Pregătirea mediului de dezvoltare

Visual Studio Code este un editor pe care putem să îl folosim pentru a crea programe în OCaml. Extensia pentru OCaml pentru Visual Studio Code împreună cu pachetele OCaml merlin și ocp-indent oferă o experiență de dezvoltare foarte bună, atât pe Windows cât și pe Linux.

Notă: Visual Studio Code este un editor foarte utilizat la momentul actual. El poate fi folosit pentru dezvoltarea de programe în diferite limbaje și medii de programare (ex de limbaje: HTML, Javascript, C#, Python, Java). Familiarizarea cu acest editor poate fi folositoare și pe viitor.

Instalarea pe Windows

Pentru a vedea pașii, puteți să urmăriți video-ul .(S-ar putea să observați mici diferențe, video-ul a fost înregistrat anul trecut și unele versiuni diferă.)

Pașii pentru a instala pe Windows 10 sunt următorii:

Instalați OPAM (managerul de pachete pentru OCaml)

De ce este necesar ?

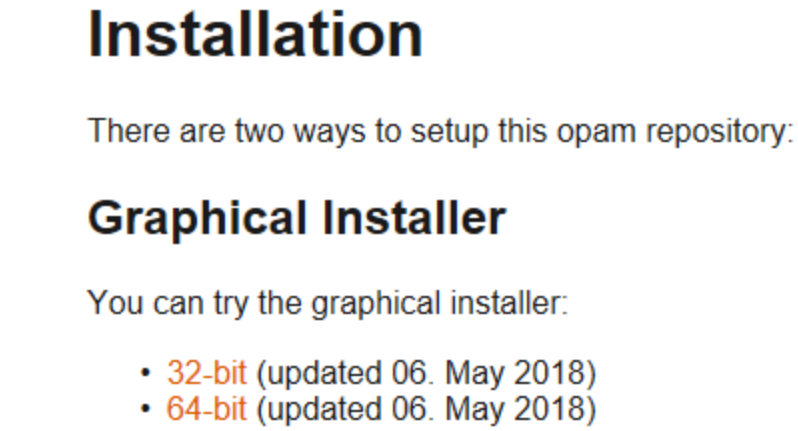
OPAM este un manager de pachete, adică un program care permite instalarea de programe utilitare dintr-o bibliotecă de utilitare (fie sub formă de programe, fie sub formă de alte biblioteci), pe care o administrează echipa OPAM. Acest model de a accesa ecosistemul unui limbaj este foarte comun în ziua de azi. Majoritatea limbajelor/framework-urilor au un astfel de sistem de a instala utilitare, de exemplu:

Pentru Javascript avem `npm` și `yarn`
- Pentru .NET avem `nuget`
- Pentru Python avem `pip`
- Pentru Java avem `maven`

OPAM va instala automat atât interpretorul cât și compilatorul OCaml, pe care le vom folosi pentru a rula comenzi OCaml și pentru a compila programe OCaml din fișierele de cod sursă.

Cum?

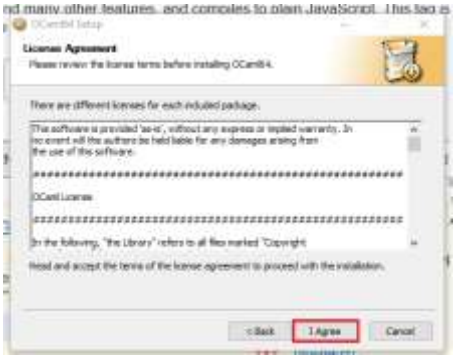
Mergeți la această adresă http://fdopen.github.io/opam-repository-mingw/installation/ și descărcați versiunea grafică de utilitare de instalare:



Alegeți 32/64 de biți, conform cu sistemul vostru de operare. Dacă nu știți care versiune o aveți, alegeți cea pe 64 de biți, dacă nu este cea bună, va apărea o eroare la instalare și atunci încercați varianta pe 32 de biți (Sau consultați acest articol pentru a afla versiunea). După ce s-a descărcat programul, rulați-l.

Se va deschide un program care vă va ghida prin instalarea OPAM. Apăsați ”Next”

Citiți acordul de licență și apăsați pe butonul „I agree” (ca semn al faptului că acceptați acordul de licență).



Următorul ecran vă permite să selectați calea unde va fi instalat programul. Dacă nu aveți experiență in instalarea de unelte de dezvoltare, este recomandabil să păstrați calea pre-selectată deoarece în continuarea acestui document vom presupune că ați instalat programul la calea implicită. În cele ce

urmează, dacă ați modificat calea, să țineți cont de acest fapt, înlocuind calea implicită cu calea selectată de voi.

