

Delphi programozás I.

Konzol alkalmazások készítése Delphiben

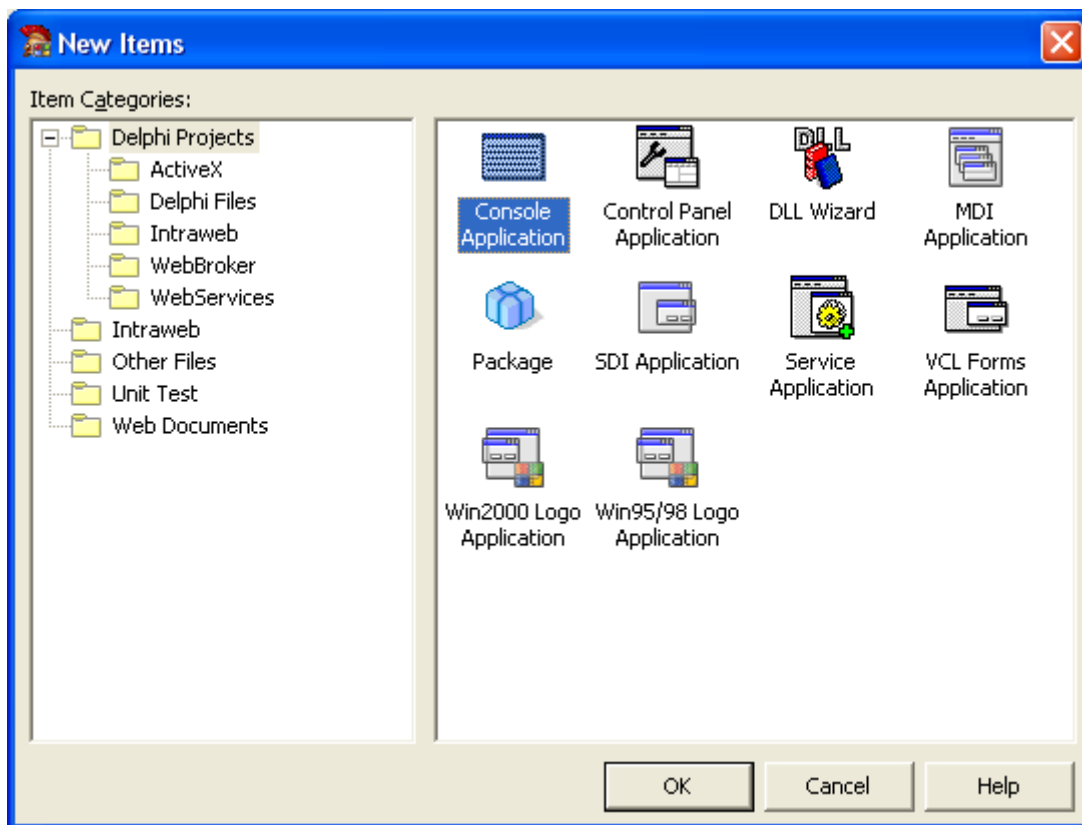
A Delphi konzol alkalmazása (console application) olyan 32 bites program, amely nem grafikus felületen, hanem egy szöveges konzol ablakban fut.

Néhány példa, amikor érdemes konzol alkalmazást készíteni:

1. Parancssorból meghívható programok készítése, pl. batch programozás során használható segédprogramok (későbbi téma).
2. Versenyfeladatok megoldásánál, amikor gyakran egy szöveges bemenet alapján egy szöveges kimenetet kell előállítani. Ilyenkor felesleges a grafikus felület használata.

Hogyan lehet létrehozni egy új console application-t?

A Turbo Delphiben a File/New/Other menüpontot kell meghívni. Ezután - a lenti ábra szerint - a Console Application ikont kell kiválasztani.



