hálózatának tervezésével bízták meg. A vállalat jelenleg két telephellyel rendelkezik. Feladata, hogy a megadott tervek és elváráslista alapján szimulációs programmal elkészítse a vállalat teszhálózatát.

A hálózat topológiája



Beállítások

1. Töltse be a `grus.pkt` állományt a szimulációs programba! A teszhálózat már tartalmazza a vállalat összes hálózati eszközét és az internet szimulálására szolgáló eszközöket. Ez utóbbiak már beállításra kerültek. A vállalati eszközök részleges konfigurációval már rendelkeznek, Önnek csak a feladatokban leírt módosításokat kell elvégeznie.
2. Az `RTR_GRUS` forgalomirányító és a `WIFI_GRUS` vezeték nélküli forgalomirányító közti kapcsolaton használja a `192.168.80.0/30` hálózatot! A hálózat első címét az `RTR_GRUS` eszköznek állítsa be, a második cím a `WIFI_GRUS` eszköz Internet portjára legyen! Ez utóbbi eszközönél állítsa be a megfelelő alapértelmezett átjárót is!
3. A GRUS hálózatban a `192.168.50.0/24` privát címtartományt szeretnék használni. A két VLAN számára VLSM használatával a lehető leghatékonyabban alakítson ki alhálózatokat a következő IP-cím igények figyelembevételével:

VLAN száma	VLAN neve	Igényelt IP-címek száma
33	LYNX	50
55	VELA	12

Az `ipcimzes.txt` fájlban a példához hasonló módon rögzítse számolásának eredményét!

Ha nem tudja elvégezni az alhálózatszámolást, akkor a továbbiakban a következő IP-címekkel dolgozzon:

VLAN azonosító	VLAN neve	Hálózat cím	Netmaszk
33	LYNX	172.28.33.0	255.255.255.128
55	VELA	172.28.55.0	255.255.255.224

A feladat a következő oldalon folytatódik

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

4. Az RTR_GRUS forgalomirányítón hozza létre a szükséges alinterfészeket úgy, hogy az alinterfészek azonosító száma egyezzen meg a használt VLAN-ok azonosítójával! Az alinterfészekre állítsa be a megfelelő hálózat első kiosztható IP-címét!
5. Az SW_2_GRUS kapcsolón hozza létre a VLAN 33-at és a VLAN 55-öt, és a fenti táblázat alapján állítsa be a VLAN-oknak a LYNX és a VELA nevet! (Az SW_1_GRUS kapcsolón már léteznek a szükséges VLAN-ok.)
6. Az SW_1_GRUS és az SW_2_GRUS kapcsolók megfelelő portjainak konfigurálásával érje el, hogy a kliensekhez csatlakozó portok hozzáférési portok legyenek és a megfelelő VLAN-ba kerüljenek!
7. Az SW_1_GRUS és az SW_2_GRUS kapcsolók egymáshoz csatlakozó portjait és az SW_1_GRUS kapcsoló forgalomirányítóhoz csatlakozó portját állítsa be trónk módúra!
8. A PRINT_VELA nyomtatónak statikusan állítsa be a megfelelő hálózat utolsó kiosztható IP-címét, a megfelelő alapértelmezett átjárót és DNS kiszolgálót (172.20.5.15)!
9. A LYNX VLAN számára hozzon létre egy DHCP hatókört az RTR_GRUS forgalomirányítón:
 - a. Biztosítsa az összes szükséges paraméter átadását!
 - b. A DNS szolgáltatást a 172.20.5.15 IP-című DNS szerver biztosítja!
 - c. Az első 13 címet ne oszthassa ki a DHCP kiszolgáló!
10. Állítsa be a LYNX VLAN számítógépeit a dinamikus IP-cím használatához!
11. Az IPv4-es forgalom irányításához OSPF protokollt használnak. Állítsa be mindkét forgalomirányítón az OSPF protokollt a következők szerint:
 - a. Használja a 10-es folyamatazonosítót!
 - b. Mindkét forgalomirányítón hirdesse az összes közvetlenül csatlakozó hálózatot a 0-s területben, kivéve az RTR_LEPUS forgalomirányítón, itt az internet (ISP) felé menő hálózatot ne hirdesse!
 - c. A forgalomirányítási információk küldésére nem használt (al)interfészeket állítsa be passzívnak!
12. Az RTR_LEPUS forgalomirányítón vegyen fel az internet felé (ISP-hez) vezető alapértelmezett statikus útvonalat! A megadásnál használja a kimenő interfészt! Az RTR_LEPUS forgalomirányítón futó OSPF folyamat kiegészítésével érje el, hogy a másik forgalomirányító is megtanulja az alapértelmezett útvonalat!
13. A LAPTOP_LEPUS kliens statikus IP konfigurációjában egy hibát vétettek, ezért a laptop jelenleg semelyik eszközt sem éri el a hálózaton. Keresse meg és hárítsa el a konfigurációban található hibát!
14. A WIFI_GRUS vezeték nélküli eszközön állítsa át az SSID értékét **GRUSWLAN**-ra és állítson be WPA2 hitelesítést **GRUS12345** jelszóval!
15. A WIFI_GRUS vezeték nélküli eszköz DHCP szolgáltatását egészítse ki úgy, hogy a csatlakoztatott kliensek megkapják a megfelelő DNS kiszolgáló IP-címét (172.20.5.15) is!
16. Csatlakoztassa a vezeték nélküli klienseket (TABLET_1_GRUS, TABLET_2_GRUS) a WIFI_GRUS eszközökhöz!