Apăsați pe butonul „Install” pentru a porni instalarea (nu deconectați calculatorul de la internet pe durata procesului de instalare).

Utilitarul de instalare va descărca și instala Cygwin (https://www.cygwin.com/), care este „o colecție de unelte GNU și Open Source care oferă o funcționalitate similară cu distribuție Linux pe Windows”. Acesta este necesar pentru a rula OCaml sub Windows.

După finalizarea instalării, va apărea următorul mesaj:

Asigurați-vă că mesajul este unul de succes. Dacă mesajul conține cuvântul „error”, probabil că a apărut o problemă. Încercați să repetați instalarea dacă a apărut o eroare. Dacă în continuare nu reușiți să instalați cu succes OPAM, consultați un asistent de laborator pentru ajutor.

Instalarea Merlin și Ocp-indent

De ce este necesar?

Merlin și ocp-indent sunt două utilitare pe care le vom instala cu ajutorul managerului de pachete OPAM.

Merlin are capacitatea de a analiza cod sursă OCaml și de a oferi informații cu privire la erorile dintr-un fișier. De asemenea, poate oferi sugestii cu privire la ce este posibil să se scrie în mod valid mai departe.

Ocp-indent are capacitatea de a formata un fișier OCaml.

Ambele utilitare se vor integra cu mediul de dezvoltare pe care îl vom instala mai târziu.

Cum?

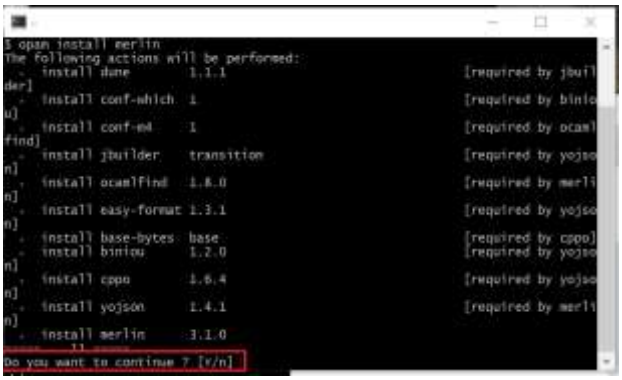
Asigurați-vă că aveți o conexiune activă la internet (instalarea va necesita descărcarea unor fișiere de către utilitarul care face instalarea).

Deschideți terminalul Bash folosind shortcut-ul OCaml64 care a apărut pe desktop. (Ce este un terminal? Vedeți aici)

Rulați comanda de a instala Merlin (scrieți exact textul de mai jos și apăsați tasta „Enter”):

```
opam install merlin
```

Veți fi întrebați dacă doriți să instalați Merlin împreună cu utilitarele de care are Merlin nevoie:



Apăsați tasta „Enter” și procesul de instalare va porni. Ar trebui să vedeți ceva asemănător cu:

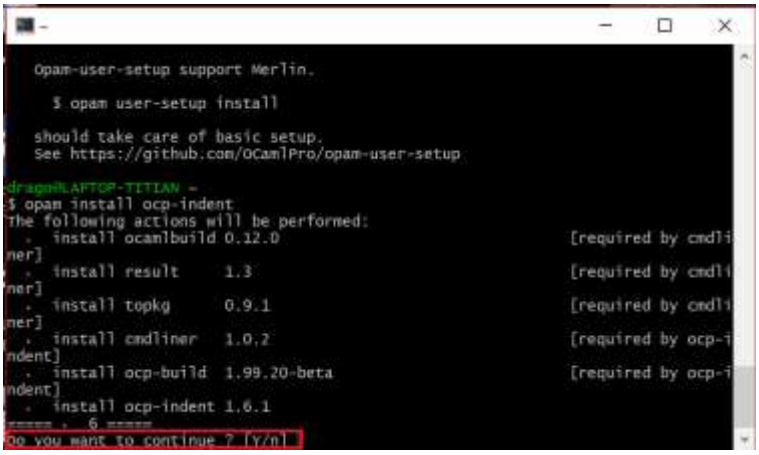
După instalare, va trebui să ne asigurăm că aceasta s-a întâmplat cu succes.

Verificați că în terminal nu a apărut textul „error”, și a apărut un text de forma `merlin-[versiune] installed successfully.` S-ar putea să trebuiască să faceți scroll în sus pentru a vedea textul.

