

ÉRETTSÉGI VIZSGA • 2023. október 20.

INFORMATIKAI ISMERETEK

KÖZÉPSZINTŰ GYAKORLATI VIZSGA

minden vizsgázó számára

2023. október 20. 8:00

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 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 esetleírását. (A rendszergazda nem segítheti a vizsgázót a dolgozat elkészítésében.)

Források:

2. feladat: https://www.ksh.hu/stadat_files/nep/hu/nep0052.html
(utolsó letöltés dátuma: 2023. 07. 25.)

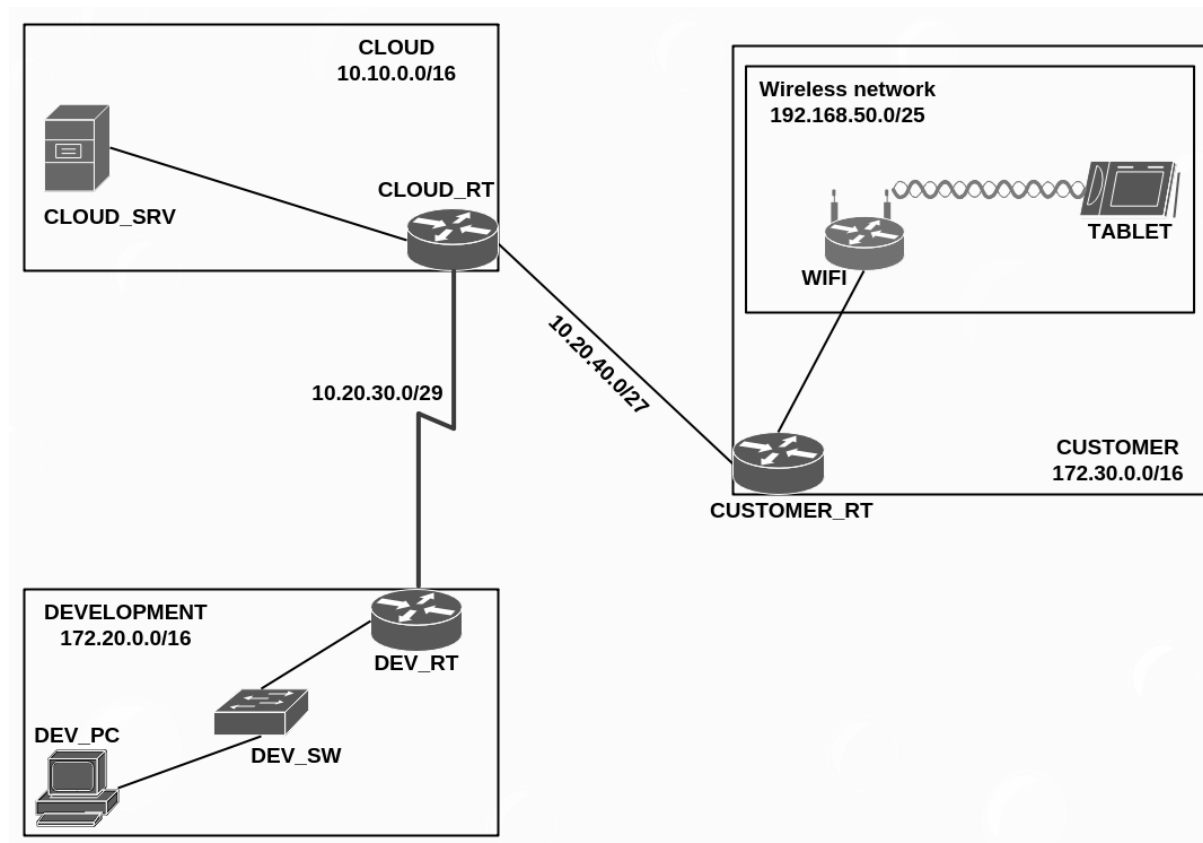
3. feladat:
<https://www.worldcubeassociation.org/competitions/SzokolyaOpen2023>
https://hu.wikipedia.org/wiki/Hanoi_tornyai
<https://hu.wikipedia.org/wiki/Tangram>
<https://hu.wikipedia.org/wiki/Rubik-kocka>
(utolsó letöltés dátuma: 2023.07.25.)

