

## Úszóverseny

Adatbázis létrehozása.		7
Létezik az <i>úszás</i> nevű adatbázis, benne az <i>időeredmények</i> , <i>egyébadat</i> és <i>település</i> nevű táblák	1	
A mezők megfelelő típusúak, és méretűek	1	
Megfelelő mezőket állította be kulcsnak, nem vett fel újabb mezőket a táblákba	1	
Az időeredmények tábla az idő mező szerint indexelt (lehet azonos)	1	
Az idő mezőt kötelezőre állította az időeredmények táblában	1	
Az egyébadatok tábla születésiidő mezőjére érvényességi szabályt állított be, amely szerint csak az 1910.01.01-2009.07.18 intervallum dátumai írhatók be	1	
Az egyébadatok táblában a "nem" mezőben található N és F értékek egy listából választhatók ki	1	
Minden lekérdezésre vonatkozik		2
Minden létező lekérdezésben pontosan a kívánt mezők jelennek meg	1	
A megadott néven mentette a lekérdezéseket. A pont akkor jár, ha legalább 4 lekérdezést készített	1	
3férfiak lekérdezés.		2
Megfelelő a táblák összekapcsolása	1	
Szűrés a "nem" szerint	1	
Például: SELECT időeredmények.rajt, időeredmények.idő, egyébadat.születésiidő FROM időeredmények, egyébadat WHERE (egyébadat.nem="F") AND (egyébadat.rajt=időeredmények.rajt);		
4korcsoportdarab lekérdezés		2
Csoportosítás korcsoport szerint	1	
Helyes szűrés a versenyzők számára	1	
Például: SELECT korcsoport, Count(*) AS darab FROM egyébadat GROUP BY korcsoport HAVING Count(egyébadat.rajt)>=200;		
5települések lekérdezés		3
Megfelelő a táblák kapcsolata	1	
Település neve szerint csökkenően rendez	1	
Minden településnév csak egyszer jelenik meg	1	
Például: SELECT DISTINCT település.település FROM település, egyébadat WHERE egyébadat.lakhely=település.irsz ORDER BY település.település DESC;		
6fővárosi lekérdezés		3

Az egyébadat táblába létrehozott egy mezőt fővárosi néven és logikaira állította a típusát	1	
Megfelelő szűrés a fővárosra	1	
Igazra állította a logikai mezőt	1	
Például: UPDATE egyébadat SET fővárosi = Yes WHERE lakhely Like "1*";		
7születésiév lekérdezés		2
Paraméteres lekérdezést készített	1	
A születési dátumból az évre szűr	1	
Például: SELECT Year([születésiidő]) AS Év, egyébadat.* FROM egyébadat WHERE Year([születésiidő])=[Mi a születési éve];		
8rajtváros lekérdezés		2
Kiválasztja az 7043-as rajtszámhoz tartozó település nevét	1	
A lekérdezésbe helyesen épül be az al-lekérdezés vagy a segéd tábla.	1	
Például: SELECT egyébadat.rajt, egyébadat.korcsoport, település.település FROM egyébadat, település WHERE település.település=(SELECT település.település FROM egyébadat, település WHERE egyébadat.rajt= 7043 AND egyébadat.lakhely=település.irszt) AND (egyébadat.lakhely=település.irszt); vagy SELECT egyébadat.rajt, egyébadat.korcsoport, [8rajtvárossegéd.település] AS helységnev FROM egyébadat, település, 8rajtvárossegéd WHERE [8rajtvárossegéd].[település]=település.település AND egyébadat.lakhely=település.irszt; ahol a 8rajtvárossegéd lekérdezés SELECT település.település FROM egyébadat, település WHERE egyébadat.rajt= 7043 And egyébadat.lakhely=település.irszt;		
9FGhölgyek lekérdezés		3
Táblakészítő lekérdezést készített	1	
Helyes a korcsoporra és nemre való szűrés, helyes a szűrőfeltételek kapcsolata	1	
Megfelelő a táblák kapcsolata	1	
Például: SELECT időeredmények.rajt, időeredmények.idő, egyébadat.korcsoport INTO 9FGhölgyektábla FROM időeredmények, egyébadat WHERE időeredmények.rajt=egyébadat.rajt AND (egyébadat.korcsoport="F" Or egyébadat.korcsoport="G") AND ((egyébadat.[nem])="N") ORDER BY időeredmények.idő DESC;		
10összesítés jelentés		4