Rulați apoi în același terminal (în aceeași fereastră) comanda de a instala ocp-indent (scrieți exact textul de mai jos și apăsați tasta „enter”):

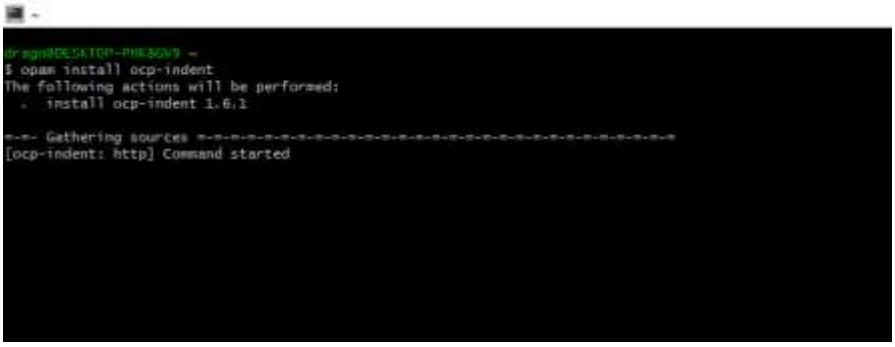
```
opam install ocp-indent
```

Veți fi întrebați dacă doriți să instalați ocp-indent, împreună cu utilitarele de care are ocp-indent nevoie:




```
```

Apăsați tasta „Enter” și instalarea va începe. Ar trebui să vedeți ceva asemănător cu:



După ce instalarea s-a finalizat, verificați că nu a apărut o eroare în terminal și că în terminal a apărut textul `ocp-indent.[versiune] installed successfully`.

Dacă au apărut erori, încercați să rulați comanda care a dat erori din nou. E posibil ca, dacă conexiunea la internet este întreruptă, instalarea să nu reușească. Dacă întâmpinați în continuare probleme, consultați un asistent de laborator.

Modificarea variabilelor de sistem

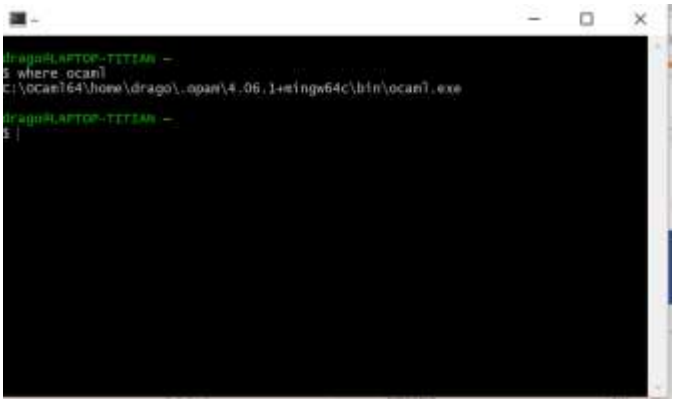
De ce este necesar?

Pentru ca mediul nostru de dezvoltare (Visual Studio Code) să găsească ușor interpretorul OCaml, trebuie să adăugăm calea unde a fost instalat acesta, acolo unde sistemul de operare caută programe (variabila de sistem PATH). De asemenea, pentru a putea rula OCaml din terminalul de Windows, trebuie să spunem interpretorului unde să găsească librăriile OCaml. (Notă: Interpretorul va funcționa din terminalul Bash care a fost instalat odată cu OPAM, terminalul pe care l-am și folosit să instalăm utilitarele, dar nu va funcționa din terminalul de Windows fără a seta variabila de sistem (OCamlLIB).

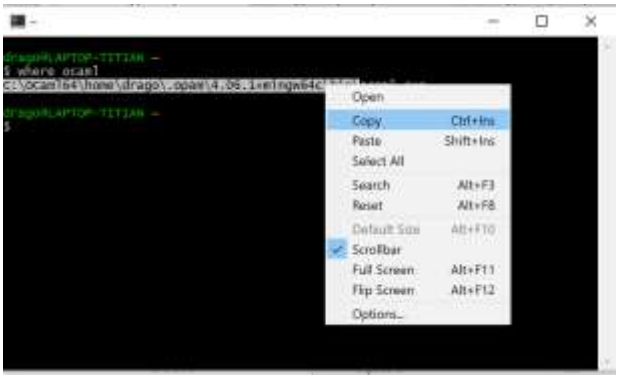
Cum?

Va trebui mai întâi să găsim calea unde este instalat OCaml. Deschideți din nou terminalul (dacă nu l-ați închis de la pașii anteriori îl puteți refolosi; dacă l-ați închis, redeschideți-l, folosind scurtătura de pe desktop) și scrieți comanda

```
where ocaml
```



În terminal va apărea calea unde se găsește instalat OCaml. Puteți să copiați calea din terminal, selectând cu mouse-ul, făcând click dreapta și selectând opțiunea „Copy”.



