A programról:  
  
A program témája horgász fogási napló vezetése.

A programban rögzítet adatok megfelelnek a papír alapú Fogási naplóval.

Rögzítésre kerülő adatok a következők:

* Egyesület azonosító
* Horgász azonosító
* IRSZ+város megnevezés
* Víztér azonosító
* Halőr azonosító
* Horgászjegy azonosító
* Halak azonosító

Természetesen mindegyik azonosító mellé tartozik egy megnevezés is stb.

Az adatfeltöltés folyamán 7db törzs állományt kell feltölteni a biztonságos működéshez.

Mindegyik törzsállomány tartalmát ki tudjuk listázni képernyőre amolyan kontrollként.

A fő táblákat a Fogási napló fej és Fogási napló sorok alkotják, mivel gyakorlatilag ide kerül rögzítésre a tényleges fogás halfajonként, vízterenként dátumra bontva.  
  
A program MAGIC Software Enterprises Inc. szoftverrel lett elkészítve.

A Magic egy gyors alkalmazásfejlesztő és futtató (**R**apid **A**pplication and **D**evelopment **D**esign) rendszer, amelynek használata a vállalkozások költségeit jelentősen csökkenti. Egy egységes fejlesztési elv keretében a Magic lehetővé teszi a fejlesztők számára, hogy vállalkozás-szintű alkalmazásokat fejlesszenek ki, amelyek a számítógépek, operációs rendszerek, és adatbázisok széles skálájára átvihetőek mind a Web-alapú, mind a hagyományos kliens/szerver alkalmazások esetében. A Magic fejlesztői környezet integrálja egy nagyon könnyen hordozható RADD-környezet előnyeit egy skálázható full-featured tranzakciófeldolgozó rendszerével, lehetővé téve a fejlesztők számára, hogy rövid határidőre és olcsón készítsenek el alkalmazásokat. A fejlesztők ugyanazt a fejlesztési elvet, fejlesztői környezetet használják a host, a kliens/szerver és az Internet alapú számítástechnika számára, beleértve az n-szálú architektúrát és az alkalmazás-partícionálási sémákat is. A Magic lehetővé teszi a fejlesztők számára a hagyományos GUI-kliens megépítését ugyanúgy, mint a fejlett HTML és JAVA felületek megépítését az Internet számára külső, HTML vagy JAVA fejlesztőeszközök segítsége nélkül. Egy vizuális fejlesztési elvrendszer használatos a Webszerver alkalmazásokhoz, GUI-kliensekhez vagy a szöveges űrlapokhoz.

A Magic elvrendszerének alapvető koncepciója a skálázható, nyílt számítástechnikai megoldások gondolata. A Magic az örökölt rendszerek legjobb integrálója a piacon, könnyedén alkalmazható a vegyes platformokkal és adatbázisokkal működő heterogén, többszálú architektúrájú környezetekben is. A szervezetek hasznosíthatják általa az örökölt hardver- és szoftvervagyonukat, ugyanakkor gazdaságos átjárást biztosíthatnak a kliens/szerver környezetbe és az Internet-alapú számítástechnikába. A munka elindulhat a kliens/szerver architektúra és a Web irányába úgy, hogy megőrzi a korábbi befektetéseket az ISAM vagy SQL alapú adatok, hálózati számítógépek, grafikus munkaállomások és webböngészők által történő egyidejű elérésével. A Magic nem használja az RDBMS fejlesztőeszközök szabadalmaztatott lock-injeit.

A Magic-ben nincs hagyományos értelemben vett forráskód és nem lehet vele futtatható kódot generálni. Interpreter típusú program nyelv, a programkód futás közben fordítódik le, nem gyártható vele például exe fájl. Futtatni és fejleszteni is csak a megfelelő Magic környezet alatt lehet.

Fejlesztés során mindig táblázatokat töltünk ki, ez zavaró lehet annak, aki hozzászokott ahhoz, hogy programlistákat böngésszen. Magic-ben a fejlesztő által írt programok csak több táblázatba való átkapcsolgatással tekinthetők meg. Van azonban lehetőség a hagyományos értelemben vett programlisták gyártására is.

A Magic önmagában nem tud fájlokat kezelni, a fájlokat (R)DBMS-ek kezelik. Ezeket a Magic ún. gateway-eken keresztül éri el, így (R)DBMS független. A tranzakciók kezelését saját maga végzi minden (R)DBMS használata esetén. A fájldefinícióknál több, a Magic által támogatott fájlkezelőt is lehet választani. Egy futó program képernyőjén is lehet, hogy több különböző fájlkezelő által kezelt fájl adata látható. Egy adott program tud például Btrieve vagy Oracle fájlban levő adatokat kezelni anélkül, hogy a programot újra kellene írni, csak a fájl definíciónál kell a fájlkezelőt módosítani. A programkód is gateway-en keresztül elért fájlkezelő által kezelt fájlban van tárolva.

A Magic Cliens/Server üzemmódban is képes működni. (Az általunk kezelt adatbázis-fájlokat, nyomtatókat, stb. akár különböző platformú szervereken is elérhetjük.) Ehhez különböző kommunikációs lehetőségek állnak rendelkezésre (pl. PC/TCP, Winsock, stb).

Az utóbbi években tapasztalható a hagyományos „vízesés” megközelítésű rendszerszervezés felcserélése a gyors alkalmazásfejlesztés és futtatás (RADD) módszerére. A hagyományos módszer a végfelhasználókat a fejlesztés korai szakaszában az igényeik meghatározására kényszerítette, nem biztosította azt a rugalmasságot, amelyre a dinamikus szervezeteknek szüksége van. A RADD lehetővé teszi az üzleti környezet számára, hogy a végfelhasználó iteratív módon, a prototípus készítési technika által résztvegyen a fejlesztés minden fázisában.

A gyors alkalmazásfejlesztés és futtatás számára az egyik legjobb szemlélet a dinamikus rendszerszervezési módszertan (Dynamic System Development Method – DSDM). A Magic a független DSDM-konzorcium egyik alapító tagja, szorosan együttműködik a szervezettel. A DSDM biztosítja a megközelítést és a módszertant a RADD projektkörnyezet számára, beleértve a nagy projekt- és konfigurációkezelő technikákat, irányelveket a kockázat- és erőforrásbecsléshez illetve a minőségi és alkalmassági kérdésekben.

A deklaratív programozás azon programozási szemléletmódok összefoglalására használatos kifejezés, amelyben a programozó erőfeszítései inkább az alkalmazás logikájára összpontosulnak, mint a megvalósítás részleteire. A deklaratív programozás az adatbázis-alkalmazás fejlesztés sok fontos aspektusán keresztül az egész Magic-ben megnyilvánul, beleértve az adatmanipulációt, az űrlapok tulajdonságait és az eseménykezelést.

Az adatmanipuláció programozásának egy nem deklaratív megközelítésében, a programozó jellemzően explicit módon ott határozza meg a kódban azt a bizonyos végrehajtandó adatbázistevékenységet, ahol logikailag végre kellene hajtódnia. Ezen tevékenységek közé tartozik egy rekord lekérdezése, egy új rekord hozzáadása, egy meglévő rekord törlése, vagy egy létező rekord módosítása.