Csak a megfelelő tábla, megfelelő mezői szerepelnek a jelentésben	1	
Csoportosítás születési idő éve alapján, azon belül nemek szerint	1	
A lista rajtszám alapján növekvően rendezett	1	
Az évszám szerinti csoportok végén szerepel, hogy hány fő van a csoportban (szám és felirat is)	1	
Összesen		30

## Napló

Adatbázis létrehozása		5
Létezik a <i>napló</i> nevű adatbázis, benne <i>forgalom</i> és <i>szerver</i> nevű táblák	1	
A mezők megfelelő típusúak, és méretűek	1	
A <i>forgalom</i> táblába felvett <i>azonosító</i> néven egy mezőt, amely számláló típusú és elsődleges kulcs; A <i>szerver</i> táblában a <i>szerverkód</i> mezőt beállította kulcsnak; Egyéb mezőket nem vett fel a táblákba	1	
A <i>forgalom</i> táblában a <i>szerverkód</i> mezőt kötelezőre állította	1	
A <i>forgalom</i> táblában a <i>szerverkód</i> szerint indexelt (lehet azonos is)	1	
Minden lekérdezésre vonatkozik		2
Minden létező lekérdezésben pontosan a kívánt mezők jelennek meg	1	
A megadott néven mentette a lekérdezéseket. A pont akkor jár, ha legalább 4 lekérdezést készített	1	
3ügyfelek lekérdezés		2
Minden ügyfél IP cím csak egyszer jelenik meg.	1	
Szűr a 80-as portra	1	
Például: SELECT DISTINCT ügyfélip FROM forgalom WHERE szerverport=80; vagy SELECT ügyfélip FROM forgalom WHERE szerverport=80 GROUP BY ügyfélip;		
4hánykülső lekérdezés		3
A Count() függvényt megfelelően használja	1	
Helyesen szűr a külső címekre	1	
A "Külső ügyfél" felirat jelenik meg és teljesen látszik	1	
Például: SELECT Count(*) AS [Külső ügyfél] FROM forgalom WHERE ügyfélip Not Like "192*";		
5összesforgalom lekérdezés		2
Fogadott bájt és azon belül küldött bájt szerint növekvően rendezett	1	
Számított mezőben megjelenik a küldött és fogadott bájtok összege	1	
Például: SELECT azonosító, ügyfélip, fogadottbájt, küldöttbájt, fogadottbájt + küldöttbájt AS [Összes forgalom] FROM forgalom ORDER BY fogadottbájt, küldöttbájt;		

6husebesség lekérdezés		4
Helyesen szűr a hu domain alá bejegyzett szervernevekre	1	
Megfelelő a táblák kapcsolata	1	
Helyesen számította ki az átlagsebességet	1	
Az átlagsebesség egész számként jelenik meg	1	
Például: SELECT forgalom.ügyfélip, forgalom.idő, szerver.szervernév, ([fogadottbájt]+[küldöttbájt])/[feldolgozásiidő]*1000 AS [sebesség(Bps)] FROM forgalom, szerver WHERE (szerver.szervernév Like "*.hu") AND (forgalom.szerverkód=[szerver].[szerverkód]);		
7forgalom35 lekérdezés		2
Kiválasztja a 192.168.1.35 klienscímhez tartozó szerverkódokat	1	
A lekérdezésbe helyesen (az IN operátorral) épül be a segédtábla vagy az al-lekérdezés	1	
Például: SELECT forgalom.* FROM forgalom WHERE szerverkód In (SELECT szerverkód FROM forgalom WHERE ügyfélip="192.168.1.35"); vagy SELECT forgalom.* FROM forgalom, 7forgalom35segéd WHERE forgalom.szerverkód In ([7forgalom35segéd].[szerverkód]); ahol a 7forgalom35segéd: SELECT DISTINCT szerverkód FROM forgalom WHERE ügyfélip="192.168.1.35";		
8időfrissít lekérdezés		3
Frissítő lekérdezést készített	1	
Az idő mező értékét megnöveli egy órával	1	
A mező továbbra is hosszú idő formátumú	1	
Például: UPDATE forgalom SET idő = [idő]+#12/30/1899 1:0:0#;		
9összidő lekérdezés		2
Csoportosít ügyfélip szerint	1	
Csoportonként összidőt számol	1	
Például: SELECT ügyfélip, Sum(feldolgozásiidő) AS SumOffeldolgozásiidő FROM forgalom GROUP BY ügyfélip;		
10kliens jelentés		5
Csak a megfelelő tábla, megfelelő mezői szerepelnek a jelentésben	1	
Csoportosítás szervertip, azon belül idő perce szerint	1	
A lista idő szerint növekvően rendezett	1	