Partea care ne interesează este calea fără numele executabilului OCaml (adică fără textul de la sfârșitul căii `\ocmal.exe`)

De exemplu, pe sistemul pe care am rulat comenzile de mai sus, avem:

Calea unde e instalat OCaml:

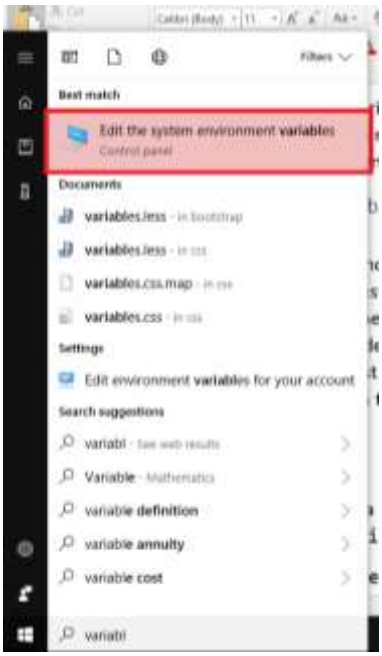
C:\OCaml64\home\drago\.opam\4.06.1+mingw64c\bin\OCaml.exe

Calea pe care o vom folosi mai departe:

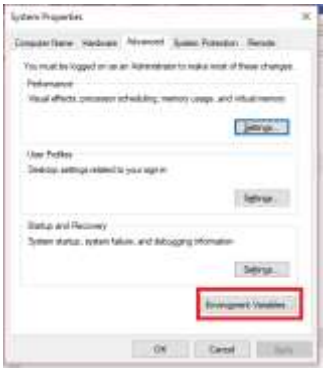
C:\OCaml64\home\drago\.opam\4.06.1+mingw64c\bin\

În funcție de sistemul pe care faceți instalarea, numele utilizatorului vostru și versiunea curentă de OCaml, părțile din cale care sunt marcate cu galben pot să difere.

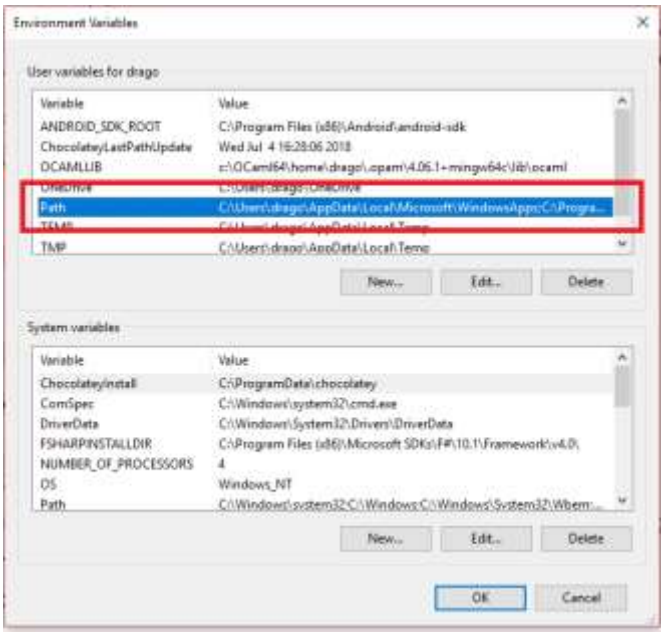
Calea pe care am găsit-o în paragraful anterior o vom adăuga acum la variabila de sistem PATH. Vom începe prin a deschide proprietățile sistemului. Există mai multe moduri de a face acest lucru. Cel mai simplu este să apăsați tasta de Windows (⊞) și să scrieți textul „variable” și să selectați prima opțiune care apare:



Din dialogul care se deschide, selectați opțiunea „Enviroment Variables”.

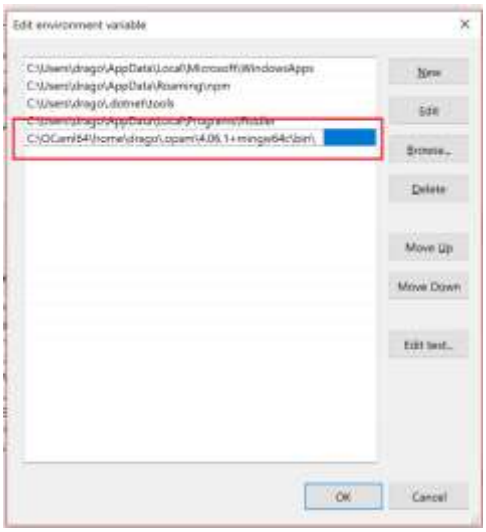


În fereastra care s-a deschis, căutați în prima secțiune variabila „PATH” și faceți dublu click pe această variabilă.