1. feladat

40 pont

Cloud

A felhőalapú számítástechnika (Cloud computing) mint szolgáltatás napjainkban már a legtöbb vállalat informatikai infrastruktúrájának szerves részét képezi. Az ügyfelek a szolgáltatások széles palettájáról tudnak választani az igényeik szerint, egészen a speciális szoftverek ideiglenes használatától a komplett virtuális hálózatokig. Egy képzeletbeli vállalat (CUSTOMER) szeretne igénybe venni felhőszolgáltatást a legújabb projektjének kivitelezéséhez. A szolgáltatónak két telephelye van, az egyik telephelyen a felhőbe kapcsolt szerverek találhatók (CLOUD), a másik telephelyen a felhőszolgáltató munkatársai dolgoznak, ahol többek közt a speciális felhőalapú alkalmazások fejlesztése is történik (DEVELOPMENT). Az alábbi képen az ügyfél és a szolgáltató hálózatának topológiai ábrája látható. Az Ön feladata, hogy az alábbi követelményeknek megfelelően bekonfigurálja a topológián található eszközöket. Munkáját Cloud néven mentse a használt szimulációs program alapértelmezett formátumában!



A feladat a következő oldalon folytatódik

Eszköz	IP-cím	Alhálózati maszk	Alapértelmezett átjáró	Név
CLOUD_RT	10.20.30.6	255.255.255.248	-	CLOUD_RT
	10.20.40.1	255.255.255.224	-	
	10.10.255.254	255.255.0.0	-	
DEV_RT	10.20.30.1	255.255.255.248	-	DEV_RT
	172.20.255.254	255.255.0.0	-	
DEV_SW	172.20.255.250	255.255.0.0	172.20.255.254	DEV_SW
CUSTOMER_RT	10.20.40.30	255.255.255.224	-	CUSTOMER_RT
	172.30.255.254	255.255.0.0	-	
CLOUD_SRV	10.10.0.1	255.255.0.0	10.10.255.254	-
DEV_PC	DHCP kliens		172.20.255.254	-
WIFI	172.30.100.100	255.255.0.0	172.30.255.254	-
	192.168.50.126	255.255.255.128	-	-
TABLET	DHCP kliens		192.168.50.126	-

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 kapcsoló legalább nyolc portos legyen!
 - A vezeték nélküli hálózathoz használja a szimulációs programban elérhető vezeték nélküli SOHO forgalomirányítók bármelyikét!
 - 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! A WIFI vezeték nélküli forgalomirányító az internet porton keresztül csatlakozzon a CUSTOMER_RT forgalomirányító tetszőleges Ethernet portjához!
- A fenti táblázat alapján ossza ki a megadott eszközöknek a megfelelő IP paramétereket! A forgalomirányítók (CLOUD_RT, DEV_RT, CUSTOMER_RT) valamint a kapcsolón (DEV_SW) állítsa be a táblázatban található nevet! DNS szervernek a 44.55.66.77 IP-cím legyen beállítva a CLOUD_SRV kliensen és a WIFI vezeték nélküli forgalomirányító internet portján is!
- A fejlesztők hálózatában (DEVELOPMENT) tegye lehetővé az alapértelmezett átjáró (forgalomirányító) távoli elérését (telnet)! Ehhez hozzon létre a DEV_RT forgalomirányítón egy **rgazda** nevű felhasználót **secret123*** jelszóval! Tegye meg a szükséges beállításokat ahhoz, hogy az imént létrehozott felhasználó a forgalomirányító első 5 virtuális terminálján be tudjon jelentkezni!
- A DEV_SW kapcsoló rendszergazdai módját (enable) védje az illetéktelen felhasználóktól! Ehhez állítsa be a **secret123*** jelszót!