A Magic-nek azonban nincsenek ilyen jellegű adatbázisműveletek végrehajtására szolgáló parancsai. A legkisebb teljes feldolgozási egység a Magic-ben a taszk. Egy üzemmód taszkhoz rendelésével, a Magic alkalmazásmotor megtudja, hogy milyen adatbázisműveletet kell végrehajtania. Az alkalmazásfejlesztő az alkalmazás fejlesztése közben eldönti a taszk végrehajtási módját (módosítás, létrehozás, törlés, lekérdezés).

A Magic által használt deklaratív programozás másik példája a Magic űrlapszerkesztőjénél játszik fontos szerepet. Nincs szükség procedurális kódolásra egy űrlap jellemzőinek megtervezéséhez és beállításához. Egy tulajdonságablak az opciók széles körét biztosítja. A Magic újraszámító motorja lehetővé teszi, hogy a módosítások eredményét azonnal lássuk a belső konzisztencia megőrzése mellett.

A Magic eseménykezelése a harmadik példánk a deklaratív programozás Magic által történő támogatására. A nem procedurális kódolás jelentősége abban áll, hogy programok vagy altaszkok végrehajtását gombnyomáshoz, eltelt időhöz vagy egy feltétel igazzá válásához köti. Például a felhasználó megnyomhat egy előre definiált gombot, amely elindít egy eseményt és végrehajtathat egy eseményprogramot egy mező szerkesztése közben. Amikor az eseményprogram vagy az altaszk befejeződik, a végfelhasználó folytathatja a mező szerkesztését ugyanattól a karaktertől, ahova a kurzor mutatott az esemény elindítása előtt.

A Magic-ben a billentyűzet beállításait át lehet állítani (pl. ha valahol nincs CTRL billentyű, vagy nem működik, nem okoz gondot). Nyomtatáshoz könnyen el lehet készíteni a nyomtatandó formátumokat és a programok módosítása nélkül lehet új nyomtatót használni a nyomtatáshoz. Ismeri a különböző nyelveknek megfelelő beállításokat (pl. a hét napjait és a hónapok neveit tudja magyarul).

Segíti a különböző kódlapokról való adatelérést (pl. CWI-ben tárolt adatokat tud megjeleníteni a 852-es kódlapnak megfelelően). A fájlokban fájlkezelőtől függően tud bármilyen ABC sorrendet kezelni (pl. a fájlban az adott mező szerinti sorrendben lehet …aábcdeé…z… sorrend és lehet …abcd…záé…).

Lehetőség van arra, hogy többen is fejlesszenek egyidőben ugyanabban az alkalmazásban, sőt miközben fejlesztenek, mások már futtathatják is az alkalmazást.

Mivel nincs forráskód és futtatható kód közti különbség, ezért ha valamit változtatunk az alkalmazáson, az azonnal érvénybe lép (pl. ha kitörlünk egy programot, azt általában újra meg kell írni, mivel nincs lehetőség arra, hogy az alkalmazáskód változtatásakor az új verziót más néven vagy máshova elmentsük). Ha azt akarjuk, hogy megmaradjon az előző állapot is az alkalmazásunkról, akkor nem a Magic-nek, hanem nekünk kell arról gondoskodnunk, hogy módosítás előtt az operációs rendszerben lemásoljuk az alkalmazásunkat egy másik helyre.

A fejlesztés és futtatás során mindig segítségünkre lehet az, hogy a Magic használata menüvezérelt. Fejlesztés során majdnem minden lépésben tudunk segítséget kérni a Magic-től (a billentyűzet beállításától függően, általában F1). Ha a menü sorai, illetve a segítő képernyő sem tudott segíteni a továbblépésben, akkor érdemes megnézni a státusz sort. Értékes információkat kaphatunk arról, hogy milyen (állapotban) módban dolgozunk éppen.

Az illetéktelen fejlesztés megakadályozására és a futtatás hangolására a Novellhez hasonló jogosultságrendszer működik.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Alkalmazás** | |  |  |  |  |  |
|  | Fájl definíciók |  | Típusok |  | Programok |  |
|  |  |  |  |  |  |  |
|  | Segítő képernyők |  | Jogok |  | Alkalmazás szintű események |  |
|  |  |  |  |  |  |  |
|  | Kinyíló menük |  | Legördülő menük |  | Alkalmazás jellemzők |  |
|  |  |  |  |  |  |  |

A Magic csökkentette a nyelv szintaxisát, míg megtartotta a szemantika gazdagságát, megszüntetve ezáltal az interpreteres nyelvekhez kapcsolódó tipikus lassúságot. A Magic-ben a feldolgozási idő nagy része az alkalmazás funkcióinak végrehajtásával telik. Ezek a funkciók az alkalmazásmotor részeként beépítettek, és erősen optimalizált, natív kódként kerülnek megvalósításra. Ugyanakkor, ezek a funkciók elegendően paraméterezettek, így lehetővé teszik a fejlesztő számára az alkalmazások széles választékának leírását. Az alkalmazás teljes leírása, beleértve a logikát, az üzleti szabályokat, az űrlapokat és az adatokat, a Magic gyűjtőjében kerülnek tárolásra. Ez a Magic kontrollfájlja. Egy Magic alkalmazás logikailag a fenti táblázat elemeinek összességéből áll, fizikailag pedig egy egyszerű adatbázis fájl, amit gyűjtőnek nevezünk. Ennek megfelelően a fejlesztő környezet

Gyűjtő menüpontja a következő alpontokból áll:

|  |  |
| --- | --- |
| **Gyűjtő** | |
|  | **Típusok** |
|  | **Táblák** |
|  | **Programok** |
|  | **Segítő képernyők** |
|  | **Jogok** |
|  | **Menük** |
|  | **Alkalm. események** |
|  | **Alkalm. jellemzők** |
|  | **Mentés/Beolvasás** |

## Magic Keresztreferencia

A Magic keresztreferencia segédprogram segít megkeresni, hogy az egyes objektumokat, mint például a típusokat, mezőket vagy programot az alkalmazás milyen helyein használunk. Például kiváncsiak lehetünk arra, hogy milyen programokban használunk egy mezőt, egy indexet vagy egy táblát.

Kereszthivatkozások listájának elkészítése a következő lépésekből áll:

1. Kiválasztjuk a gyűjtőben a vizsgálni kívánt objektumot.
2. Elindítjuk a segédprogramot az Opciók/Kereszthivatkozás menüpont aktivizálásával.
3. Kiválasztjuk a kereső területet. Bizonyos esetekben a Magic kérdéseket tesz fel, és ezek során választhatjuk ki a kereső területet. Például típusokat kereshetünk programokban vagy táblákban.
4. A kapott listából újabb táblákat nyithatunk meg az F5 gombbal.