A feladat a következő oldalon folytatódik

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

17. Az RTR_LEPUS forgalomirányítón állítson be statikus NAT szolgáltatást, amellyel biztosítja, hogy a HTTP szerver kívülről (a TesztPC-ről) a 98.70.50.5 IP-címmel legyen elérhető!
18. Az RTR_LEPUS forgalomirányítón állítson be dinamikus túlterheléses NAT (PAT) szolgáltatást, amellyel biztosítja, hogy a belső hálózatot elhagyó csomagok forráscíme az RTR_LEPUS forgalomirányító internethez (ISP-hez) csatlakozó címére forduljanak le!
19. Az RTR_LEPUS forgalomirányítóhoz csatlakozó hálózat számára IPv6-os elérést is kell biztosítani. Az RTR_LEPUS forgalomirányítón az alábbiak szerint állítson be IPv6-címeket:
 - a. Serial0/0/1 interfész globális IPv6 cím: 2020:20::1/64
 - b. GigabitEthernet0/1 interfész globális IPv6 cím: 3232:30:20::1/64, link-local cím: FE80::1
20. Az RTR_LEPUS forgalomirányítón vegyen fel alapértelmezett IPv6 útvonalat az internet (ISP) irányába a 2020:20::2 következő ugrás cím használatával!
21. A DNS szervernek állítsa be a 3232:30:20::15/64 IPv6-címet, a HTTP szervernek pedig a 3232:30:20::20/64 IPv6 címet! Az alapértelmezett átjáró mindkét esetben a forgalomirányító link-local címe legyen!
22. A LAPTOP_LEPUS kliens SLAAC segítségével jusson IPv6-címhez!
23. A forgalomirányítók és a kapcsolók mentse el a konfigurációt úgy, hogy azok újraindítás után is megőrizzék a beállításokat!

Hálózat működésének tesztelése:

- A forgalomirányítás működik a forgalomirányítók között, a routing táblában jelennek meg OSPF-től tanult bejegyzések
- A LYNX VLAN kliens gépeiről elérhető az ipv4.web (84.20.10.10)
- A TesztPC-ről elérhető a http.lepus (98.70.50.5) és közben a statikus NAT működik
- Az IPv6-os hálózatból (pl LAPTOP_LEPUS) elérhető a 3030:10:20::10

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

2. Playfair-négyzet

40 pont

A Playfair-négyzet egy kódolási módszer, melyet 1854-ben Charles Wheatstone talált fel, de legfőbb támogatójának, Lord Playfairnek a nevét viseli. A módszer az angol szövegből megadott szabályok szerint betűpárokat képez, majd egy 5×5 -ös tábla felhasználásával kódolja azt. Ebben a feladatban már rendelkezésre állnak a szabály szerint kialakított betűpárok, Önnek a kódolással kapcsolatban kell feladatokat megoldania. A megoldás során vegye figyelembe a következőket:

- *Megoldását választása szerint Java vagy C# programozási nyelven kell elkészítenie! Az Ön által választott programozási nyelvet jelölje meg a feladatlap fedőlapján! A javítás során csak a megjelölt nyelven készült megoldás lesz értékelve!*
- *A program készítése során törekedjen az objektum orientált (OOP) megoldásra, amire a feladatsor ajánlásokat is tartalmaz. Amennyiben a programot ilyen módon nem tudja elkészíteni, akkor a feladatokat saját osztály létrehozása nélkül is megoldhatja, de így kevesebb pontot ér a megoldása. Ebben az esetben, ha a feladat jellemző vagy metódus létrehozását kéri, akkor Önnek saját alprogramot (függvényt, eljárást) kell készítenie, amely **paramétereken keresztül** kommunikál a hívó programmal!*
- *A képernyőre írást igénylő részfeladatok eredményének megjelenítése előtt írja a képernyőre a feladat sorszámát (például: 6. feladat)!*
- *Az egyes feladatokban a kiírásokat a minta szerint készítse el!*
- *Az ékezetmentes kiírások is elfogadottak!*
- *Az azonosítókat kis- vagy nagybetűkkel is kezdheti!*
- *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 `kulcstabla.txt` forrásállomány tartalmazza a kódoláshoz szükséges 5×5 -ös **kulcstáblát**. Mivel az angol ábécé 26 betűből áll, így a „Q” karaktert nem tudja a módszer kódolni:

P	L	A	Y	F
I	R	E	X	M
B	C	D	G	H
K	N	O	J	S
T	U	V	W	Z

1. Készítsen konzolos alkalmazást a következő feladatok megoldására, melynek a projektjét `playfair` néven mentse el!
2. Hozzon létre saját osztályt `PlayfairKodoLo` azonosítóval, melynek konstruktora legyen alkalmas paraméteren keresztül a forrásállomány nevének átadására!
3. Az osztály konstruktora tárolja el a paraméterben átadott állományból az adatokat a feladatok megoldásához célszerűen megválasztott típusú adattagban (például: karakterlánc vektor, karakterlánc lista, karakter mátrix, stb.)! A feladat megoldásához vizsgálja meg a `kulcstabla.txt` állomány felépítését!
4. Hozzon létre osztálypéldányt (objektumot) a `PlayfairKodoLo` osztályból, paraméterezze konstruktorát a forrásállomány nevével (`kulcstabla.txt`)!

A feladat a következő oldalon folytatódik

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

- Készítsen a `PlayfairKodolo` osztályba egész típusú adattal visszatérő metódusokat `SorIndex()` és `OszlopIndex()` azonosítóval, melyek a paraméterben megadott **nagybetűs** karakter helyét (indexét) határozzák meg értelemszerűen a kulcstáblában! A sorok és oszlopok indexelése nullával indul. Ha a megadott karakter nem található a kulcstáblában, akkor a metódusok -1 értékkel térjenek vissza!
- Kérjen be a felhasználótól egy nagybetűt az angol ábécéből, majd határozza meg és írja ki a minta szerint az indexeit! Megoldása több pontot ér, ha felhasználja az előző feladatban elkészített metódusokat!
- Készítsen a `PlayfairKodolo` osztályba karakterlánc típusú adattal visszatérő metódust `KodolBetupar()` azonosítóval, amely a paraméterben megadott nagybetűs, két karakter hosszúságú betűpárt kódolja a következő szabályok szerint:

P	L	A	Y	F
I	R	E	X	M
B	C	D	G	H
K	N	O	Q	S
T	U	V	W	Z

- Ha a betűpár mindkét betűje ugyanabban a **sorban** jelenik meg a kulcstáblán, akkor a tőlük közvetlenül jobbra állóval kell helyettesíteni őket!

Például: **EX** -> **XM**

- Ha történetesen az egyik betű a sor jobb szélén van, akkor a sor bal szélén álló betűvel kell helyettesíteni!

Például: **MR** -> **IE**

- Ha a betűpár mindkét betűje ugyanabban az **oszlopban** jelenik meg a kulcstáblán, akkor közvetlenül az alattuk állóval kell helyettesíteni őket!

Például: **ED** -> **DO**

- Ha az egyik betű az oszlop alján van, akkor az oszlop tetején álló betűvel kell helyettesíteni!