6. Ahhoz, hogy az ügyfél (CUSTOMER) el tudja érni a számára szükséges szolgáltatást (CLOUD), irányító protokoll szükséges. Ehhez definiáljon OSPFv2-t mind a CLOUD_RT, mind a CUSTOMER_RT forgalomirányítók az alábbiak szerint:
- Mindkét forgalomirányítón használja a **40-es** folyamatazonosítót! Az OSPF terület legyen **100-as**!
 - A CUSTOMER_RT forgalomirányítón az összes közvetlenül csatlakozó hálózat kerüljön hirdetésre!
 - A CLOUD_RT forgalomirányítón csak a belső hálózat (CLOUD) és a CUSTOMER_RT forgalomirányítóval közös hálózat kerüljön hirdetésre!
 - A biztonságos működés érdekében tegye meg azt a beállítást, amely után a CUSTOMER_RT nem fog irányítási információkat küldeni a hozzá közvetlenül csatlakozó LAN-ba!
7. A fejlesztők hálózatában (DEVELOPMENT) található gépek számára a DEV_RT forgalomirányító osztja ki a szükséges IP paramétereket. Állítson be DHCP szolgáltatást az DEV_RT forgalomirányítón az alábbiak szerint:
- A DEVELOPMENT hálózat hostok számára kiosztható címtartományának utolsó 20 IP-címe statikusan lesz felhasználva, így az ne kerüljön kiosztásra! Ehhez tegye meg a megfelelő beállítást!
 - Definiáljon egy DHCP tartományt (a név tetszőleges) a DEVELOPMENT hálózati címét felhasználva!
 - A kliensek a DHCP szolgáltatással kapják meg a megfelelő alapértelmezett átjárót is! Ehhez is tegye meg a megfelelő beállításokat!
 - DNS szervernek minden kliens a 44.55.66.77 címet kapja meg!
8. Tesztelje a DHCP szolgáltatást! Ellenőrizze, hogy a DEV_PC kliens megkapta-e a megfelelő IP paramétereket! Ha hibát tapasztal, hárítsa azt el!

A feladat a következő oldalon folytatódik

9. Az ügyfél hálózatában (CUSTOMER) a dolgozók a munkájukat vezeték nélküli klienseken (TABLET) végzik. Állítsa be a vezeték nélküli hálózatot (Wireless network) az alábbi szempontok szerint!

Az alábbi beállításokat a WIFI vezeték nélküli forgalomirányítón végezze el:

- A belső hálózat a 192.168.50.0/25 címtartományt használja. Ezt figyelembe véve állítsa be a WIFI eszközt, mint a belső oldali alapértelmezett átjáróját, a címtartomány utolsó kiosztható címét felhasználva!
 - A vezeték nélküli hálózat kliensei az IP paramétereket a WIFI eszköztől kapják meg. Ehhez definiáljon DHCP szolgáltatást az említett eszközön az alábbiak szerint:
 - Az első cím a kiosztható címtartomány első címe legyen és maximum 100 kliens számára tudjon egyidőben szolgáltatást nyújtani!
 - DNS szervernek a 44.55.66.77 IP-cím legyen kiosztva!
 - A vezeték nélküli kliensek csatlakozásához az **ALKALMAZOTTAK** SSID legyen beállítva. Az esetleges támadások kivédése ellen az imént említett SSID ne kerüljön direktben szórásra a vállalat területén!
 - Az ügyfél dolgozói a munkájuk során biztonságtechnikailag érzékeny adatokkal dolgoznak így a vezeték nélküli hálózati forgalom biztonságosan kell, hogy megvalósuljon. Ehhez tegye meg az alábbi beállításokat:
 - A vezeték nélküli protokollnak a **WPA2** legyen alkalmazva!
 - A hitelesítéshez használt szöveg legyen: **alkalmazott123***
10. Tesztelje a vezeték nélküli hálózatot a vezeték nélküli klienssel (TABLET)! Tegye meg a megfelelő beállításokat ahhoz, hogy a csatlakozás létrejöjjön! A TABLET eszközön állítsa be az automatikus IP-cím kérést! Ellenőrizze, hogy a megfelelő IP paramétereket kapta-e meg a kliens! Ha hibát tapasztal, hárítsa azt el!
11. A szolgáltató munkatársainak (DEVELOPMENT) a szoftverfejlesztési munkájuk során folyamatosan el tudják érni a felhőbe kötött szervereket (CLOUD_SRV). Ennek biztosítása érdekében végezze el az alábbi beállításokat:
- A DEV_RT forgalomirányítón állítson be port alapú hálózati címfordítást (PAT)! A címfordítást a teljes belső hálózatra engedélyezze és csomagok a forgalomirányító külső (serial) interfészének IP-címét kapják meg (túlterhelés)!
 - A CLOUD_RT forgalomirányítón állítson be statikus NAT-ot! Ehhez párosítsa össze a CLOUD_SRV szerver belső helyi címét a 10.20.30.5 címmel!
 - Tesztelje le a kapcsolatot, a DEV_PC -ről el kell tudja érni a CLOUD_SRV szerver a 10.20.30.5 címmel (ping). Ha hibát tapasztal, hárítsa azt el!