Például egy programot hívó programok listáját úgy kapjuk meg, hogy ha a program gyűjtő táblában kijelöljük a programot és az Opciók/Kereszthivatkozás menüpontra rákattintunk. Vagy egy adott indexet használó programok listáját úgy kapjuk meg, hogy a tábla gyűjtőt megnyitjuk, és kiválasztjuk azt a táblát, amelyik a keresett indexet tartalmazza. Zoommal az index gyűjtőbe jutunk, ahol ki kell választanunk a keresett indexet, majd rá kell kattintanunk az Opciók/Kereszthivatkozás menüpontra. Azoknak a programoknak illetve indexeknek a listáját, amelyek egy bizonyos mezőt használnak, úgy kapjuk meg, hogy ha a tábla gyűjtőben kiválaszjuk azt a táblát, amelyik az adott mezőt tartalmazza, megnyitjuk az oszlop táblát és az Opciók/Kereszthivatkozás menüpontra rákattintunk. Azoknak a programoknak a listáját, amelyek egy bizonyos táblát használnak, úgy kapjuk meg, hogy ha a tábla gyűjtőben kiválaszjuk azt a táblát és az Opciók/Kereszthivatkozás menüpontra rákattintunk.

A segédprogramot rekurzívan is használhatjuk, azaz ha a kapott listából egy elemet kiválasztunk, még egyszer rákattinthatunk az Opciók/Kereszthivatkozás menüpontra. Például, ha megkaptuk egy bizonyos típust használó programok listáját, akkor kijelölhetünk egy programot és az Opciók/Kereszthivatkozás menüponttal megkapjuk azoknak a programoknak a listáját, amelyek ezt a programot használják. Vagy ha megkaptuk egy bizonyos mezőt használó indexek listáját, akkor kijelölhetünk egy indexet és Opciók/Kereszthivatkozással megkapjuk azoknak a programoknak a listáját, amelyek azt az indexet használják.

A következő táblázat mutatja, hogy ha egy objektumot kiválasztunk, akkor milyen azt használó objektumok listáját kaphatjuk meg:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Típus |  |  |  |  |
|  |  |  |  |  |
| Tábla |  |  |  |  |
| zoom |  |  |  |  |
| Mező |  |  |  |  |
| zoom |  |  |  |  |
| Kulcs |  |  |  |  |
| zoom |  |  |  |  |
| Kulcs szegmens |  |  |  |  |
|  |  |  |  |  |
| Program |  |  |  |  |
| zoom |  |  |  |  |
| Taszk |  |  |  |  |

## A Vezérlő motorra érvényes végrehajtási szabályok

### A taszk ciklus szintjei

Az alkalmazásvezérlő motor alapciklusa látható az ábrán. Az ábrán mindegyik téglalap egy-egy specifikus taszk végrehajtási szintet jelöl, és a szintek téglalapjaiban lévő újabb téglalap jelöli a hozzájuk tartozó műveleti táblázatot.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  | **TASZK CIKLUS** |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  | Taszk előfeldolgozási szint |  |  |  |  |
|  |  |  |  | Taszk előfeldolgozási műveletek |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  | **REKORD CIKLUS** |  |  |  |  |
|  |  |  |  | Rekord Előfeldolgozási szint |  |  |  |  |
|  |  |  |  | Rekord Előfeldolgozási műveletek |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  | **MEZŐ CIKLUS** |  |  |  |  |
|  |  |  |  | Rekord Főfeldolgozási szint |  |  |  |  |
|  |  |  |  | Rekord Főfeldolgozási műveletek |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  | Rekord Utófeldolgozási szint |  |  |  |  |
|  |  |  |  | Rekord Utófeldolgozási műveletek |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  | Taszk utófeldolgozási szint |  |  |  |  |
|  |  |  |  | Taszk utófeldolgozási műveletek |  |  |  |  |
|  |  |  |  |  |  |  |  |  |

Kilépés a taszkból

A motor alapvető ciklusai a fő fájl rekordjain

Az egyes műveleti táblázatokban szereplő műveletek csupán egy részét jelentik mindannak a tevékenységnek, amit a vezérlő motor a megfelelő szinten elvégez. A motor adatbázis-kezelési műveleteket és még más egyéb színfalak mögött zajló tevékenységeket is végez.

A motor taszk ciklusán belül a következő két egymásba ágyazott ciklus van:

* A rekord ciklus, amelynek szintjei minden egyes rekordra ismétlődnek.
* A mező ciklus, amely a képernyőn látható rekord és a végfelhasználó közötti interakciót valósítja meg. Ez csak online taszkok esetében létezik.

Az alábbi leírások ismertetik, hogy a motor milyen feltételek mellett hajtja végre az egyes szinteket, és milyen lépéseket tesz meg az egyes szinteken.

### A taszk ciklus

A motor taszkonként egyszer hajtja végre az alábbi lépéseket.

#### Taszk előfeldolgozási szint

1. Ha a taszkot egy másik taszk hívta, minden paramétert virtuális változókban vesz át.
2. Megalkotja a taszk adatnézetét, vagyis eldönti, hogy mely rekordokat kell felhasználni, és hogyan kell ezeket sorrendbe állítani.
3. Végrehajtja a Műveletek [Taszk Előfeld] táblázatban szereplő műveleteket abban a sorrendben, ahogy a táblázatban megjelennek.
4. Ha online taszkról van szó módosítási (vagy lekérdezési) üzemmódban, akkor kiszámítja az Init és a Link kifejezéseket, és ezek alapján meghatározza, hogy – az Ismétlési terület határai és az Ismétlődő terület definíciókat figyelembe véve – hány rekord fér el a képernyőn és meg is jeleníti a rekordokat.
5. Végrehajtja a rekord ciklust.

#### Taszk Utófeldolgozási szint

1. A rekord ciklus végén végrehajtja a Műveletek [Taszk Utófeld] táblázatbeli összes műveletet abban a sorrendben, amilyenben a táblázatban szerepelnek.
2. Kilép a taszkból.

### A Rekord ciklus

Az alábbi lépéseket a motor online taszkban egyszer hajtja végre minden olyan rekordra, amelyet a végfelhasználó fellapozott, batch taszkban pedig egyszer hajtja végre az adatnézet összes rekordjára, az első rekordtól kezdve.

#### Rekord ciklus folyamatdiagramjai

A következő oldalakon látható a rekord ciklus vezérlési folyamatának grafikai ábrázolása online és batch taszkokra folyamatdiagramok segítségével. A diagramok segítenek eligazodni, ha pontosan akarjuk. tudni, mit csinál a vezérlő motor a taszk végrehajtása közben.

#### Rekord ciklus online illetve batch taszkban

Az alábbi ábra bemutatja az alkalmazásvezérlő motor végrehajtási folyamatát, amíg végigmegy egy online taszk rekord ciklusán. Az utána levő ábra pedig az alkalmazásvezérlő motor végrehajtási folyamatát mutatja be, amíg végigmegy egy batch taszk rekord ciklusán. Az ábrák részletei segítenek eldönteni, hogy melyik táblázatba helyezzünk el egy adott műveletet, illetve segítenek megtalálni az okát, hogy a taszkunk miért nem működik úgy, ahogyan vártuk.

Taszk utófeld.

végrehajtása

Taszk vége

(utána)

Igaz

ESC lett leütve

Módosítások mentése lemezre

Rekord törlésre

jelölése

Rekord utófeld.

műv.végrehajt.

törlés módban

Rekord utófeld.

műveletek

módosító módban

Mező ciklus

Rekord főfeld.

műveletek

Rekord előfeld.

műveletek

Taszk vége (előtte)

Igaz

Init és link

kifejezések

kiszámítása

Következő

rekord

behozása

Taszk előfeld.végreh.

után

Rekord

módosult

Törlés

aktiválva?