Például: **EV** -> **DA**

P	L	A	Y	F
I	R	E	X	M
B	C	D	G	H
K	N	O	Q	S
T	U	V	W	Z

P	L	A	Y	F
I	R	E	X	M
B	C	D	G	H
K	N	O	Q	S
T	U	V	W	Z

- Ha a betűpár betűi nincsenek sem egy sorban, sem egy oszlopban, akkor tekintsük azt a kulcstábla mezőiből felépülő **téglalapot**, amelynek a két betű a két szemközti csúcsa. A betűket a saját sorukban, a téglalap másik csúcsánál lévő betűkkel helyettesítjük.

Például: **IH** -> **MB**

Az osztályát tetszőleges kód- és adattagokkal bővítheti a feladat megoldásához!

- Kérjen be a felhasználótól egy nagybetűs karakterpárt az angol ábécéből, majd határozza meg és írja ki a minta szerint a kódolt megfelelőjét! Megoldása több pontot ér, ha felhasználja az előző feladatban elkészített metódust!
- Készítsen grafikus alkalmazást, melynek a projektjét `playfairGUI` néven mentse el!

A feladat a következő oldalon folytatódik

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

10. A grafikus alkalmazásban a következő feladatokat végezze el:

- A kódolandó szöveg ellenőrzését végző alkalmazás felhasználói felületét alakítsa ki egy címkével és egy többsoros bevitelmezővel a minta szerint! Az ablak címsorában megjelenő felirat (title) „playfairGUI” legyen!
- A többsoros bevitelmező alapértelmezett értékét állítsa be a minta szerint! A karakterláncot a `szoveg.txt` állományból tudja kimásolni!
- Az „Előkészített szöveg” címke betűszíne a program indulásakor legyen zöld!
- Az „Előkészített szöveg” címke betűszíne váltson azonnal pirosra, ha szöveg nem két karakter hosszúságú betűpárokból áll, melyek 1-1 darab szóközzel tagoltak!
- Az „Előkészített szöveg” címke betűszíne váltson azonnal kékre, ha a „d” pontban leírt hibák nem jelentkeznek, de legalább egy karakter nem az angol ábécé nagybetűs karaktere! Például: ”MG KR HN SL Kt BP”
- Az „Előkészített szöveg” címke betű színe váltson azonnal magentára, ha a „d” és az „e” pontban leírt hibák nem jelentkeznek, de legalább egy betűpárt azonos karakterek alkotnak! Például: ”MG KR HN SL KT BP PP”

Konzolos minta:

```
6. feladat - Kérek egy nagybetűt: S
A karakter sorának az indexe: 3
A karakter oszlopának az indexe: 4
8. feladat - Kérek egy karakterpárt: DA
Kódolva: OE
```

Grafikus minta:



--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

3. Oldtimer

40 pont

A következő feladatban egy veterán autókat kölcsönző vállalkozás komplex webalkalmazásának fejlesztésében kell részfeladatokat elvégeznie. Az Ön feladata két részből áll:

- a nyilvánosságnak szánt weboldal véglegesítése („Weboldalak kódolása” feladatrész);
- adatbázis lekérdezések elkészítése az üzletvezetés által használt adminisztrációs oldal számára („Adatbázis-kezelés” feladatrész).

A webalkalmazás¹ főbb elemei már rendelkezésére állnak a vizsgakönyvtár 3_feladat\oldtimer mappájában. Csak a feladatokban felsorolt állományokat módosítsa!

Amennyiben munkája közben a mappa tartalmában véletlenül olyan módosításokat végezne, ami után a kiszolgáló nem működik megfelelően, akkor a helyreállításhoz a mappa eredeti tartalmát megtalálja a forrásállományok között található oldtimer.zip tömörített archív állományban.

Figyelmeztetés: a teljes archívum kitömörítése több percig is eltarthat!

A webszerver indításához a Windows parancssorában tegye aktuálissá a vizsgakönyvtár oldtimer nevű mappáját, majd futtassa az `npm run start` parancsot.

Weboldalak kódolása

A webszerver indítása után az autókölcsönző nyilvános oldala a `http://localhost:8000` URL-en keresztül érhető el.

Az alábbi utasításoknak megfelelően végezze el a nyitólap fejlesztését. Ügyeljen rá, hogy az oldal a módosítások után is megtartsa a reszpönzív viselkedését.