A hálózat működésének tesztelése:

- Az DEV_PC PC-ről elérhető a DEV_RT forgalomirányító TELNET kapcsolattal és pinggel egyaránt.
- A DEV_PC PC-ről elérhető a CLOUD_SRV pinggel.
- Az TABLET-ről elérhető a CLOUD_SRV pinggel a belső hálózati címét használva (10.10.0.1).

2. feladat

40 pont

Országok népessége

A világ népessége rohamosan növekszik, az ENSZ adatai szerint 2022 novemberében hivatalosan is átlépte a nyolcmilliárdot. A Központi Statisztikai Hivatal (KSH) oldalán, az egyes országok népességének 2021-es adatait tekinthetjük át. Az oldalról származó adatok felhasználásával kell feladatokat megoldania. A megoldás során vegye figyelembe a következőket:

- *A képernyőre írást igénylő feladatok eredményének megjelenítése előtt írja a képernyőre a feladat sorszámát (például: 4. 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 igyekezzen betartani az adott programozási nyelv konvencióit!*
- *A program megírásakor az 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 forrás állomány UTF-8 és Win-1250 kódolással is rendelkezésre áll, az állomány nevében utalva a kódolásra. A megoldáshoz csak az egyiket, az Ön számára megfelelőbbet kell felhasználnia. Az adatok-utf8.txt és az adatok-1250.txt állomány első sora az adatoszlopok tartalmát írja le. A második sortól az adatsorok következnek. Az adatok elválasztására pontosvesszőt használtunk.

A forrásállomány néhány sora:

```
Ország;Terület;Népesség;Főváros;Főváros népessége  
Görögország;132000;1064g;Athén;3154  
Románia;238400;1912g;Bukarest;1812  
Ausztria;83900;8960000;Bécs;1915  
Lengyelország;312700;3775g;Varsó;1776  
Belgium;30500;11590000;Brüsszel;2065  
Bulgária;110400;6880000;Szófia;1277
```

Az egyes adatsorokban rendre a következő adatokat tároltuk:

- Ország: az ország neve, például: Ausztria
- Terület: az ország területe km²-ben megadva, például: 83900
- Népesség: az ország népessége főben megadva, például: 8960000
- Főváros: az ország fővárosának neve, például: Bécs
- Főváros népessége: a népesség ezer főben megadva, például: 1915

A feladat a következő oldalon folytatódik

1. Készítsen **konzolos alkalmazást** *nepesség* néven a következő feladatok megoldására!
2. Tegye elérhetővé programjában az ön által használt programozási nyelvnek megfelelő, a forrásoknál megadott `java.txt`, vagy az `csharp.txt` állományokban definiált `Orszag` osztályt! Ennek felhasználásával olvassa be megfelelő adatszerkezetbe a forrásállomány tartalmát! Az osztály definíciójában két konstruktort talál. Az egyiket használja fel, amit tetszőlegesen átalakíthat a megoldáshoz, a másik kódját törölheti.
Az ország népszám adatainál néhány helyen a számadat végén egy „g” betű szerepel. (Az állomány elején, három helyen talál ilyen sort.) Ez azt jelenti, hogy az így szereplő számok tízezer főben megadott becslött értékek. Az ilyen számadatokat szoroznia kell tízezerrel és az így átalakított numerikus értékeket kell tárolnia.
Ha nem tudja elvégezni programjával az átalakítást, akkor a forrásállományban cserélje a „g” karaktereket a „0000” karaktersorozatra!