Rekord

módosult

Rekord utófeld.

műv.végrehajt.

módosító módban

Rekord ciklus online taszkban

Rekord törlés jelölése

Utófeldolgozás végrehajtása törlés módban

Aktuális rekord váltási előfeld. végrehajtása

Előző rekord váltási utófeld. végrehajtása

Taszk utófeld.végreh.

Módosítások mentése lemezre

Taszk vége (előtte)  
 igaz

Van még további rekord?

Taszk vége (utána)

igaz

Rekord törlés végrehajtása igaz

Init és link

kifejezések

kiszámítása

Következő

rekord

behozása

Taszk előfeld.végreh. után

Váltási mező módosult?

Kezd. Mód törlés

Rekord utófeldolg. műveletek végrehajtása

Rekord ciklus batch taszkban

#### Rekord inicializálási szint

1. Behozza a rekordot a memóriába úgy, hogy a mezői elérhetők legyenek a feldolgozás számára.
2. Kiértékeli a Select virtual műveletek Init kifejezéseit (a Rekord Főfeldolgozás táblázatból) és inicializálja a megfelelő változókat.
3. Ha Vége ellenőrzés = Előtte (lásd: taszkvezérlő paraméterek), kiértékeli a Taszk vége kifejezést, és ha ennek értéke Igaz, akkor kilép a taszkból.
4. Végrehajtja a Műveletek [Rekord Előfeld] táblázat műveleteit.
5. Ha a taszk Beviteli üzemmódban van, kiértékeli a Select fields műveletek Init kifejezéseit (a Rekord Főfeldolgozás táblázatból) és inicializálja a megfelelő változókat.
6. Kiszámítja az aktuális rekordhoz tartozó összes Linket.
7. Ha online taszkról van szó, végiglép a Rekord Főfeldolgozás műveleti táblázat műveletein, ami azt jelenti, hogy végrehajtja a Mező ciklust (ez nem más, mint a végfelhasználóval történő interakció).

#### Rekord lezárási szint

1. Végrehajtja a Rekord Utófeldolgozás műveleti táblázatot

* Ha batch taszkról van szó, végrehajtja a Műveletek [Rekord Utófeld] táblázat összes műveletét.
* Ha online taszkról van szó, akkor csak abban az esetben hajtja végre a Rekord Utófeldolgozási szintet, ha az aktuális logikai rekord módosult. A Magic akkor tekinti úgy, hogy az aktuális logikai rekord módosult, ha az alábbiak közül bármelyik fennáll:

1. A végfelhasználó átszerkesztett egy mezőt (vagy egy érték beírásával, vagy a Field Editor segítségével).
2. Egy bármilyen fajta Update Var művelet módosított egy változót.
3. Egy bármilyen fajta Update Var művelet az aktuális taszk egy altaszkjának végrehajtásával módosított egy változót.
4. Egy Input Form művelet adatot vitt be egy input fájlból valamelyik változóba.
5. Az Utófeldolgozás Mindig paraméter értéke Igaz. (A paramétert a Taszkvezérlés párbeszédablakban lehet beállítani.)
6. Kezeli a rekordtörlési eljárást.

Lásd az ábrát az adatnézet felépítéséről.

#### Rekordtörlés online taszkban

Ha egy online taszkban a végfelhasználó a Szerkesztés/Sortörlés (F3) parancsot választja ki, vagy ha a Rekordtörlés paramétert Igazra állítja, a vezérlő motor elindítja az alábbi törlési eljárást:

1. Ha a rekord nem módosult, akkor csak egyszer hajtja végre a Rekord Utófeldolgozást, mégpedig Törlés üzemmódban.
2. Ha a rekord módosult, végrehajtja a Rekord Utófeldolgozást Módosítás üzemmódban. Ezzel egyrészt elvégez minden ott definiált aktualizálási műveletet, másrészt végigviszi a feldolgozás logikai menetét, és másodszor is végrehajtja a Rekord Utófeldolgozást, de most Törlés üzemmódban, és ezzel elvégez minden Törlés üzemmódhoz kapcsolt (STAT(0,'T'MODE)) műveletet, mint például az "egy-a-sokhoz" relációban álló láncolt rekordok törlése egy altaszkkal, vagy elvégzi az Incremental mode-ot használó Update műveleteket.

#### Rekordtörlés batch taszkban

1. Ha a taszk kezdeti üzemmódja Törlés volt, végrehajtja egyszer a Rekord Utófeldolgozást Törlés üzemmódban.
2. Ha a Rekord törlés értéke Igaz:

* Végrehajtja a Rekord Utófeldolgozást Módosítás üzemmódban, és ezzel egyrészt elvégez minden ott definiált aktualizálási műveletet, másrészt végigviszi a feldolgozás logikai menetét.
* Másodszor is végrehajtja a Rekord Utófeldolgozást, de most Törlés üzemmódban és ezzel elvégez minden Törlés üzemmódhoz kapcsolt műveletet.

Mindhárom fenti esetben, miután végrehajtotta a Rekord Utófeldolgozást Törlési üzemmódban, a rekordot ''törölt”-nek nyilvánítja.

1. Miután végrehajtotta a Rekord Utófeldolgozást, a Fő fájlt és a csatolt fájlokat érintő minden módosítást lemezre ment. Ha a rekord nem módosult, nem menti lemezre.
2. Ha Vége ellenőrzés = Utána (lásd: taszkvezérlő paraméterek), kiértékeli a Taszk vége kifejezést, és ha ennek értéke Igaz kilép a Rekord ciklusból.

### A mező ciklus

A mező ciklus csak online taszkok esetében érdekes és mindössze egy szintet tartalmaz, a Rekord Főfeldolgozási szintet. A vezérlő motor ezt a fázist a Rekord Főfeldolgozási táblázat átvizsgálásával hajtja végre.

#### Kilépés a mező ciklusból

A mező ciklust a felhasználó az alábbi módokon zárhatja le:

* Leüt egy "rekord lezáró" billentyűt: ↑, ↓, PgUp, PgDn, Home, End vagy Esc.
* Igaz értéket ad a Taszk vége feltételnek.
* Kitölti a rekord utolsó mezőjét feltéve, hogy a Ciklikus Rekord Főfeldolgozás paraméter Nem-re lett állítva a Taszkvezérlés párbeszédablakban. Ezzel automatikusan átlép a következő rekordra.

A mező ciklus lezárása azt jelenti, hogy a végfelhasználó által a rekordon végrehajtott minden módosítás kikerül a felhasználó hatásköréből, és a motor továbbmegy a Rekord Utófeldolgozás szintjére. Ha a Taszk vezérlés párbeszédablakban a Módosítás jóváhagyása paraméter értéke Igaz, akkor a módosított rekord elhagyása előtt megjelenik a felhasználónak a Művelet jóváhagyása párbeszédablak.

### Váltási szintek

A váltási szintek további végrehajtási szintek, amelyek csak batch taszkok esetén léteznek. Ezek segítségével lehet a jelentésekben az adatokat csoportosítva megjeleníteni. (A csoportok az ún. kontrolszakítások összegezési szintek stb.).

Fontos tudni, hogy sok munkát takaríthat meg, ha a többszörös kontrol megszakításokat kezelő jelentéseket a Report Generator (Jelentéskészítő) segédprogram segítségével állítja elő.