A feladat megoldása során a következő állományokat kell módosítania:

- 3_feladat\oldtimer\web\index.html
- 3_feladat\oldtimer\web\oldtimer.css
- 3_feladat\oldtimer\web\oldtimer.js

A munkája végén ezeket az állományokat a feladatlap „Az elkészült munka beadása” részben megadottak szerint a vizsgakönyvtár 3_feladat\beadott mappájába kell másolnia!

A formázási beállításokat az oldtimer.css stílusállományban végezze el, lehetőleg úgy, hogy az új szelektorok létrehozása a stílusállomány végén történjen!

1. Állítsa be, hogy a böngésző címsorában megjelenő cím „Autókölcsönzés” legyen!
2. Helyezzen el hivatkozást az oldtimer.css stíluslapra! Ügyeljen rá, hogy az oldtimer.css stíluslap formázásai felülírják a bootstrap.min.css formázási beállításait.
3. Az egész dokumentumra vonatkozóan a használt generikus betűcsalád sans-serif legyen. Állítson be háttérszínnek #1714149c értéket minden dia-elem-szovegdoboz osztályú elemre!

A feladat a következő oldalon folytatódik

¹ A webalkalmazás szerver oldali komponense NodeJS és ExpressJS technológia segítségével készült.

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

4. Alakítsa ki a weboldal navigációs részét az alábbiak szerint:
 - a. „*Navigáció*” részben hozza létre az alábbi menüelemeket:
 - Rólunk
 - Kínálat
 - Hírlevél
 - Kapcsolat
 - b. Ehhez a „****Menüelemek****” szöveget cserélje ki egy négyelemű számozatlan listára, a lista külső HTML-eleme () navbar-nav és ml-auto, a listaelemek pedig nav-item osztálykijelölővel legyenek formázva!
 - c. Az egyes listaelemekhez hozzon létre nav-link osztályba sorolt hivatkozásokat, melyek sorrendben a rolunk, kinalat, hirlevel és kapcsolat horgonyra mutatnak!
5. A „*Diasor szekcióban*” található dia osztályú keretekben a „****Kép helye****” szöveget cserélje le a kepek/moszkvics_fejlec_900_350.jpg, kepek/lada_fejlec_900_350.jpg és kepek/polszki_fejlec_900_350.jpg képekre! Alkalmazza a képekre a d-block, img-fluid és a w-100 Bootstrap osztályokat!
6. „*Diasor*” szekcióban az egyik dia osztályú keretben található szöveg megjelenése különbözik a másik kettőtől. Keresse meg a hibát és javítsa ki!
7. Állítsa be a dokumentum navigációs sávján (<nav>) belül a betűvastagságot a következők szerint: állapítsa meg böngészőjének fejlesztői eszköze segítségével, hogy mennyi az aktuális vastagság, és állítson be ennél 200 egységgel nagyobb értéket!
8. A „*Kínálat*” szekcióban jelenleg 5 autó kártyája szerepel. Szűrjön be a jelenlegi kártyasor mintájára, a kártyasor végére egy újabb kártyát (így összesen 6 autó szerepel majd a kínálatban)! Ennek paramétereit a vizsgakönyvtár Forras\3. Oldtimer mappájában található mustang.txt forrásfájlban találja meg!
9. A kártyák számára hozzon létre reszponzív viselkedést biztosító rácsot (grid-et)! Közepes (>=768px) és nagy kijelző méret esetén két sorban, soronként három-három kártyát, míg kicsi és extra kicsi kijelzőjű mobil eszközök esetén egyesével, soronként egy-egy kártyát jelenítsen meg! *(Ne feledje, a feladatsor megoldása során használhatja a Zeal dokumentumkezelőt, amiben Bootstrap dokumentációt is talál.)*
10. Helyezzen el hivatkozást az oldtimer.js JavaScript fájlra a weboldal végén!
11. Az oldtimer.js fájlban hozzon létre olyan JavaScript kódot, ami a szervertől Ajax kéréssel (a weboldal betöltődése után) lekérdezi, hogy melyik autó nem érhető el átmenetileg, és a válaszüzenetben kapott autó(k) nevét betölti a nem-elerheto tagkijelölővel rendelkező bekezdésbe! A megfelelő szerver oldali végpont az alábbi jellemzőkkel bír:
 - Kérés típusa: GET
 - URL: api/nemelerheto
 - Választípus: JSON
 - Válasz adat: {nemElerhetoAutok}*