Az Orszag osztályt tetszőlegesen bővítheti, átalakíthatja a megoldások érdekében.

3. Hány ország adatait tartalmazza a forrás? A beolvasott adatok alapján az eredményt írja ki a képernyőre!
4. Készítsen jellemzőt/metódust az osztályban a népsűrűség meghatározására! A népsűrűséget a matematikai szabályok betartásával egész főre kerekítve kell meghatározni. (A népsűrűség az 1 négyzetkilométeren lakók száma.) Írja ki Kína népsűrűségét a képernyőre!
5. Határozza meg és írja a képernyőre, hogy Kínában hány fővel többen éltek a vizsgált időpontban, mint Indiában! Az eredményt írja a képernyőre!
6. Az adatokból egyértelműen látszik, hogy Kína és India a két legnépesebb ország. Írja a képernyőre, hogy melyik ország a harmadik helyezett a lakosság számát tekintve!
7. Néhány ország lakossága a fővárosban koncentrálódik. Sorolja fel azokat az országokat, a főváros nevével együtt, ahol a fővárosban lakók aránya meghaladja a 30%-ot! A megoldáshoz használja ki azt, hogy ez egy adott ország tulajdonsága, azaz bővítse az `Orszag` osztályt egy logikai értéket visszaadó függvényvel, amely csak akkor ad vissza igaz értéket, ha a fővárosban lakók aránya meghaladja a 30%-ot! A kiírásnál a behúzáshoz használjon tabulátort a minta szerint!

8. Készítsen **grafikus alkalmazást** az állomány bővítésére! A grafikus alkalmazás projektjét **adatbovites** néven hozza létre! A konzolos alkalmazásból tetszőlegesen emelhet át kódrészleteket.
- a. Alakítsa ki a felhasználói felületet az alábbi minta alapján! Az elrendezésre, a beviteli mezők igazítására és a feliratokra figyeljen, a színeket és a méreteket tetszőlegesen választhatja meg. A beviteli mezők alapértékei a mintán látható értékek legyenek!

Új ország felvétele

Ország: Chile

Terület: 756950

Népesség: 19458000

Főváros: Santiago

Főváros lakossága: 6257516

Kérem adja meg az adatokat!

Mentés

Kilépés

- b. A Kilépés gombra az alkalmazásablak záródjon be!
- c. A Mentés gombra kattintva először ellenőrizze a népesség és a főváros lakossága értékeit! A főváros lakossága nem lehet több, mint az ország lakossága. Ha ilyen adatot adott meg a felhasználó, a minta szerint értesítsük róla, és a főváros lakosságának értékét állítsuk be az ország népességének nagyságára (Ilyenkor nem kell mentenie). Ha az adatok megfelelnek a feltételeknek, bővítse a beírt adatokkal az `ujadat.txt` állományt! Elválasztó karakterként pontosvessző karaktert használjon! Ha az állomány nem létezik, akkor hozza létre fejléc sor nélkül! A sikeres mentésről a minta szerint tájékoztassa a felhasználót! A használandó feliratok: „A mentés sikeres!” és „A főváros lakossága nem lehet több a népességnél!”

Minta a konzolos alkalmazáshoz:

```
4. feladat
A beolvasott országok száma 49.

5. feladat
Kína népsűrűsége: 147 fő/km^2.

6. feladat
Kínában a lakosság 18950000 fővel volt több.

7. feladat
A harmadik legnépesebb ország: Egyesült Államok, a lakosság 331890000 fő.

8. feladat - A következő országok lakosságának több mint 30%-a a fővárosban lakik:
Észak-Macedónia (Szkopje)
Lettország (Riga)
Észtország (Tallinn)
Málta (Valletta)
Izland (Reykjavík)
```