Minden percsorozat alatt szerepel az összesített küldött- és fogadott bájtok összege	1	
Minden adat és felirat teljesen olvasható; Az összesítésnél a megfelelő szövegű címke is szerepel	1	
Összesen		30

## Webstatisztika

Adatbázis létrehozása		6
Létezik az <i>webstatisztika</i> nevű adatbázis, benne a <i>látogatások</i> és <i>részletek</i> nevű táblák	1	
A mezők megfelelő típusúak, és méretűek	1	
Megfelelő mezőket állította be összetett kulcsnak, nem vett fel újabb mezőket a táblákba	1	
Minden lekérdezésben, jelentésben csak a szükséges mezők jelennek meg	1	
Minden lekérdezés, jelentés neve megfelelő	1	
Helyesek a mezőnevek a 2., 4. és 7. lekérdezésben, és helyes a jelentés címe is	1	
2órás lekérdezés		3
Helyesen szűr az időre; Mindig az aktuális időhöz képes előre és visszafelé 1-1 óra közti rekordok; A két határ már nincs benne	1	
Település és azon belül idő szerint növekvően rendez	1	
Egy számított mezőben összefűzte a böngésző és az operációs rendszer mezőket; Az operációs rendszer neve zárójelben van	1	
Például: SELECT település, idő, [böngésző] & "(" & [oprendszer] & ")" AS rendszer FROM részletek WHERE idő<Time()+#1:0:0# And idő>Time()-#1:0:0# ORDER BY település, idő;		
3opr böngésző lekérdezés		2
Operációs rendszer és azon belül böngésző szerint csoportosít	1	
Táblakészítő lekérdezést készített, amely opr böngésző néven készít új táblát a rekordokból	1	
Például: SELECT oprendszer, böngésző INTO oprböngésző FROM részletek GROUP BY oprendszer, böngésző;		
4megnézőránként lekérdezés		2
Az idő mezőbeli órák alapján csoportosít	1	
Megszámolja a Count függvénnyel a csoportba tartozó rekordokat	1	
Például: SELECT Hour([idő]) AS óra, Count(IPcím) AS [látogatók száma] FROM látogatások GROUP BY Hour([idő]);		
5nem magyar lekérdezés		2
Szűr a nem magyarországi települések nevére	1	
Helyesen kapcsolja össze a két táblát	1	
Például: SELECT kapcsolathely, település, látogatások.dátum, látogatások.idő FROM látogatások, részletek WHERE település Not Like "(HU)" AND látogatások.dátum=részletek.dátum		