Definiálhatunk többszörös váltási szinteket is különböző mezők alapján.

Mindegyik váltáshoz meg lehet adni váltás előfeldolgozási és/vagy váltás utófeldolgozási műveleteket. A vezérlő motor minden olyan esetben végrehajtja ezeket a műveleteket, amikor érzékeli, hogy a váltást definiáló mező megváltozott és csak ezután kezd hozzá az aktuális adatnézeti rekord feldolgozásához a rekord fázisban. Ezzel tehát figyelembe veszi azokat a műveleteket is, amelyeket az egyes rekordblokkok végén kell végrehajtani, ilyen p1. a részösszegek vagy teljes összegek nyomtatása.

Miután a vezérlő végrehajtotta a váltási utófeldolgozási műveleteket az előző adatnézet-rekordra, végrehajtja a váltás előfeldolgozási műveleteket az aktuális adatnézeti rekordra, és csak utána lát hozzá a feldolgozásához. Ez lehetővé teszi, hogy a vezérlő motor elvégezzen akármilyen inicializálás-típusú műveletet, például a számlálók lenullázását vagy a fejlécek kinyomtatását és csak ezután kezdje el az új értéket tartalmazó adatnézeti rekordblokk feldolgozását.

A főfájlnak – vagy a Kulcs vagy a Rendezés paraméter szerint – rendezettnek kell lennie a váltási szintben megadott mezők alapján. A váltási szinteket fordított sorrendben kell elhelyezni a szint definíciós táblázatban ahhoz a sorrendhez képest, amelyben a kulcs mező szegmensek megjelennek a taszkhoz deklarált Kulcsban vagy a taszkhoz megadott Rendezési kulcsban:

* Elől állnak a kisebb váltási szintek, ezeket követik a közepes váltási szintek és végül a nagyobb váltási szintek. A közvetlenül a Rekord szintje után következő szint a legalacsonyabb és a közvetlenül a Taszk szintje előtti szint a legmagasabb. Ha helytelen a szintek sorrendje a táblázatban, akkor a motor helytelenül fogja végrehajtani a kontrolszakításokat.

Egy magasabb szintű váltási szinten történt módosítás változást okoz az összes alacsonyabb szintű váltási szinten is. Az előfeldolgozás a magasabbtól az alacsonyabb felé haladva, az utófeldolgozás az alacsonyabbtól a magasabb felé haladva megy végbe.

**Megjegyzés**: a váltási szintek végrehajtási sorrendjét mutatja a Taszk végrehajtási táblázat # oszlopa: a nagyobb szám jelöli a magasabb váltási szintet.

#### Hogyan hajtja végre a vezérlő motor a váltási szinteket?

Amikor egy batch taszk végrehajtása megkezdődik, a vezérlő először a Taszk Előfeldolgozást hajtja végre. Ezután az alsó adatnézeti rekord hatására végrehajtja az összes váltási szint előfeldolgozását a legmagasabbtól a legalacsonyabbig és végül a legalacsonyabb szint váltási előfeldolgozását. Mindaddig, amíg változást nem észlel, minden egyes rekordra végrehajtja a Rekord Utófeldolgozási műveleteket.

Változás bármelyik váltási szinten előfordulhat. Az első változás következtében a már feldolgozott rekordblokkra vonatkozóan a motor végrehajtja az összes váltási utófeldolgozást a legalacsonyabb szinttől a megváltozott szintig. Majd az ezután feldolgozandó rekordblokkra vonatkozóan végrehajtja az összes váltási szint előfeldogozását a megváltozott szinttől egészen a legalacsonyabb szintig.

Ezután a következő adatnézeti rekordblokkot dolgozza fel a rekord szinten egészen a következő változásig és így tovább, amíg az összes rekordot fel nem dolgozta. A végén, amikor vége feltételt észlel – vagy az összes adatnézeti rekord végén, vagy ha a Taszk vége kifejezés értéke Igaz lesz (lásd a Taszkvezérlési paramétereket) – ezt ekvivalensnek veszi a legmagasabb váltási szinttel és ennek hatására a már feldolgozott rekordblokkra vonatkozóan az összes váltási szintnek csak az utófeldolgozását hajtja végre, a legalacsonyabbtól egészen a legmagasabb szintig. Csak ezután hajtja végre a Taszk Utófeldolgozását.

#### Példák a váltási szintekre

Vegyünk például egy olyan batch taszkot, amelyik részösszegeket nyomtat ki egy rendelési sorokból álló fájlból, amelyben a sorok három kulcs szerint vannak rendezve: vevők szerint (magas szint) a rendelés száma szerint (közepes) és a termék száma szerint (alacsony). A legalacsonyabb váltási szint a termékszám megváltozásán, a második váltási szint a rendelésszám megváltozásán és a harmadik váltási szint a vevő sorszámának megváltozásán alapul.

Az első rekord feldolgozása alatt a vezérlő motor végrehajtja az összes szint váltási előfeldolgozását a legmagasabbtól a legalacsonyabbig. Miután feldolgozta az első vevő alsó rendelésében az első terméket (például kiszámolta vagy kinyomtatta), végrehajtja a termékszám (azaz az első váltási szint) utófeldolgozását.

Miután feldolgozta az első rendelésben szereplő összes terméket, végrehajtja a középső váltási szint (tehát a rendelésszám szerinti szint) utófeldolgozását, hogy ez alapján ki tudja nyomtatni az első rendelés összértékét. Ezután végrehajtja ugyanezen vevő összes rendelésére a váltási szinteket mind a termékszámra, mind a rendelésszámra vonatkozóan és csak utána hajtja végre a következő vevőhöz tartozó váltási előfeldolgozást.

Ha a rekord végrehajtása közben egy magasabb szintű váltási mező értéke megváltozik, akkor előbb végrehajtja az összes váltási utófeldolgozást egészen eddig a szintig, és csak utána kezdi el a következő rekord feldolgozását. Miután feldolgozta az összes rekordot, végrehajtja a Taszk Utófeldolgozást, vagyis ebben a példában kinyomtatja az összes vevőhöz tartozó összértékeket.

A példához tartozó különböző lehetséges végrehajtási folyamatokat illusztrálja a következő három ábra.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Végrehajtási folyamat váltás nélkül | Szint | Végrehajtás folyamata váltással |  |  |  |  |  |
|  | Taszk előfeldolgozás | taszkonként egyszer |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Vevő előfeldolgozás | vevőnként egyszer |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Rend.szám előfeld. | rendelés számonként egyszer |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Termék előfeld. | termékenként egyszer |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Rekord Előfeld. | rekordonként egyszer az adatnézetben |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  | Rekord Utófeld. |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Termék utófeld. |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Rend.szám utófeld. |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Vevő utófeld. |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Taszk utófeld. |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

Ha a termék változik

|  |  |  |  |
| --- | --- | --- | --- |
| Taszk előfeldolgozás |  |  |  |
|  |  |  |  |
| Vevő előfeldolgozás |  |  |  |
|  |  |  |  |
| Rend.szám előfeldolgozás |  |  |  |
|  |  |  |  |
| Termék előfeldolgozás |  |  | Termék utófeldolgozás |
|  |  |  |  |
| Rekord Főfeldolgozás |  | Rekord Utófeldolgozás |  |