A feladat a következő oldalon folytatódik

Minták a grafikus alkalmazáshoz:

Hibás lakosság adatok megadása

Sikeres mentés

```
Chile;756950;19458000;Santiago;6257516
Togo;56785;7797694;Lomé;837437
Chile;756950;150;Santiago;150
Marokkó;446550;37076584;Rabat;572717
```

Minta az ujadat.txt állományhoz

3. feladat

40 pont

Egyszemélyes logikai játékok

A következő feladatban egy weboldalt kell készítenie játékeszközök bemutatására a feladatleírás és a minta szerint, valamint egy verseny 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 `jatekok.html` állományt és szerkessze annak tartalmát az alábbiak szerint:

1. A weboldal karakterkódolása UTF-8, nyelve magyar, a böngésző címsorában megjelenő cím „Egyszemélyes logikai játékok” legyen!
2. A weboldal fejrészében helyezzen el hivatkozásokat a `jatekok.css` stíluslapra, valamint a `rekordok.js` állományra a meglévő hivatkozások után!
3. A dokumentum fejrész elemét formázza a meg a `jumbotron` osztálykijelölővel!
4. A „Hanoi tornyai” játékhoz tartozó blokkban végezze el a következő módosításokat:
 - a. Az első két bekezdés tartalmát vonja össze egy bekezdésbe a minta szerint!
 - b. A jobb oldali hasábjában helyezze el a `hanoi.png` képfájlt, és formázza az `img-fluid` osztálykijelölővel! Ha a kép valamiért nem jeleníthető meg vagy ha a felhasználó a kép fölé viszi az egérkurzort, mindkét esetben a „Hanoi tornyai” szöveget jelenítse meg!
5. A „Tangram” játékhoz tartozó blokkban végezze el a következő módosításokat:
 - a. A blokk hasábjainak 6:6 arányú szélességét javítsa 4:8 arányúra!
 - b. A hasábjok tartalmát cserélje fel: a bal oldaliban legyen a kép, a jobb oldaliban pedig a szöveges tartalom!
6. A „Rubik-kocka” játékhoz tartozó blokkban végezze el a következő módosításokat:
 - a. A `fieldset` HTML tag első bekezdését alakítsa át egy `legend` elemmé!
 - b. Állítsa be a megfelelő attribútumokkal, hogy a „*csúszka*” típusú beviteli elem értéke ne lehessen 7-nél nagyobb, és hogy az oldal betöltésekor a 3-as értéket vegye fel!
7. A weboldal lábrészében a forrást megjelölő bekezdés „Wikipédia.hu” szövegrészletére készítsen alapértelmezetten új lapon megnyíló hivatkozást! A hivatkozás URL címét a bekezdés végén találja megjegyzésként a forráskódban!
8. Nyissa meg a `jatekok.css` állományt, majd módosítsa a következők szerint:
 - a. A bekezdések szövege legyen sorkizárt igazítású!
 - b. A fejrészben elhelyezett első szintű címsor szövegszíne fehér legyen!
 - c. A `kmeret`, `kido`, és `knev` azonosítókhoz állítson be 150%-os betűméretet!
9. Nyissa meg a `rekordok.js` állományt, majd módosítsa a `frissit()` függvényt a következők szerint:
 - a. Az időérték kiírásnál a szám mögött jelenítse meg a „mp” mértékegységet jelölő szót!
 - b. Cserélje le a `rubik` azonosítóval ellátott kép forrását a `kepfajl` változó értékére!

A feladat a következő oldalon folytatódik

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

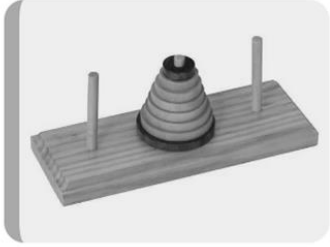


Egyszemélyes logikai játékok

Hanoi tornyai

A Hanoi tornyai matematikai játék, amihez a hasonló nevű matematikai feladvány kapcsolódik. Ez úgy is ismert, mint Brahma tornyai, vagy világvége feladvány. A játék szabályai szerint az első rúdról az utolsóra kell átrakni a korongokat úgy, hogy minden lépésben egy korongot lehet áttenni, nagyobb korong nem tehető kisebb korongra, és ehhez összesen három rúd áll rendelkezésre.


