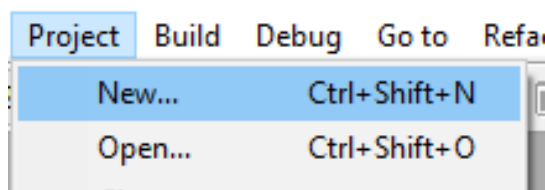
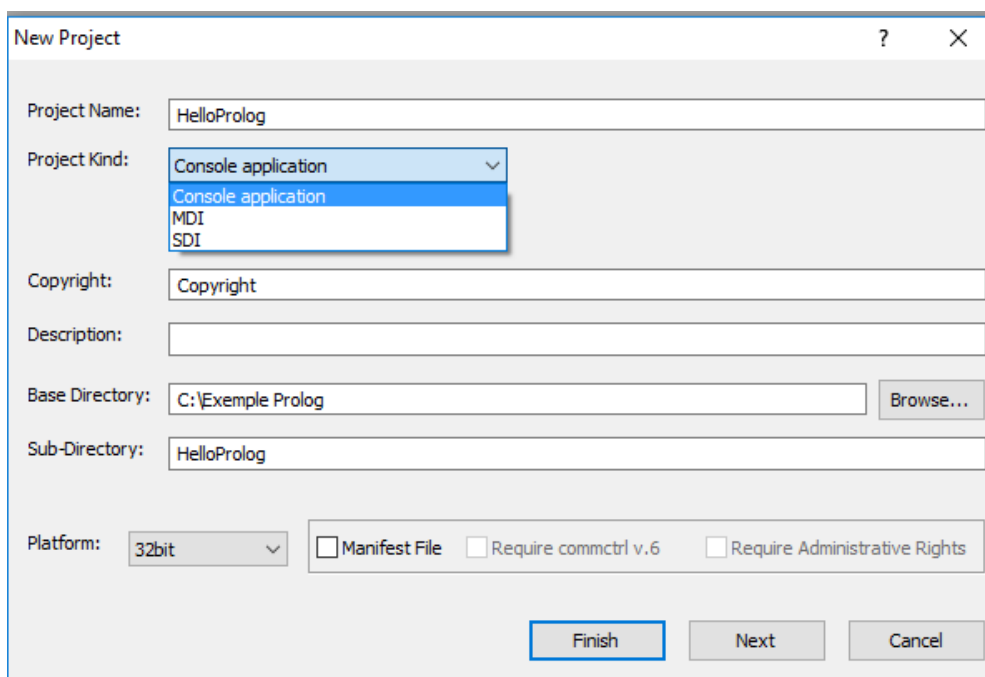


Aplicații consolă în Visual Prolog 7.5
Crearea unui nou proiect de tip consolă
<https://visual-prolog.software.informer.com/7.5/>

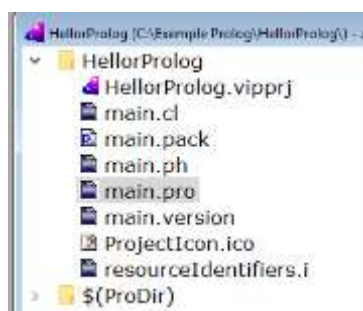
Selectați din meniu **Project -> New**



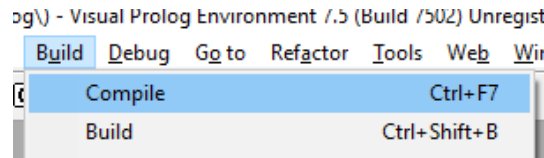
Selectați opțiunea **Console Application** și dați un **nume** sugestiv proiectului:



În acest moment avem generate următoarele fișiere:



Dacă acestea nu apar (depinde de versiunea de Visual Prolog care rulează) s-ar putea să aveți nevoie să compilați proiectul (**meniul Build**)



În acest moment ne interesează doar fișierul **main.pro**, care conține implementarea clasei main și scopul care **rulează clauza run**.

```
main.pro - zoom: 160 %
% Copyright

implement main
  open core

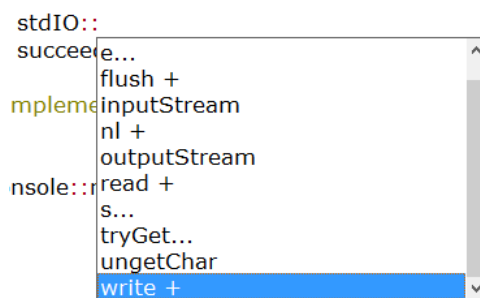
clauses
  run() :-
    succeed. % place your own code here

end implement main

goal
  console::runUtf8(main::run).
```

2.2 Afișarea în consolă

Visual Prolog oferă o serie de clase predefinite care permit o serie de operații utile. De exemplu **afișarea se face** utilizând **metoda write din clasa stdIO**. Scriind în cod secvența "stdIO::" va apărea lista de metode:



Din aceasta listă selectăm opțiunea write:

Exercițiu:

Să se afișeze mesajul "Hello Prolog !"

```
implement main           // implicit în Prolog
  open core              // implicit în Prolog

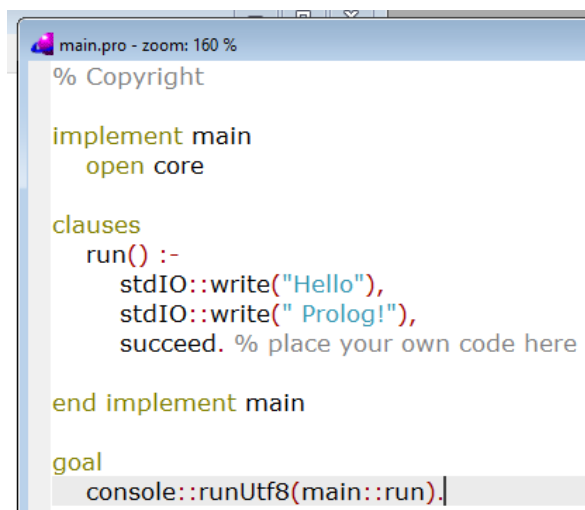
  clauses                // implicit în Prolog
    run():-              // implicit în Prolog
      console::init(),   // initializez consola
      stdIO::write("Hello"), // afișare în Prolog
      stdIO::write(" Prolog!"), // afișare în Prolog
      succeed.

end implement main       // implicit în Prolog

goal                    // implicit în Prolog
  mainExe::run(main::run). // implicit în Prolog
```

Aceste metode reprezintă predicate speciale care returnează tot timpul adevărat, deci nu influențează executarea regulilor în care apar.

O versiune similară a programului ar fi următoarea:



```
main.pro - zoom: 160 %
% Copyright

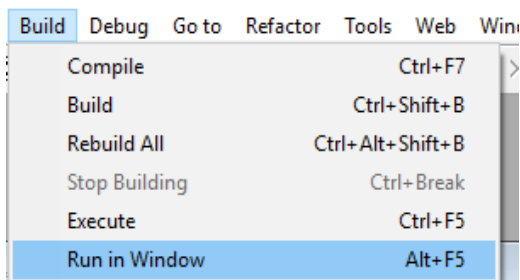
implement main
  open core

  clauses
    run() :-
      stdIO::write("Hello"),
      stdIO::write(" Prolog!"),
      succeed. % place your own code here

end implement main

goal
  console::runUtf8(main::run).
```

Rularea se face prin selectarea Run în Window:



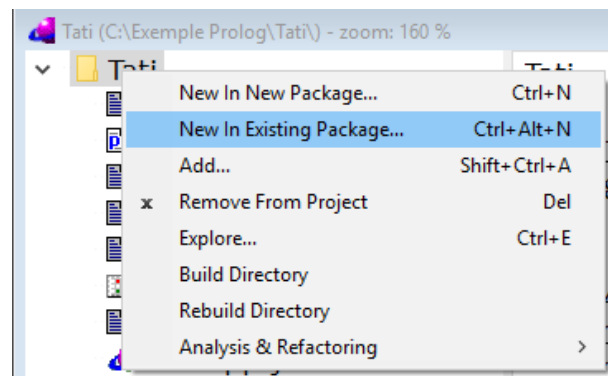
Rezultatul fi de forma următoare:

```
C:\WINDOWS\system32\cmd.exe

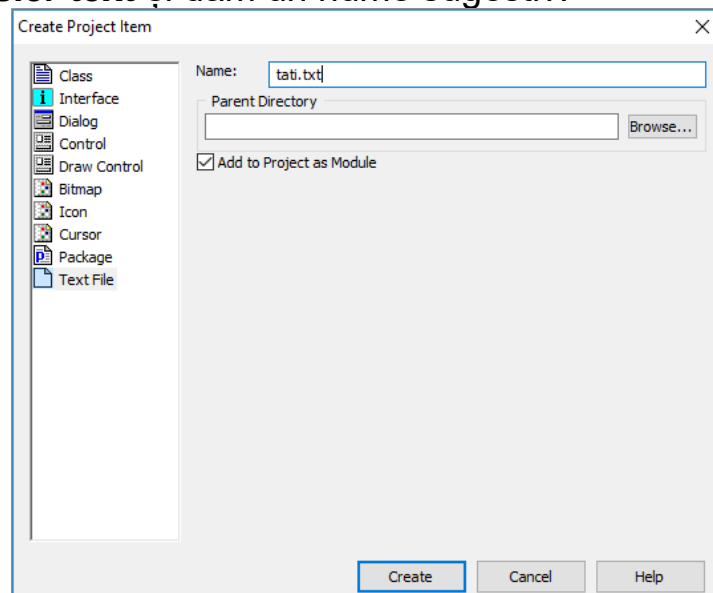
C:\Exemple Prolog\HellorProlog\Exe>"C:\Exe
Hello Prolog!
C:\Exemple Prolog\HellorProlog\Exe>pause
Press any key to continue . . .
```

Vom porni un proiect nou de consolă.

Adăugăm un nou fișier în proiect, fișier care va conține faptele dorite: (.txt)



Selectăm opțiunea **fișier text** și dăm un nume sugestiv:



Adăugăm în acest **nou fișier** faptele dorite:

```
tati.txt - zoom: 160 %

clauses
  tata("Alex","Gabriel").
  tata("Ion","Gabriel").
  tata("Gabriel","Razvan").
  tata("Monica","George").
```