Váltási szint végrehajtási folyamat – termék változik

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Végrehajtási folyamat váltás nélkül | Szint | Végrehajtás folyamata váltással |  |  |  |  |  |
|  | Taszk előfeldolgozás | taszkonként egyszer |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Vevő előfeldolgozás | vevőnként egyszer |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Rend.szám előfeld. | rendelésszámonként egyszer |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Termék előfeld. | termékenként egyszer |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Rekord Előfeld. | rekordonként egyszer az adatnézetben |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  | Rekord Utófeld. |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Termék utófeld. |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Rend.szám utófeld. |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Vevő utófeld. |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Taszk utófeld. |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

Ha a rendelésszám változik

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Taszk előfeldolgozás |  |  |  |  |
|  |  |  |  |  |
| Vevő előfeldolgozás |  |  |  |  |
|  |  |  |  |  |
| Rend.szám előfeldolgozás |  |  |  | Rend.szám utófeld. |
|  |  |  |  |  |
| Termék előfeldolgozás |  |  | Termék utófeld. |  |
|  |  |  |  |  |
| Rekord Főfeldolgozás |  | Rekord Utófeldolgozás |  |  |

Váltási szint végrehajtási folyamat – rendelésszám változik

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Végrehajtási folyamat váltás nélkül | Szint | Végrehajtás folyamata váltással |  |  |  |  |  |
|  | Taszk előfeldolgozás | taszkonként egyszer |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Vevő előfeldolgozás | vevőnként egyszer |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Rend.szám előfeld. | rendelésszámonként egyszer |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Termék előfeld. | termékenként egyszer |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Rekord Előfeld. | rekordonként egyszer az adatnézetben |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  | Rekord Utófeld. |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Termék utófeld. |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Rend.szám utófeld. |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Vevő utófeld. |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Taszk utófeld. |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

Ha a vevő változik

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Taszk előfeldolgozás |  |  |  |  |  |
|  |  |  |  |  |  |
| Vevő előfeldolgozás |  |  |  |  | Vevő utófeld. |
|  |  |  |  |  |  |
| Rend.szám előfeldolgozás |  |  |  | Rend.szám utófeld. |  |
|  |  |  |  |  |  |
| Termék előfeldolgozás |  |  | Termék utófeld. |  |  |
|  |  |  |  |  |  |
| Rekord Főfeldolgozás |  | Rekord Utófeldolgozás |  |  |  |

Váltási szint végrehajtási folyamat: vevő változik

## A motor vezérlési folyamatának részletes leírása

Ez a fejezet részletesen ismerteti mindazokat a lépéseket, amelyeket az alkalmazásvezérlő motor elvégez a taszkok végrehajtáskor.

A motor végrehajtási lépései függnek a taszk típusától (online vagy batch) és a kívánt működési módtól. Ezek miatt az eltérések miatt külön-külön ismertetjük az egyes szituációkat. Ennek köszönhetően, ha egyszer sikerült lokalizálni melyik szituációt akarjuk megvizsgálni, akkor már nem kell túl sok utat végigjárni.

E szabály alól az egyetlen kivétel az online lekérdezési mód. Ez azonos a módosítási üzemmóddal, azzal a lényeges különbséggel, hogy az input minden mezőre automatikusan le van tiltva, és a Szerkesztés/Sortörlés inaktív. Lekérdezési módban nem történik lemezaktualizálás.

Batch lekérdezési módban a vezérlő nem végez írás műveletet a lemezre és az összes fájlt csak olvasásra nyitja meg függetlenül a hozzájuk tartozó taszk DB fájl táblázat definíciótól.

A folyamatot tekintve a módosítás és lekérdezés üzemmódok egyformán viselkednek, ezért ezeket együtt ismertetjük.

### Online taszk folyamat

Az online taszkok módosítási és beviteli szabályainak pontos megértéséhez ismernünk kell a két interakciós üzemmódot: a Lépéses és a Gyors üzemmódot.

##### Online taszk - módosítás (lekérdezés) mód

**A: taszk inicializálása**

1. A virtuális mezők a hívó taszktól kapnak paraméter értéket, ha van egyáltalán hívó taszk. A paraméter értéket nem kapott virtuális mezők értéke 0, ha a mező numerikus, üres, ha alfanumerikus és Hamis, ha logikai.
2. Megnyitja a taszkhoz tartozó DB fájl táblázatban lévő összes DB fájlt (Fő fájl, csatolt fájlok és egyéb DB fájlok).
3. Definiálja a taszk adatnézetét a Fő fájlból, a csatolt fájlokból és a virtuális fájlból történő Select műveleteknek megfelelően. Az adatnézet korlátainak beállításához kiértékeli a Tartomány -tól -ig kifejezéseket.
4. Megnyitja az I/O fájlokat.
5. Ha nincs üzemmód váltás, végrehajtja a Taszk előfeldolgozás műveleti táblázatot.
6. Elvégzi a tervező által megadott rendezést, ha van ilyen.
7. Kikeresi az első adatnézet rekordot.
8. Megjelenít annyi rekordot, amennyit a taszk Fő ablak formátumán megadott Görgetési határok és Ismétlődő terület méretei megengednek.

**B: rekord inicializálása**

1. Behozza a rekordot. Ha nem talál rekordot a taszk adatnézetében, átvált beviteli módba.
2. Felépíti az adatnézet rekordot, felhasználva a fő és a csatolt fájl rekord mezőit valamint azokat a virtuális mezőket, amelyekhez tartozó Init kifejezéseket kiértékelte.
3. Ha Vége ellenőrzés = Előtte vagy Azonnal, akkor teszteli a Taszk vége feltételt. Ha ennek értéke Igaz, a motor az E: taszk lezárása fázisba lép át.
4. Végrehajtja a Rekord Előfeldolgozás műveleti táblázatot.

**C: felhasználói interakció**

A Rekord Főfeldolgozás műveleti táblázat végrehajtása

1. Gyors üzemmódban hajt végre minden olyan műveletet, amely megelőzi annak a mezőnek a Select műveletét, amelyen a kurzornak meg kell állnia.
2. Miközben a felhasználó mezőről mezőre lép az aktuális rekordon belül a megfelelő Select műveletek közé beágyazott műveleteket Lépéses vagy Gyors üzemmódban is végre lehet hajtani:

* Lépéses üzemmódban csak akkor, ha a végfelhasználó az Előző és a Következő mező műveletekhez hozzárendelt billentyűk segítségével vitte a kurzort az előző vagy a következő mezőre,
* Gyors üzemmódban minden más esetben, beleértve azt az esetet, amikor az egér segítségével kerül át a kurzor az előző vagy a következő mezőre.

1. A végfelhasználó az alábbi módokon zárhatja le a rekord interakciót:

* A Kilépés (Exit) parancsot választva;
* A Sortörlés parancsot választva;
* Az Opciók (Bevitel, Módosítás) egyikét választva;
* A következő műveletekhez rendelt billentyűk leütésével: Előző sor, Következő sor, Előző lap, Következő lap, Lap eleje, Lap vége, Táblázat eleje, Táblázat vége;
* A Következő mező művelethez rendelt billentyű leütésével abban az esetben, ha a kurzor az utolsó mezőn áll és a Taszkvezérlésben a Ciklikus Rekord Főfeldolgozás paraméter "Nem"-re lett állítva.