A játékot 1883-ban Édouard Lucas francia matematikus találta fel. Az ötletet egy legendából vette, ami szerint a világ megteremtésekor egy 64 korongból álló tornyot kezdtek átmozgatni Brahma szerzetesei. A szabályok azonosak voltak a ma ismert hanoi torony szabályaival. A legenda szerint, amikor a szerzetesek végeznek majd a korongok áttűztetésével a harmadik rúdra, a kolostor összeomlik, és a világunk megszűnik létezni.



Tangram

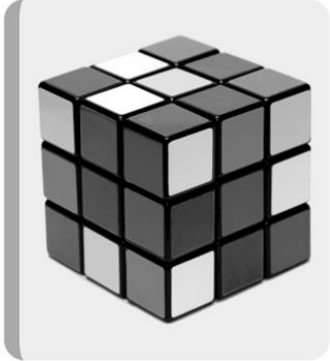
A tangram egy kínai kirakós játék, amely feltehetően az i. e. 8. és 4. század között keletkezett. A türelemjáték nyugati neve a kínai Tang-dinasztiához köthető műszó. A legenda szerint egy szerzetes utazni küldte a tanítványát, hogy a világ szépségének lényegét egy táblára fesse. A tábla azonban eltört, és a tanítvány hiába próbálta meg négyzet alakban összerakni. Próbálkozásai közben sok szép minta keletkezett. Végül a tanítvány megértette, hogy nem kell beutaznia a világot, mivel a világ szépségét és sokféleségét meg lehet találni a szét tört tábla darabjaiban.

A játék hét egyszerű mértani alakzatot tartalmaz: egy négyzetet, egy paralelogrammát, a többi darab derékszögű egyenlő szárú háromszög, amely közül kettő nagy, egy közepes, és két kicsi darab. Ezekből számtalan figura kirakható, amelyek árnyékszerűen emberekre, állatokra, tárgyra emlékeztetnek. Hagyományosan az összes darabot fel kell használni, és a darabok nem fedhetik át egymást.



Rubik-kocka

A Rubik-kocka háromdimenziós mechanikus logikai játék, amit 1974-ben talált fel ifjabb Rubik Ernő. A Rubik-kocka eredeti neve a magyar kereskedelemben bűvös kocka volt. A szabadalmi leírásban a feltaláló térbeli logikai játékként nevezte meg a mechanikus, egyéni logikai játékot, amelynek célja, hogy egy előzetesen összekevert kockából forgatással visszaállítsuk az eredeti, rendezett színösszeállítást, vagyis minden oldalon azonos színű lapocskák legyenek.




Magyar rekordok:


Kockaméret: 3 x 3

Magyar rekord: 5,56 mp

Magyar Rekord: Slezák Gábor



A weboldal szövegének forrásai: Wikipédia.hu



A második feladatrészben a „Szokolya Open 2023” elnevezésű Rubik-kocka verseny szűrt eredményeit tartalmazó adatbázissal kell dolgoznia.

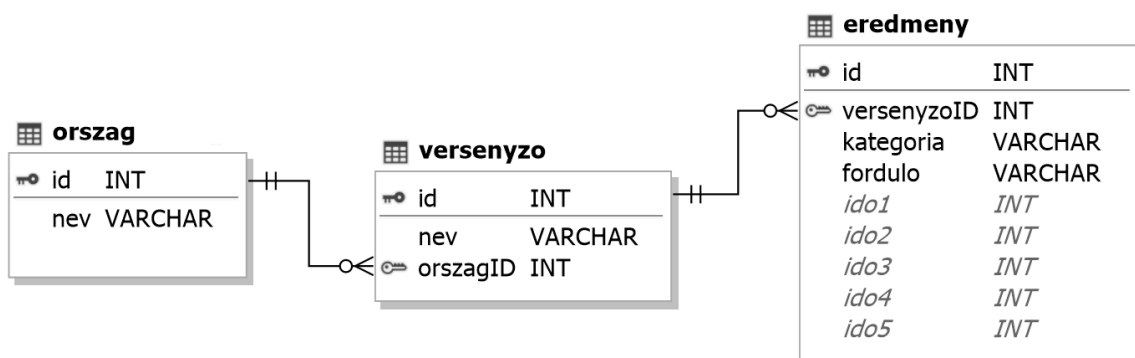
Rendelkezésre állnak a versenyzők és országaik nevei, valamint az elért eredményeik: kategóriánként, milyen időeredményeket értek el az egyes fordulókban. A nagy indulószámmal rendelkező kategóriákban két forduló után tartották a döntőt, a kisebb létszámú kategóriákban egyből a döntő került megrendezésre.

