

Adatbázis rendszerek

Tárgyfelelős: dr. Kovács László

Szak: mérnök informatikus mesterszak

Kód: GEIAL521M

hét	előadás	gyakorlat
1	Hierarchikus adatmodell, Az LDAP rendszer struktúrája, konfiguráció. LDAP adatkezelő műveletek, LDAP API nyelv használata	LDAP API programozása
2	Az XML adatmodell; XMLSchema nyelv, az eXistDB használat alapjai,	Az eXistDB használata, programozása, xQuery
3	Hierarchikus adatmodell; az xQuery nyelv elemei; xQuery használata az eXistDB rendszerben	Az eXistDB használata, programozása, xmlSchema (egyéni programozási feladat)
4	Objektumok perzisztens tárolása, Hibernate, HQL nyelv, JPA architektúra,	Hybernate, HQL programozása
5.	Objektumok perzisztens tárolása, myBatis architektúra. Az Oracle ORDBMS architektúra, UDT elemek létrehozatala	JPA programozása
6	ORDBMS-SQL elemei, a procedurális komponensnek elkészítése	Oracle ORDBMS kezelése (egyéni programozási feladat)
7	NoSQL rendszerek áttekintése. A dokumentum adatbázisok, MongoDB működési modell	MongoDB kezelése, API programozása
8	MongoDB adatbázis kezelési parancsok áttekintése. Védelmi és optimalizálási elemek. API bemutatása.	MongoDB kezelése, API programozása (egyéni programozási feladat)
9	Gráf adatbázis modell. A Neo4J működési modellje. Védelmi és optimalizálási elemek.	Neo4J kezelése, API programozása
10	Gráf adatbázis modell programozása, API bemutatása.	Neo4J kezelése, API programozása (egyéni programozási feladat)
11	Big Data rendszerek, Cloud architektúra, Hadoop architektúra,	Cassandra adatbázis kezelése
12	Big Data rendszerek,, Cassandra	Protege kezelése

adatbázis, API elemek	
-----------------------	--

Kötelező irodalom:

- Kovács László: Adatbázis rendszerek elektronikus jegyzet (www-db.iit.uni-miskolc.hu)

A tárgy lezárásnak módja: aláírás és vizsgajegy

Évközi számonkérések:

- A félév során négy egyéni programozási feladat
- gyakorlati feladatok teljesítése

Vizsga formája:

írásbeli és szóbeli

Az írásbeli rész legalább elégséges teljesítése után következik a szóbeli rész. Az írásbeli és szóbeli rész értékelése:

0%-50% : elégtelen, 51%-62% : elégséges
63%-75% : közepes 76%-88% : jó
89%-100% : jeles

Az eredő teljesítmény a $0.667 \cdot \text{írásbeli} + 0.333 \cdot \text{szóbeli}$ képlettel kerül meghatározásra, melyhez jegy a megadott táblázat szerint rendelődik.

Elégtelen írásbeli vagy elégtelen szóbeli elégtelen vizsgajegyet jelent. A szóbelin a megjelenés kötelező. Az a hallgató, aki az írásbeli részen részt vett, de a szóbelin nem, „Nem jelent meg” Neptun bejegyzést kap.

A vizsgáztató oktatónak – ellenőrzési célból – joga van az írásbeli dolgozat egyes kérdéseinek szóban való ismételt reprodukálását kérni a hallgatótól.

Minta vizsga kérdései

1. LDAP védelmi modell elemei , felhasználók nyilvántartása
2. ORDBMS: metódusok típusai, definíciós és felhasználási parancsok
3. LINQ: lambda kalkulus alapú működési modell
4. A Neo4J adatbázis adatmodell és adatkezelő parancsai

Vizsga javítókulcs

1. LDAP védelmi modell elemei , felhasználók nyilvántartása (10 pont)
 - minden felhasználó egy LDAP entry
 - az entry-ben van password attribute
 - azonosító: entry DN
 - ACL alapú
 - TO nodes BY user operation
 - felhasználó megadás: * | self | anonymous | users | dn.regex = DN
 - objektum megadás: * | dn.base=DN | dn.one=DN | dn.subtree=DN | dn.regex = DN filter=kif attrs=tul.lista
 - művelet: none | auth | search | read | write
 - konfiguráció fileban adott
 - fontos a sorrend
2. ORDBMS: metódusok típusai, definíciók és felhasználási parancsok (10 pont)
 - metódusok: objektum, osztály, konstruktor, konvertáló
 - MEMBER method; STATIC method; CONSTRUCTOR method; MAP method; ORDER method
 - aktuális objektum: SELF
 - MEMBER FUNCTION
 - MEMBER PROCEDURE
 - CONSTRUCTOR FUNCTION name (args) RETURN SELF AS RESULT
 - hívás: objektum.tagmetódus()
 - hívás: osztály.tagmetódus()
 - overloading
 - overriding
3. LINQ: lambda kalkulus alapú működési modell (10 pont)
 - elemi, noname függvény
 - lambda.x jelentése
 - metodus (x => y)
 - projekció: C.Select(x => x.m)
 - szelekció: C.Where(x => exp)
 - redukció: C.Distinct()
 - kiválasztás:C.First()
 - rendezés: C.OrderBy(x => x.m)
 - aggregáció: C.Aggregate(init, (a,b) => exp)
 - join: C.Join(D, x => x.m, y => y.n, (x,y) => exp)
 - csoportképzés: C.GroupBy (x => exp)
4. A Neo4J adatbázis adatmodell és adatkezelő parancsai (10 pont)
 - NoSQL
 - gráf alapú
 - JSON objektumok
 - pointer alapú kapcsolat + indexelés
 - CYPHER nyelv
 - node : (), edge : - []->

- elemek: típus (label) + tulajdonság (attribute)
- felvitel: CREATE (a : lab { p1:v1,...})
- lekérdezés: MATCH exp :
- módosítás: MATCH exp SET kif
- törlés? MATCH exp DELETE kif