* Példák a kiszolgáló válaszára:

```
{nemElerhetoAutok: "Ferrari Spider 1992, Moszkvics 408"}  
{nemElerhetoAutok: "Jelenleg minden autó elérhető!"}
```

A feladat a következő oldalon folytatódik

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

12. Amennyiben nem tudja elvégezni az adatlekérést a szerverről, illessze be a "Jelenleg minden autó elérhető!" szöveget JavaScript kód segítségével a nem-elérhető tagkijelölővel rendelkező bekezdésbe!.

13. Írjon olyan JavaScript kódot, ami a weboldal „*Vélemények*” szekcióban található <textarea> mezőbe beírt megjegyzést elküldi a webszervernek.

- Kérés típusa: POST
- URL: /api/velemeney
- Elküldött adat típusa: JSON
- Elküldött adat: {velemeney} (string objektum, értéke a szövegdobozban található szöveg. Pl: {velemeney: "Tetszett a szolgáltatás!"})
- Választípus: JSON
- Válasz adat: {velemeney} (az elküldött objektummal egyezik meg)

Amennyiben a küldés sikeres (200-as státuszkód), törölje a weboldalon a bejegyzést a szövegdobozból, és egy felugró ablakban jelenítse meg az alábbi üzenetet: „*Véleménye fontos számunkra!*”!


A feladat a következő oldalon folytatódik

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

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

Oldtimer Veteránautó

RólunkKínálatHírlévélKapcsolat



Élje át gyerekkorát!


Ezek a matuzsálemek mindenki szívében egy parányi nostalgiát hordoznak

Veteránautó kölcsönzés

Rólunk


Az Oldtimer Veteránautó-kölcsönző cég küldetése, hogy ügyfelei számára visszahozza a múlt egy igen jellemző szelétét. Autóinkat a lehető legkülönbözőbb korokból válogattuk össze, kifejezetten azzal a céllal, hogy visszarépíthessük Önt az elmúlt évszázad autóiapri csodáinak birodalmába!

Kínálatunk




Trabant 601
Ár: 2.500 Ft. / óra
A Trabant keletnémet gépkocsitípus, melyet a német VEB Sachsenring gyártott, a legelterjedtebb autó volt Kelet-Németországban.

Foglalás




Moszkvics 408
\$24.99
A Moszkvics egy Moszkvában 1947 és 1968 között gyártott szovjet autómárka. A Moszkvicskij Zavod Malilitratsznh Avtomobilej (MZMA) készítette, akár a 130 km/h-s sebességre is képes

Foglalás




DeLorean DMC-12
Ár: 12.000 huf / óra
Ezt a világhírű sportautót a John DeLorean által alapított DeLorean Motor Company gyártotta 1981-ben. A DeLorean egy módosított változata a Vissza a jövőbe filmtrilógiában szerepelt mint időgép.

Foglalás




Cadillac Fleetwood Brougham
Ár: 15.000 Ft. / óra
Ezt a luxusautót az amerikai Cadillac gyár kapuján gurult ki 1977 és 1986 között. A Fleetwood jelzővel az összes lehetséges extrával felszerelt modell illeték.

Foglalás



Ferrari 348
Ár: 18.000 huf / óra
A Ferrari 348-as 1989 és 1995 között készült, egyedi V8-as motorral ellátott sportautó. Ez volt a legutolsó modell, amit még maga Enzo Ferrari tervezett.

Foglalás



1970 Ford Mustang
Ár: 20.000 huf / óra
Eredetileg 1969 és 1970 között készült, módosított Ford Mustang Boss 302, mely nem összekeverendő a 2012-ben újraélesztett, hasonló nevű modellel. Ez volt az a szőria, ami igazán népszerűvé tette a márkát.

Foglalás

Átmenetileg nem elérhető autóink:

Ferrari Spider 1992, Moszkvics 408

Feliratkozás a hírlevélre:

Értesüljön elsőként akcióinkról, és új autóinkról!

Email cím

Feliratkozás

Bérelt már nálunk?

Mondja el véleményét!