Az adatbázis a következő táblákat tartalmazza:

ország	
id	Egész szám, az ország azonosítója, PK
nev	Szöveg, az ország neve (<i>egyedi érték</i>)
versenyző	
id	Egész szám, a versenyző azonosítója, PK
nev	Szöveg, a versenyző neve
országId	Egész szám, a versenyző országának azonosítója, FK
eredmény	
id	Egész szám, az eredmény azonosítója, PK
versenyzőID	Egész szám, a versenyző azonosítója, FK
kategoria	Szöveg, a kategória megnevezése, például: „2x2”, „3x3”...
fordulo	Szöveg, a forduló jelölése: „f1” első forduló, „f2” második forduló és „d” döntő esetén
ido1	Egész szám, leggyorsabb kirakás ideje század másodpercben
ido2	Egész szám, a második leggyorsabb kirakás ideje század mp-ben Értéke NULL, ha nem volt (időre) második kirakás
ido3	Egész szám, a harmadik leggyorsabb kirakás ideje század mp-ben Értéke NULL, ha nem volt (időre) harmadik kirakás
ido4	Egész szám, a negyedik leggyorsabb kirakás ideje század mp-ben Értéke NULL, ha nem volt (időre) negyedik kirakás
ido5	Egész szám, az ötödik kirakás ideje század másodpercben Értéke NULL, ha nem volt (időre) ötödik kirakás

Az elsődleges kulcsokat PK-val, az idegenkulcsokat 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.**

A feladat a következő oldalon folytatódik

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 kockaverseny 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 kockaverseny adatbázisban!
12. Törölje az országok táblájából „Svájc” rekordját! A törlést az ország azonosítójának meghatározása nélkül végezze el! **(12. feladat:)**
13. Határozza meg, hogy összesen hány versenyző szerepelt a versenyen! Állítsa be a számított mező címkéjét a minta szerint! **(13. feladat:)**

versenyzoszam
58

14. Készítsen lekérdezést, mely megjeleníti kategóriánként a versenyen elért legjobb egyéni időt másodpercben kifejezve! A listát ne rendezze! A számított mezőben a megjelenített tizedesjegyek számát, és a tizedes elválasztót (pont vagy vessző) hagyja alapértelmezetten! Állítsa be a számított mező címkéjét a minta szerint! **(14. feladat:)**

kategoria	leggyorsabb
...	
3x3	5, 6
...	

15. Készítsen lekérdezést, mely megjeleníti azoknak a versenyzőknek az ábécébe rendezett névsorát, eredményének kategóriáját és fordulóját, akik az adott fordulóban pontosan négy alkalommal rakták ki időre a kockát! **(15. feladat:)**

nev	kategoria	fordulo
Ádám Szabó	3x3	f2
Alexey Tsvetkov	2x2	f1
...		

16. Készítsen lekérdezést, amely megjeleníti, hogy a klasszikus 3x3-as kategóriában mely országok képviselték magukat a versenyen! Minden országnév csak egyszer jelenjen meg a listában! A listát ne rendezze! **(16. feladat:)**

orszag
...
Magyarország
...

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

dátum

javító tanár

Számítógépen megoldott gyakorlati feladatok	pontszáma egész számra kerekítve	
	elért	programba beírt

dátum

dátum

javító tanár

jegyző