1. Amikor a felhasználó lezárja az aktuális rekordot:

* Ha a rekord módosult, vagy Utófeldolgozás mindig = Igen, vagy ha a program "Választó tábla" program és a rekord az ENTER leütésével lett lezárva, akkor a vezérlő motor Gyors üzemmódban hajtja végre a Rekord Főfeldolgozási táblázat hátralevő részét az aktuális mezőtől a táblázat végéig.

**D: rekord lezárása**

1. Csak akkor, ha a rekord módosult vagy Utófeldolgozás mindig = Igen:

* Végrehajtja a Rekord Utófeldolgozást Beviteli üzemmódban;
* A módosított adatnézeti rekordokat (a fő és a csatolt fájlokból) lemezre írja;
* Feltéve, hogy a taszk nem lekérdezési módban van, a módosított adatnézeti rekordokat (a fő és a csatolt fájlokból lemezre írja).

1. Ha a rekord törölve lett (vagyis ha a felhasználó a Sortörlés parancsot választotta, vagy ha a Rekord törlés feltétel értéke Igaz):

* Végrehajtja a Rekord Utófeldolgozást Törlés üzemmódban;
* Ha nem lekérdezési módban van, törli a fő fájl rekordot;

1. Ha Taszk vége = Utána vagy Azonnal, teszteli a Taszk vége feltételt.
2. Az alábbi szituációk esetén a taszk véget ér

* A felhasználó a Kilépés parancsot választotta;
* A Taszk vége feltétel kiértékelésekor az eredmény: igaz.

1. Ha a taszk valóban véget ért a vezérlés átkerül az E: taszk lezárása fázisra, egyébként visszalép a B: rekord inicializálása fázishoz.

**E: taszk lezárása**

1. Ha a felhasználó új működési módot választott, a vezérlés visszakerül az A: taszk inicializálás fázis 3. Lépeséhez. Minden más esetben:
2. Végrehajtja a Taszk Utófeldolgozást.
3. Bezárja az I/O fájlokat.
4. Bezárja az összes DB fájlt (fő és csatolt fájlok).
5. Bezárja és törli a rendezési fájlokat (ha van ilyen).
6. Kilép a taszkból.

##### Online taszk - beviteli mód

**A: taszk inicializálása**

1. A virtuális mezők a hívó taszktól kapnak paraméter értéket, ha van egyáltalán hívó taszk. A paraméter értéket nem kapott virtuális mezők értéke 0, ha a mező numerikus, üres, ha alfanumerikus és Hamis, ha logikai.
2. Megnyitja a taszkhoz tartozó DB fájl táblázatban lévő összes DB fájlt (fő fájl, csatolt fájlok és egyéb DB fájlok).
3. Definiálja a taszk adatnézetét a Fő fájlból, a csatolt fájlokból és a virtuális fájlból a Select műveleteknek megfelelően. Az adatnézet korlátainak beállításához kiértékeli a Tartomány -tól -ig kifejezéseket.
4. Megnyitja az I/O fájlokat.
5. Ha nincs üzemmód váltás, végrehajtja a Taszk Előfeldolgozás műveleti táblázatot.
6. Elvégzi a tervező által megadott rendezést, ha van ilyen.

**B: rekord inicializálása**

1. Felépíti az adatnézet rekordot:

* A valós mezők Init kifejezéseinek kiértékelésével;
* Az Init kifejezéssel nem rendelkező valós mezők alaphelyzetbe állításával;
* A virtuális mezők Init kifejezéseinek kiértékelésével;
* A csatolt rekordok behozásával.

1. Ha Vége ellenőrzés = Előtte vagy Azonnal, akkor teszteli a Taszk vége feltételt. Ha ennek értéke Igaz, a motor az E: taszk lezárása fázisba lép át.
2. Végrehajtja a Rekord Előfeldolgozás műveleti táblázatot.

**C: felhasználói interakció**

A Rekord Főfeldolgozás műveleti táblázat végrehajtása

1. Gyors üzemmódban hajt végre minden olyan műveletet, amely megelőzi annak a mezőnek a Select műveletét, amelyen a kurzornak meg kell állnia.
2. Miközben a felhasználó mezőről mezőre lép az aktuális rekordon belül, a megfelelő Select műveletek közé beágyazott műveleteket Lépéses vagy Gyors üzemmódban is végre lehet hajtani:

* Lépéses üzemmódban csak akkor, ha a végfelhasználó az Előző és a Következő mező műveletekhez hozzárendelt billentyűk segítségével vitte a kurzort az előző vagy a következő mezőre.
* Gyors üzemmódban minden más esetben, beleértve azt az esetet, amikor az egér segítségével kerül át a kurzor az előző vagy a következő mezőre.

1. A végfelhasználó az alábbi módokon zárhatja le a rekord interakciót:

* A Kilépés parancsot választva;
* A Sortörlés parancsot választva;
* Az Opciók (Bevitel, Módosítás) egyikét választva;
* A következő műveletekhez rendelt billentyűk leütésével: Előző sor, Következő sor, Előző lap, Következő lap, Lap eleje, Lap vége, Táblázat eleje, Táblázat vége;
* A Következő mező művelethez rendelt billentyű leütésével abban az esetben, ha a kurzor az utolsó mezőn áll és a Taszkvezérlésben a Ciklikus Rekord Főfeldolgozás paraméter "Nem"-re lett állítva.

1. Amikor a felhasználó lezárja az aktuális rekordot:

* Ha a rekord módosult, vagy Utófeldolgozás mindig = Igen, vagy ha a program "Választó tábla" program és a rekord az ENTER leütésével lett lezárva, akkor a vezérlő motor Gyors üzemmódban hajtja végre a Rekord Főfeldolgozási táblázat hátralevő részét az aktuális mezőtől a táblázat végéig.

**D: rekord lezárása**

1. Csak akkor, ha a rekord módosult vagy Utófeldolgozás mindig = Igen.

* Végrehajtja a Rekord Utófeldolgozást beviteli üzemmódban;
* A módosított adatnézeti rekordokat (a fő és a csatolt fájlokból) lemezre írja.

1. Ha Taszk vége = Utána vagy Azonnal, teszteli a Taszk vége feltételt.
2. Az alábbi szituációk esetén a taszk véget ér:

* A felhasználó a Kilépés parancsot választotta;
* A Taszk vége feltétel kiértékelésekor az eredmény Igaz.

1. Ha a taszk valóban véget ért, a vezérlés átkerül az E: taszk lezárása fázisra, egyébként visszalép a B: rekord inicializálása fázishoz.

**E: taszk lezárása**

1. Ha a felhasználó új működési módot választott, a vezérlés visszakerül az A: taszk inicializálás fázis 4. lépéséhez. Minden más esetben:
2. Végrehajtja a Taszk Utófeldolgozást.
3. Bezárja az I/O fájlokat.
4. Bezárja az összes DB fájlt (fő és csatolt fájlok).
5. Bezárja és törli a rendezési fájlokat (ha van ilyen).
6. Kilép a taszkból.