Beküldés

Oldtimer Veteránautó-kölcsönző
9999 Réghalomb
Fő út 1
Telefon: +36 55 555-5555
Email: kapcsolat@veteranautokolcsenzo.hu

A feladat a következő oldalon folytatódik

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Adatbázis-kezelés

Az adatbázis az autókölcsönző tulajdonában álló autóparkot és a 2019-es év januári bérleseit tartalmazza. Az Ön feladata az SQL lekérdezések elkészítése és beillesztése a vizsgakönyvtárban található

3_feladat\oldtimer\lekerdezések\lekerdezések.sql fájlba. A munkája végén ezt az állományt a feladatlap „Az elkészült munka beadása” részében megadottak szerint a vizsgakönyvtár 3_feladat\beadott mappájába kell másolnia! Az adatbázis-kezelési feladatok javítása során ennek az állománynak a tartalma lesz csak értékelve.

Az üzletvezetés számára is készült egy adminisztrációs oldal, amin keresztül az autóbérléssel kapcsolatos információkat és statisztikákat lehet megjeleníteni. Az oldalt a böngészőjében a <http://localhost:8000/admin> URL-en keresztül nyithatja meg. Az adatokat a rendszer adatbázisából kérdezi le a szerveralkalmazás, ami már fel van készítve arra, hogy a 3_feladat\oldtimer\lekerdezések\lekerdezések.sql fájlból kiolvassa a megfelelő SQL utasítást, és ennek megfelelően szolgáltatson adatokat az adminisztrációs oldal számára. A megoldásait ennek megfelelően ellenőrizheti a böngészőjében a <http://localhost:8000/admin> URL megnyitásával is. Itt ellenőrizheti a szerver és az adatbázis elérhetőségét is.

A következő adatbázis-kezelési feladatok megoldására elkészített SQL parancsokat a lekerdezések.sql állományba illessze be a feladatok végén zárójelben jelölt elem értékeként! Figyeljen rá, hogy az állomány szerkezetét ne módosítsa (pl. ne törölje a *** karaktereket tartalmazó elválasztó sorokat), mert ellenkező esetben az adminisztrációs oldal nem lesz képes megjeleníteni a lekérdezések eredményét. Az SQL parancsokat több sorba is törheti. Az adminisztrációs oldal csupán kiegészítő lehetőséget kínál a megadott lekérdezések ellenőrzésére, a helyes megoldásra abban az esetben is megkapja a pontot, ha az adminisztrációs oldal valamilyen oknál fogva nem képes az eredmény megjelenítésére.

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

A feladat a következő oldalon folytatódik

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Az adatbázis a következő táblákat tartalmazza:

kategoriak

id	Egész szám, az autó kategória típus azonosítója, PK
nev	Szöveg, a kategória megnevezése

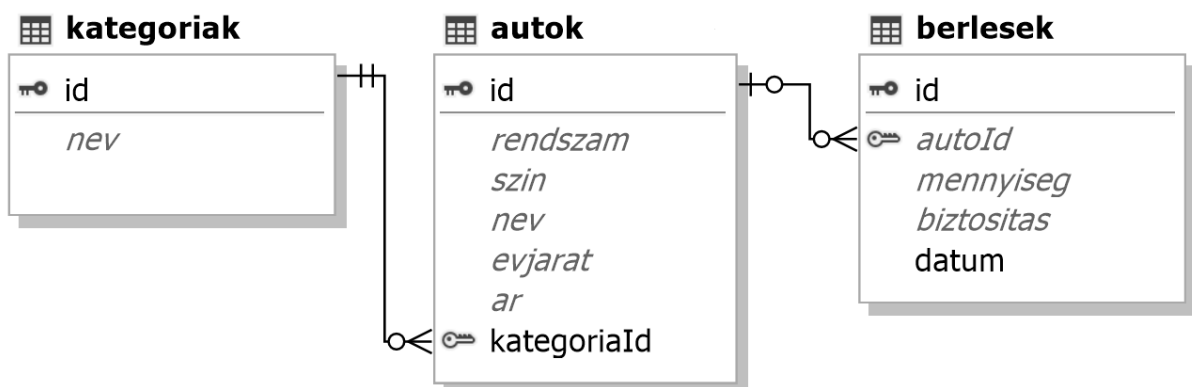
autok

id	Egész szám, az autó egyedi azonosítója, PK
rendszam	Szöveg, az autó rendszáma
szin	Szöveg, az autó színezése
nev	Szöveg, az autó elnevezése
evjarat	Egész szám, évszám, az autó forgalomba állításának ideje
ar	Egész szám, az autó bérlésének ára forint / nap
katgoriaId	A termék kategóriájának azonosítója, FK

berlesek

id	Egész szám, a bérlet azonosítója, PK
autoId	Egész szám, a kibérelt autó azonosítója, FK
mennyiseg	Egész szám, napban kifejezve a bérlet időtartama
biztositas	Decimális érték, biztosítási díj forintban
datum	Dátum/idő, a rendelés dátuma