AND látogatások.idő=részletek.idő;		
6szhböngésző lekérdezés.		3
Al-lekérdezést, segédlekérdezést készített, amelyben meghatározta a szigethalmiak által használt böngészők típusát; A település nevében a Szigethalom kezdetre szűr.	1	
A főlekérdezésben szűr az al-lekérdezés/segédlekérdezés által visszaadott böngészők típusára; Az allekérdezés/segédlekérdezés több rekordot ad vissza és a főlekérdezés mindegyikre szűr	1	
Minden kapcsolathely neve csak egyszer jelenik meg és mindkét lekérdezésben helyes a táblák kapcsolata	1	
Például: SELECT DISTINCT kapcsolathely FROM látogatások, részletek WHERE böngésző In (SELECT böngésző FROM látogatások, részletek where látogatások.idő = részletek.idő AND látogatások.Dátum = részletek.dátum and település Like "szigethalom*") AND látogatások.dátum=részletek.dátum AND látogatások.idő=részletek.idő;		
7különbözőip lekérdezés		3
Meghatározza az egy kapcsolathelybeli értékhez tartozó különböző IP címeket; Fontos, hogy találunk a naplóban egy kapcsolathelyhez tartozóan azonos és eltérő IP címeket is	1	
Helyesen adja meg az egy kapcsolathelybeli értékhez tartozó különböző IP címeket	1	
Csak a több IP címet használó kapcsolathelyhez tartozó rekordok jelennek meg	1	
Például: SELECT kapcsolathely, Count(kapcsolathely) AS [kapcsolatok száma] FROM KülönbözőIPB GROUP BY kapcsolathely HAVING Count(kapcsolathely)>1;  ahol KülönbözőIPB Például: SELECT kapcsolathely, IPcím FROM látogatások GROUP BY kapcsolathely, IPcím;		
8elsőidő lekérdezés		4
Rákérdez a dátumra és az időre	1	
Helyesen szűr dátumra és időre	1	
Helyesen rendez dátumra és időre	1	
Kiválasztja a megadott időponthoz legközelebbi következő bejegyzést	1	
Például: SELECT TOP 1 kapcsolathely, dátum, idő FROM látogatások WHERE (dátum>=[dátum?]) AND (látogatások.idő>[időpont?]) ORDER BY dátum, idő;		
9szolgáltatóváros jelentés		5
Több táblából veszi az adatokat. Lekérdezést készített a jelentéshez vagy összekapcsolta a táblákat	1	



Csoportosít település, azon belül kapcsolathely szerint	1	
Rendezett dátum és azon belül idő szerint	1	
A városok szerinti csoportok végén szerepel, hogy hány dátumbejegyzés van a csoportban	1	
Oldalszélességen elfér, minden adat olvasható az álló jelentésen. Település és kapcsolathely típusa piros betűszínnel, minden más karakter fekete. A település szerinti csoportláb háttere teljes szélességben sárga	1	
Összesen		30

## Tankönyvárusítás

Adatbázis létrehozása		9
Létezik az <i>tankönyvárusítás</i> nevű adatbázis, benne a <i>diák</i> , <i>tankönyv</i> és <i>kedvezmények</i> nevű táblák	1	
A mezők megfelelő típusúak, és méretűek	1	
A diák és kedvezmények nevű táblákban a megfelelő mezőt állította be kulcsnak; A tankönyv táblában a megfelelő mezőket állította be összetett kulcsnak; Egyik táblába sem vett fel újabb mezőket	1	
Minden lekérdezésben, jelentésben csak a szükséges mezők jelennek meg	1	
Minden lekérdezés, jelentés neve megfelelő	1	
Helyes a számított mezők neve az 5. és 7. lekérdezésben, és helyes a jelentés címe is	1	
Az évfolyamhoz csak 5, 6, 7, 8 számok adhatók meg. <i>Diák</i> és <i>tankönyv</i> táblában is	1	
Az osztály betűjele egy listából választható ki; A lista csak az <i>a</i> , <i>b</i> , <i>c</i> betűket tartalmazza; Más beírt betűt nem fogad el a tábla. <i>Diák</i> és <i>tankönyv</i> táblában is	1	
A <i>tankönyv</i> tábla <i>normatív</i> mezője és a <i>kedvezmények</i> tábla <i>H</i> , <i>P</i> , <i>T</i> és <i>GY</i> mezői jelölőnégyzetként jelennek meg	1	
3biológia lekérdezés		2
Helyesen szűr a biológia tantárgyra; A biológia szó bárhol szerepelhet a tankönyv nevében	1	
A tankönyv kódja szerint növekvően rendezett	1	
Például: SELECT cím, tkkód, évfolyam FROM tankönyv WHERE cím Like "*biológia*" ORDER BY tkkód;		
4tankönyvigény lekérdezés		1
Minden tankönyvigényt leadó osztály (szám és betű is) megjelenik, de csak egyszer	1	
Például: SELECT DISTINCT évfolyam, osztály FROM tankönyv;		
5kedvezményesár lekérdezés		3
A felhasználótól bekéri az évfolyamot és az osztály betűjelét	1	
Megadja a tankönyvcsomag árát	1	
A kedvezményes tankönyvcsomag árát adja meg	1	
Például: SELECT évfolyam, osztály, Sum(ár) AS [könyvcsomag kedvezményes ára] FROM tankönyv WHERE normatív=0 GROUP BY évfolyam, osztály HAVING évfolyam=[évfolyam?] AND osztály=[osztály?];		
6kedvezménytkap lekérdezés		3
Helyes a táblák kapcsolata	1	