### Batch taszk folyamat

##### Batch taszk - módosítás (lekérdezés, törlés) mód

**A: taszk inicializálása**

1. A virtuális mezők a hívó taszktól kapnak paraméter értékeket, ha van egyáltalán hívó taszk. A paraméter értéket nem kapott virtuális mezők értéke 0, ha a mező numerikus, üres, ha alfanumerikus és Hamis, ha logikai
2. Megnyitja a taszkhoz tartozó DB fájl táblázatban lévő összes DB fájlt (fő fájl, csatolt fájlok és egyéb DB fájlok).
3. Definiálja a taszk adatnézetét a fő fájlból, a csatolt fájlokból és a virtuális fájlból a Select műveleteknek megfelelően. Az adatnézet korlátainak beállításához kiértékeli a Tartomány -tól -ig kifejezéseket.
4. Ha az Opciók engedélyezése = Igen, megjelenik a Végrehajtás indítása párbeszédablak. Ha a felhasználó a Nemet választja, a vezérlő motor áttér a C: taszk lezárása fázis 6. lépésére.
5. Elvégzi a tervező által megadott rendezést, ha van ilyen.
6. Megnyitja az I/O fájlokat.
7. Az összes virtuális mezőt alaphelyzetbe állítja.
8. A váltás-jelzőt (flag) bekapcsolja (on állásba).
9. Végrehajtja a Taszk Előfeldolgozás műveleti táblázatot.

**B: rekord végrehajtása**

1. Behozza a rekordot. Ha nem talál rekordot a taszk adatnézetében, a motor átlép a C: taszk lezárása fázisba.
2. Felépíti az adatnézet rekordot felhasználva

* A fő és a csatolt fájl rekord mezőit a behozott értékeikkel együtt
* A virtuális mezőket, a hozzájuk tartozó Init kifejezések kiértékelésének eredményeként adódó értékekkel.

1. Ha Abort = Igen és a felhasználó a Kilépés parancsot választja, a vezérlő motor átlép a C: taszk lezárása fázisba.
2. Ha Vége ellenőrzés = Előtte vagy Azonnal, akkor teszteli a Taszk vége feltételt. Ha ennek értéke Igaz a motor a C: taszk lezárása fázisba lép át.
3. Ha a váltás-jelző be van kapcsolva, a motor végrehajtja az összes alacsonyabb Váltási Előfeldolgozási műveletet, a magastól az alacsonyig.
4. Végrehajtja a Rekord Előfeldolgozás műveleti táblázatot.
5. Ha a Kezdeti üzemmód Lekérdezés, akkor a motor végrehajtja a Rekord Utófeldolgozást és folytatja a 10. lépéstől.
6. Ha a Kezdeti üzemmód nem Törlés

* A motor végrehajtja a Rekord Utófeldolgozást Módosítás módban;
* A módosult adatnézeti rekordokat (fő vagy csatolt fájlok) lemezre írja.

1. Ha Kezdeti üzemmód = Törlés vagy Rekordtörlés = Igen:

* Végrehajtja a Rekord Utófeldolgozást Törlés módban;
* Törli a rekordot a lemezről.

1. Ha Vége ellenőrzés = Utána vagy Azonnal, akkor teszteli a Taszk vége feltételt. Ha ennek értéke Igaz, a motor a C: taszk lezárása fázisba lép át.
2. Megvizsgálja a Váltási feltételt. Ha az értéke Igaz, bekapcsolja a váltás-jelzőt és végrehajtja az összes alacsonyabb Váltási Utófeldolgozást az alacsonytól a magasig.
3. A motor átlép a B: rekord végrehajtása fázisba.

**C: taszk lezárása**

1. Végrehajtja egyszer az összes Váltási Utófeldolgozást, az alacsonytól a magasig, feltéve, hogy legalább egy rekord fel lett dolgozva.
2. Ha a Form Lines be lett állítva a motor kitölti a kívánt számú részlet sort.
3. Végrehajtja a Taszk Utófeldolgozást.
4. Bezárja az I/O fájlokat.
5. Ha az Opciók engedélyezése = Igen, megjelenik az Ismételt végrehajtás párbeszédablak. Ha a felhasználó az Igen-t választja, a vezérlő motor visszatér az A: taszk inicializálása fázis végrehajtására a 6. lépéstől kezdve.
6. Bezárja az összes DB fájlt.
7. Bezárja és törli a rendezési fájlokat (ha van ilyen).
8. Kilép a taszkból.

##### Batch taszk - beviteli mód

**A: taszk inicializálása**

1. Megnyitja a taszkhoz tartozó DB fájl táblázatban lévő összes DB fájlt (fő fájl, csatolt fájlok és egyéb DB fájlok).
2. Definiálja a taszk adatnézetét a Fő fájlból a csatolt fájlokból és a virtuális fájlból a Select műveleteknek megfelelően. Az adatnézet korlátainak beállításához kiértékeli a Tartomány -tól -ig kifejezéseket.
3. Ha Opciók engedélyezése = Igen, megjelenik a Végrehajtás indítása párbeszédablak. Ha a felhasználó a Nem-t választja, a vezérlő motor áttér a C: taszk lezárása fázis végrehajtására az 5. lépéstől kezdve.
4. Elvégzi a tervező által megadott rendezést, ha van ilyen.
5. Megnyitja az I/O fájlokat.
6. Az összes virtuális mezőt alaphelyzetbe állítja.
7. Végrehajtja a Taszk előfeldolgozás műveleti táblázatot.

**B: rekord végrehajtása**

1. Ha Abort = Igen és a felhasználó a Kilépés parancsot választja, a vezérlő motor átlép a C: taszk lezárása fázisba.
2. Az összes valós változót inicializálja vagy alaphelyzetbe állítja.
3. Ha Vége ellenőrzés = Előtte vagy Azonnal, akkor teszteli a Taszk vége feltételt. Ha ennek értéke Igaz, a motor átlép a C: taszk lezárása fázisba.
4. Végrehajtja a Rekord Előfeldolgozás műveleti táblázatot.
5. Végrehajtja a Rekord Utófeldolgozást Beviteli módban.
6. A módosult adatnézeti rekordokat (fő vagy csatolt fájlok) lemezre írja.
7. Ha Rekordtörlés = Igen, akkor végrehajtja a Rekord Utófeldolgozást Törlés módban. Törli a fő fájl rekordot a lemezről.
8. Ha Vége ellenőrzés = Utána vagy Azonnal, akkor teszteli a Taszk vége feltételt. Ha ennek értéke Igaz, a motor a C: taszk lezárása fázisba lép át.
9. A motor visszatér a B: rekord végrehajtása fázis végrehajtásához.

**C: taszk lezárása**

1. Ha a Form Lines be lett állítva, a motor kitölti a kívánt számú részlet sort.
2. Végrehajtja a Taszk Utófeldolgozást.
3. Bezárja az I/O fájlokat.
4. Ha Opciók engedélyezése = Igen, megjelenik az Ismételt végrehajtás párbeszédablak. Ha a felhasználó az Igen-t választja, a vezérlő motor visszatér az A: taszk inicializálása fázis végrehajtására a 4. lépéstől kezdve.
5. Bezárja az összes DB fájlt.
6. Bezárja és törli a rendezési fájlokat (ha van ilyen).
7. Kilép a taszkból.