Ha lenyomjuk az OK-t, akkor a Delphi létrehoz egy programvázatot. A programszöveg érdekessége, hogy egy `{ $APPTYPE CONSOLE }` direktíva található benne, amellyel grafikus alkalmazások esetén nem találkozunk (az alapértelmezés: `{ $APPTYPE GUI }`).

```
program Project1;

{$APPTYPE CONSOLE}

uses
  SysUtils;

begin
  { TODO -oUser -cConsole Main : Insert code here }
end.
```

Ha ezt a rövid kis programot megpróbáljuk futtatni a fejlesztőkörnyezetből, akkor az Avast vírusnak érzékeli, és nem engedi elindítani :-). A forrást ezért kiegészítjük néhány sorral, és elkészítjük a klasszikus Hello World alkalmazás Delphi verzióját:

```
program Hello;

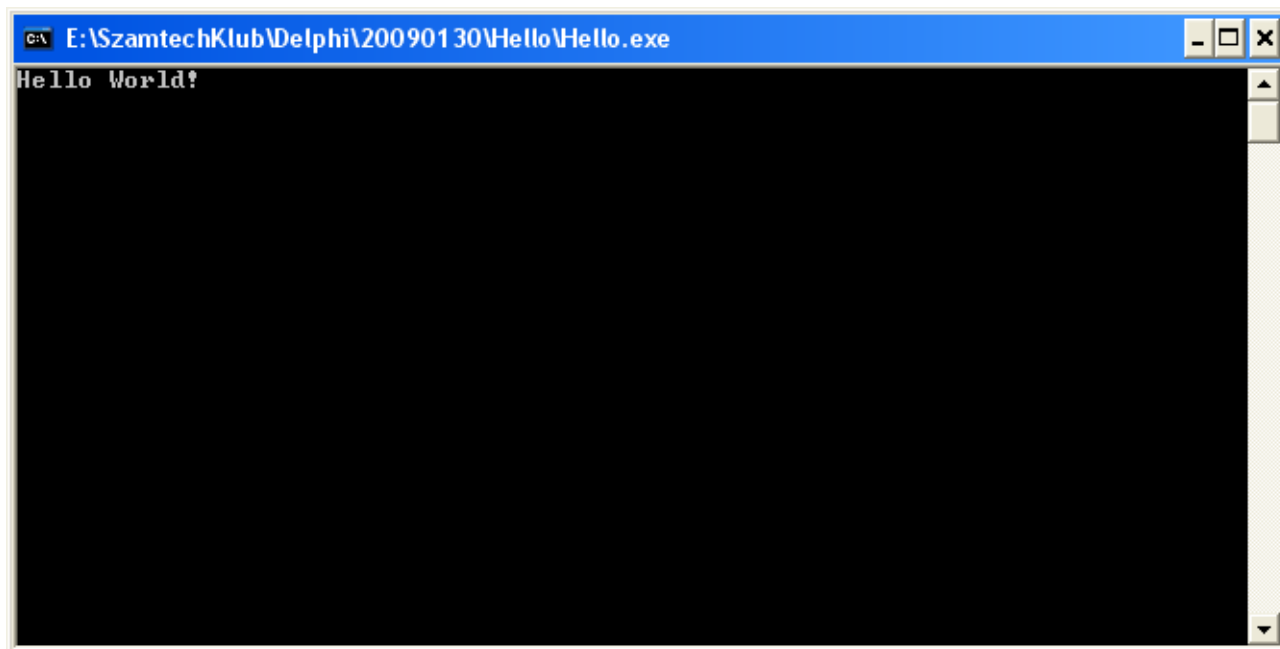
{$APPTYPE CONSOLE}

uses
  SysUtils;

begin
  // Hello World program
  Writeln('Hello World!');

  Readln;
end.
```

Ha futtatjuk, a program az alábbi ablakot jeleníti meg:



A program végén azért szerepel egy Readln utasítás, mert egyébként nem tudnánk megnézni a kimenetet, ui. a Delphi rögtön lecsukná a konzol ablakot.

