

10. tétel

10. tétel

Protected. Adatelérési komponensek

A, Protected: az osztályok bármennyi adatot és tagfüggvényt tartalmazhatnak. Az adatokat el kell rejtenünk, be kell tokoznunk az azokat használó osztály belsejébe. A betokozás nagy előnye, hogy az osztály létrehozója bármikor módosíthatja az osztály belső ábrázolását.

A betokozás lényege egyszerű: úgy kell elképzelni egy osztályt, mint egy 'fekete dobozt', amelynek csak kis része látható. A doboz belsejét nem látjuk, kizárólag a rendelkezésünkre álló felületen, az interfészen, keresztül férhetünk hozzá. Az objektumok használata közben a kód nagyobb része rejtve marad. Általában nem tudjuk, milyen belső adatokat tárol az objektum, és nem is férhetünk hozzá közvetlenül. Az adatokat csak tagfüggvényeken keresztül érhetjük el, amelyek megvédik az osztályt az illetéktelen hozzáféréstől. Ezt az adatok elrejtésének (data hiding, information hiding) nevezzük. A Delphi osztály alapú betokozást valósít meg. Az egységek felületirő (interface) részében meghatározott azonosítók a program többi egységéből is elérhetők, feltéve, hogy a 'uses' utasításban felsoroljuk az azonosítót tartalmazó egység nevét.

Hozzáférési szintek:

1. *private*: utasítással megjelölt tagfüggvények vagy mezők az osztályt bevezető egységen kívül nem érhetők el.
2. *protected*: az adott mező vagy tagfüggvény korlátozott láthatósággal rendelkezik. Kizárólag az adott osztályban, illetve annak leszármazottaiban férhetünk hozzá. Tehát a mező tartalmához csak az osztály, az alosztályok és az egységen belüli kódrészletek férhetnek hozzá.
3. *public*: a program tetszőleges pontján, szabadon hozzáférhetők.

Az osztályok mezőinek általában privátnak kell lenniük, a tagfüggvények többnyire nyilvánosak.

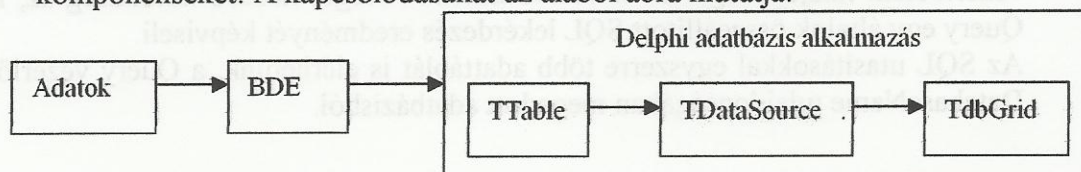
A hozzáférést szabályozó kulcsszavak kizárólag az adott egységen kívüli kóddal szemben védik az osztály felületleíró részben bevezetett tagjait. Ez azt jelenti, hogy ha ugyanabban az egységben határozzunk meg két osztályt, nem tudjuk megvédeni azok privát mezőit.

B, Adatelérési komponensek: a Delphi alkalmazásban az adatok kezelésére két komponens paletta áll a rendelkezésünkre.

DataAccess: szerepe a nem vizuális komponensek adatbázisokkal való kapcsolat BDE segítségével történő kiépítésére, adatbázisok adataiból felépített lekérdezések táblái, az SQL használatának támogatása.

DataControls: vizuális vezérlők adatbázis-kezelés eredményeinek megjelenítéséhez használjuk, adattáblák megjelenítésére, navigációs elemek, szövegek, képek stb. Csak az adatok megjelenítésére szolgáló komponensek tartoznak ide. Ilyen például a TdbGrid komponens, amely az adatállomány táblázatos megjelenítését teszi lehetővé.

Az adatelérési és megjelenítési komponensek kapcsolata: a delphi alkalmazásban elkülönítjük a felhasználói felületen megjelenő adatokat, valamint a tényleges adattároló komponenseket. A kapcsolódásukat az alábbi ábra mutatja:



10. tétel

A TDataSource egyfajta híd szerepét tölti be az adatelérési és adatmegjelenítési komponensek között. Az adatelérési komponensek az alkalmazás *adatmoduljában* találhatóak, míg az adatmegjelenítő komponensek a felhasználói felületen (formon) találhatóak.