Az elsődleges kulcsokat PK-val, az idegenkulcsokat FK-val jelöltük.



- Hozzon létre a lokális SQL szerveren oldtimer néven adatbázist! Az adatbázis alapértelmezett rendezési sorrendje a magyar szabályok szerinti legyen! Állítsa be az UTF8 kódolást alapértelmezettnek az adatbázis létrehozásánál!
- A forras mappában található adatbazis.sql állomány tartalmazza a táblákat létrehozó és az adatokat a táblákba beszűrő SQL parancsokat. Futtassa az adatbazis.sql parancsfájlt az oldtimer adatbázisban!
- Az autok tábla katgoriaId mezőjében 4-es érték jelzi, ha az adott autó sportautó típusú. Írassuk ki, hány sportautó szerepel a flottában! A számított mező neve sportautok-szama legyen! (16. feladat:)
- Listázza ki az összes "Limuzin" kategóriába tartozó autó nevét és színét! Rendezze őket névsorba! A lekérdezés során a kategória kijelölését "Limuzin" név felhasználásával tegye! (17. feladat:)

A feladat a következő oldalon folytatódik

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

18. Adjon hozzá az autókhoz egy új gépkocsit: **(18. feladat:)**

- rendszám: OT44-01
- szín: Fekete-piros
- categoriaId: 3
- nev: GMC Vandura Szupercsapat kiadás
- evjarat: 1983
- ar: 18000

19. Írassa ki, mely autótípusok voltak a legnépszerűbbek eddig, azaz mely típusokat bérelték ki összesen a leghosszabb időre (az azonos nevű autók azonos típusúnak számítanak). Írassa ki a legnépszerűbb 5 típus nevét, illetve a számított mezőt, melynek neve mennyiség legyen! **(19. feladat:)**

20. Határozza meg, mennyi volt az adott időszakban az autókölcsönző összes bevétele és mely biztosítás volt százalékos arányban a legmagasabb! **(20. feladat)**

Egy bérlesemből származó jövedelem képlete:

$$\text{Bérles_bevétel} = (\text{napok száma} * \text{autó napi díja}) + \text{bérlet biztosítás}$$

Egy bérlet esetén a befolyt összegben a biztosítás aránya százalékban kifejezve:

$$\text{Biztosítás_aránya} = (\text{Adott bérlet biztosítása} / \text{Bérlet_bevétel}) * 100$$

Tehát pl. Ha egy autó ára 5000 HUF / nap, 3 napra bérelték ki és 2500 forint volt a biztosítás:

$$\text{Bérlet_bevétel} = (3 * 5000) + 2500 = 17500$$

$$\text{Biztosítás_aránya} = (2500 / 17500) * 100 = 14(\%)$$

A számított mezők neve összes-bevétel és max-biztositas-arany legyen.

Az elkészült munka beadása:

Munkájának befejezése után másolja az alábbi állományokat a vizsgakönyvtár 3_feladat\beadott mappájába!

- 3_feladat\oldtimer\web\index.html
- 3_feladat\oldtimer\web\oldtimer.css
- 3_feladat\oldtimer\web\oldtimer.js
- 3_feladat\oldtimer\lekerdezések\lekerdezések.sql

Munkája a „Weboldalak kódolása és adatbázis-kezelés” feladatnál csak a beadott mappában található állományok tartalma alapján lesz értékelve! Amennyiben más fájlokat is létrehozott vagy módosított, azok a javítás során nem lesznek figyelembe véve.

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

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

dátum

javító tanár

	pontszáma egész számra kerekítve	
	elért	programba beírt
Számítógépen megoldott gyakorlati feladatok		

dátum

dátum

javító tanár

jegyző