Helyesen szűr a kedvezményekre; Bármely ok miatt kap kedvezményt egy diák, szerepeljen a listában	1	
Egy jogosultak nevű táblába kiírja az adatokat	1	
Például: SELECT név, diák.diákkód INTO jogosultak FROM diák, kedvezmények WHERE (H=-1 OR P=-1 OR T=-1 OR GY=-1) and diák.diákkód = kedvezmények.diákkód;  vagy SELECT név, diák.diákkód INTO jogosultak FROM diák INNER JOIN kedvezmények ON diák.diákkód = kedvezmények.diákkód WHERE H=Yes OR P=Yes OR T=Yes OR GY=Yes;		
7kedvezményeskönyv lekérdezés		4
Helyes a táblák kapcsolata	1	
Az évfolyam és osztály mezőket összefűzte	1	
Az összefűzött osztályazonosító és azon belül a diák neve szerint növekvően rendezett	1	
Helyesek a szűrőfeltételek és helyes a szűrőfeltételek kapcsolata	1	
Például: SELECT [diák.évfolyam] & [diák.osztály] AS [osztály jele], név, tkkód, cím, ár FROM diák, tankönyv, kedvezmények WHERE ((normatív=-1 AND H=-1) OR (normatív=-1 AND P=-1) OR (normatív=-1 AND T=-1) OR (normatív=-1 AND GY=-1)) and diák.diákkód=kedvezmények.diákkód and diák.évfolyam=tankönyv.évfolyam and diák.osztály=tankönyv.osztály ORDER BY [diák.évfolyam] & [diák.osztály], név;		
8osztálytárs lekérdezés		3
Meghatározza a 343888 azonosítójú diák osztályazonosítóját; Évfolyamot és osztályt is	1	
Helyes a táblák kapcsolata.	1	
Helyesen szűri ki a több jogcímen is támogatásra jogosultakat	1	
Például: SELECT diák.név FROM 8osztálytárssegéd, diák, kedvezmények WHERE diák.évfolyam=[8osztálytárssegéd.évfolyam] AND diák.osztály=[8osztálytárssegéd.osztály] AND ([H]+[P]+[T]+[GY])<-1 and diák.diákkód = kedvezmények.diákkód;  ahol a 8osztálytárssegéd: SELECT évfolyam, osztály FROM diák WHERE diákkód="343888";  vagy SELECT név FROM diák, kedvezmények		

WHERE diák.évfolyam=(SELECT évfolyam FROM diák WHERE diákkód="343888") AND diák.osztály=(SELECT osztály FROM diák WHERE diákkód="343888") AND ([H]+[P]+[T]+[GY])<-1 and diák.diákkód = kedvezmények.diákkód;		
9sokgyerek jelentés		5
A minta szerinti formában jelenik meg az osztály azonosítója; Szám és betűjel is	1	
A minta szerint az osztály azonosítója és azon belül a diák neve szerint csoportosított	1	
Csak három vagy többgyerekes kedvezményre jogosult diákok kedvezményes könyveit sorolja fel; Például lekérdezésen alapul a jelentés	1	
Osztályonként összesítette a tankönyvek árát	1	
Álló tájolású a jelentés szélességben minden mező elfér egy oldalon; Az összesítés felirata és összege 14 pontos piros betűkkel jelenik meg	1	
Összesen		30