Táblák: TTable komponens valósítja meg. 1 db adattáblát érünk el 1 db TTable komponenssel. TTable táblát nem használhatunk lekérdezésre. SQL lekérdezésre a TQuery komponens alkalmas. Ez már több táblás lekérdezésre is alkalmas. A lekérdezéseket paraméterezni lehet.

```
Table1.Active :=false;
Table1.DatabaseName:='dbDemos';
Table1.TableName:='valami.db';
Table1.Active:=true;
```

A Table vezérlőelem segítségével minden olyan adattáblára hivatkozhatunk, amelyet a BDE eszközeivel elérünk. Az elérni kívánt adatbázis logikai nevét a vezérlőelem DatabaseName tulajdonságában kell megadni. A TableName tulajdonság értékét pedig az adatbázison belüli táblák egyikének a nevére kell beállítanunk.

A TTable alapvető tulajdonságai: az adatelérési komponensek közül ez az egyik legfontosabb. Ez jeleníti meg az alkalmazásunkban az egyes adattáblákat. Legfontosabb tulajdonságai:

Active: *property Active: boolean*; az adattábla aktiválása (megnyitása) és inaktiválása (bezárása). Hatása azonos az Open és a Close metódusokkal.

DatabaseName: *property DatabaseName: string*; az adattáblához kapcsolódó adatbázis alias neve.

Exclusive: *property Exclusive:boolean*; kizárólagos megnyitás beállítására szolgáló tulajdonság. Átállítása csak bezárt adatállomány esetén lehetséges.

Filter: *property Filter:string*; az adattábla szűrésére szolgáló tulajdonság. A szűrési feltételt egyszerű sztringként kell megadni. Pld.: VAROS='Budapest'

Filtered: *property Filtered:boolean*; a tulajdonság igaz, ha a tábla szűrés alatt áll.

FilterOptions: szűrési opciók beállítása. A kis és nagybetűk megkülönböztetését, valamint a részleges találatok engedélyezését tehetjük lehetővé.

IndexFiles: *property IndexFiles:TStrings*; az indexállományok beállítására szolgáló tulajdonság. Az IndexFileName és az IndexName tulajdonság is ezt állítja.

ReadOnly: *property ReadOnly:boolean*; csak olvasható módon nyitja meg a táblát.

TableName: *property TableName:TfileName*; az adattábla nevének (tényleges file-nevének, fizikai nevének) megadására szolgáló tulajdonság. Az adott alias-névhez tartozó elérési úton levő adattáblákat mutatja, ebből választhatjuk ki a táblát.

TableType: az adattábla típusának beállítására szolgál. Alapértelmezése a ttDefault, amely a tábla eredeti formátumát jelenti.

A Query komponens: a Table mellett a Query vezérlőelemet is használhatjuk adatok eléréshez.

A vezérlő tulajdonságainak és metódusainak nagy része megegyezik a Table vezérlőelem tulajdonságaival és metódusaival. A legfontosabb különbség az, hogy a Query egy általunk összeállított SQL lekérdezés eredményét képviseli.

Az SQL utasításokkal egyszerre több adattáblát is elérhetünk, a Query vezérlőelem DatabaseName tulajdonságában megadott adatbázisból.

10. tétel

TDataSource komponens: az adatok megjelenítését teszi lehetővé úgy, hogy a különböző adatelérési komponenseket összekapcsolja az adatmegjelenítő komponensekkel. Alapvető tulajdonságai:

DataSet: annak az adathalmaznak (pld.: TTable komponensnek) a neve, amelynek tartalmát továbbítja az adatmegjelenítő komponensekhez, és amelybe visszaír, ha a felhasználó a megjelenítő komponenseken módosítja az adatokat.

AutoEdit: beállítása során ha a felhasználó az adatmegjelenítő komponenseken változtat, akkor az adathalmaz dsEdit állapotba kerül és az adattábla módosítható lesz.

TdbGrid komponens: az egyik legjobban használható és legegyszerűbb adatmegjelenítő komponens. A megadott adatforráshoz rendelt adathalmaz adatait jeleníti meg táblázatos formában.

Navigációs komponensek: az adattáblában történő pozicionálást valósítják meg.

A, First: Table1.First; // az adattábla első rekordja

B, Last: Table1.Last; // az adattábla utolsó rekordja

C, Next: Table1.Next; // az adattábla következő rekordja

D, Prior: Table1.Prior; // az adattábla előző rekordja

E, MoveBy: a táblában való előre (+) vagy vissza (-) való léptetés megadott számú rekorddal:
Table1.MoveBy(-2);

F, BOF, EOF: az adattábla elejének (BOF) vagy végének (EOF) tesztelése:
if Table1.BOF then... illetve **if Table1.EOF then...**