Ha megnézzük a program könyvtárát, akkor a következőket vehetjük észre:

1. Nem találunk pas kiterjesztésű filet; a forrás egy dpr kiterjesztésű fileban található (pl. Hello.dpr)
2. Létrejött egy exe kiterjesztésű file (pl. Hello.exe)

Ha a Hello.exe programot nem a Delphi fejlesztőkörnyezetből akarjuk futtatni, hanem a parancssorból, akkor a következőket tehetjük:

1. el kell indítani a Windows parancssort (Programok / Kellékek / Parancssor vagy cmd.exe)
2. be kell menni a Delphi program könyvtárába (a cd utasítással)
3. be kell írni a parancssorba: Hello.exe
4. Enter

Ezután meglátjuk a képernyőn a program kimenetét. Ha parancssorban dolgozunk, akkor felesleges a program végén található Readln utasítás, ui. a parancssor nem záródik le automatikusan.

A bemenet és a kimenet átirányítása

Készítsünk egy programot, amely a billentyűzeten beírt sorokat kiegészíti egy kezdő és egy záró # jellel.

```
program IOproba;
{$APPTYPE CONSOLE}
uses
  SysUtils;
var s : string;
begin
  while not eof do
    begin
      Readln(s);
      Writeln('#' + s + '#');
    end;
end.
```

Ha elindítjuk a programot, akkor az a billentyűzeten begépelte sorokat kiegészíti a kezdő és záró # jellel. A kilépés a Ctrl + Z billentyűvel lehetséges.

Ha parancssorból futtatjuk a programot, akkor az inputot fileból is vehetjük. Legyen például az a.txt szövegfile tartalma:

```
1
2
3
4
```

A parancssorba a következőket kell írni:

```
ioproba.exe < a.txt
```

Ekkor a képernyőn látható kimenet az alábbi lesz:

```
#1#  
#2#  
#3#  
#4#
```

A kimenetet is átirányíthatjuk, mégpedig a következő módon:

```
ioproba.exe < a.txt > b.txt
```

Ebben az esetben nem látunk semmit a képernyőn, és a kimenet a b.txt fileba kerül.

Megjegyzés: az átirányítás ezen módját természetesen más programoknál is alkalmazhatjuk.

Parancssori paraméterek kezelése

A parancssorban a meghívott programoknak gyakran paramétereket is adhatunk.

Nézzük meg a következő esetet:

```
program param;  
  
{$APPTYPE CONSOLE}  
  
uses  
    SysUtils;  
  
var i : integer;  
  
begin  
    Writeln('A paraméterek száma: ', ParamCount);  
  
    Writeln('A program neve: ', ParamStr(0));  
  
    if (ParamCount>0) then  
    begin  
        Writeln('A megadott paraméterek:');  
  
        for i:=1 to ParamCount do  
            Writeln(ParamStr(i));  
    end;  
end.
```

A program a ParamCount és a ParamStr függvényeket használja. A ParamCount a parancssori paraméterek számát adja meg. A ParamStr pedig a megadott pozíción lévő paramétert, vagy egy

üres sztringet, ha nem volt ilyen paraméter. A ParamStr(0) Windows alatt a futtatott program útvonalát és nevét adja vissza.

Ha a következő parancsot futtatjuk:

```
param.exe 1 2 abc "a b c"
```

akkor a kimenet az alábbi lesz:

```
A parameterek szama: 4
A program neve: ...\\param.exe
A megadott parameterek:
1
2
abc
a b c
```

Látható, hogy a paramétereket szóközzel kell egymástól elválasztani. Ha olyan paramétert szeretnénk átadni, amely szóközt is tartalmaz, akkor azt "" jelek közé kell tenni.

A fejlesztés során hasznos lehet a következő lehetőség: a Turbo Delphi Run / Parameters menüpontjában a Parameters mezőbe be lehet írni azokat a paramétereket, amelyekkel meg szeretnénk hívni a programot. Így nincs arra szükség, hogy a lefordított exe-t külön, parancssorból kelljen futtatni.

Versenyfeladatok

Ha versenyfeladatokat szeretnénk megoldani, akkor a következő címeken érdemes próbálkozni:

<http://acm.uva.es/problemset/> (innen elköltözött az oldal, de találhatunk még hasznos infókat)

illetve

<http://icpcres.ecs.baylor.edu/onlinejudge/index.php>

Ezen a címen több száz érdekes programozási problémát találhatunk, melyek megoldásához nincs szükség a grafikus felület használatára.