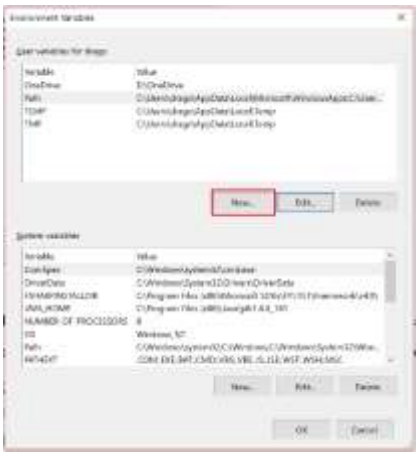
În fereastra care se deschide, apăsați pe butonul „New”:

Introduceți calea către OCaml, pe care am găsit-o mai devreme:



Apăsați „OK” pentru a salva modificările și a închide fereastra.

Mai trebuie să adăugăm o nouă variabilă de sistem OCamlLIB. Pentru aceasta, apăsați pe butonul „New” în fereastra de „Environment Variables”.



În fereastra care apare, introduceți în câmpul „Variable name” textul `OCamlLIB.` În câmpul „Variable value” introduceți calea către OCaml pe care am găsit-o anterior, dar înlocuind terminația `\bin` a căii cu `\lib\OCaml`.

De exemplu:

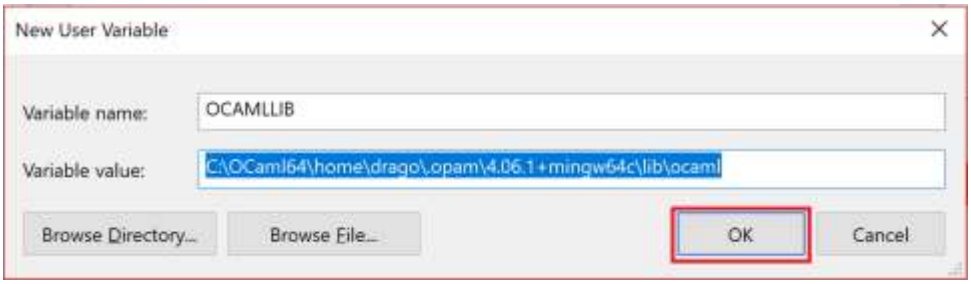
Dacă calea către OCaml este

`C:\OCaml64\home\drago\.opam\4.06.1+mingw64c\bin\`

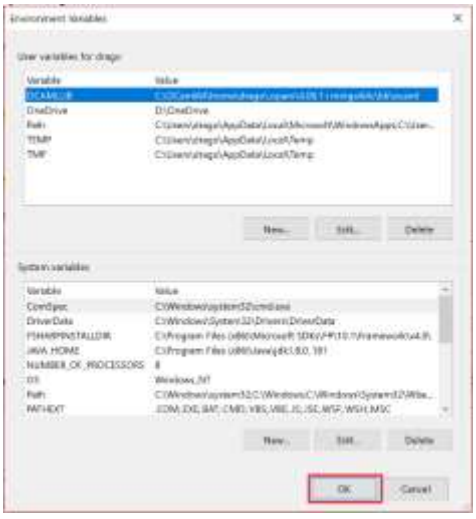
Valoarea pentru OCamlLIB va fi

`C:\OCaml64\home\drago\.opam\4.06.1+mingw64c\lib\OCaml`

După ce ați introdus valorile, apăsați pe butonul „OK”.

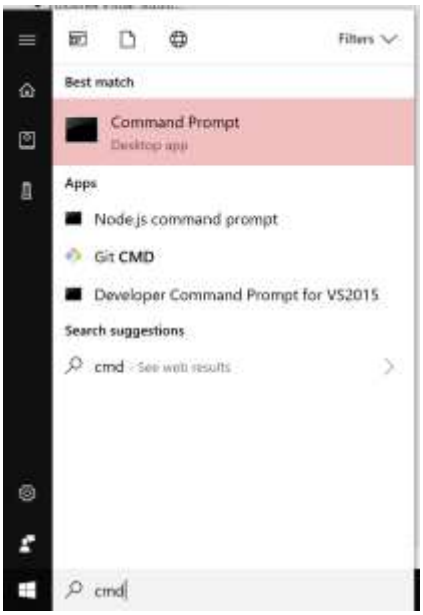


În fereastra „Environment Variables” apăsați pe „OK” pentru a salva modificările la variabilele de sistem.

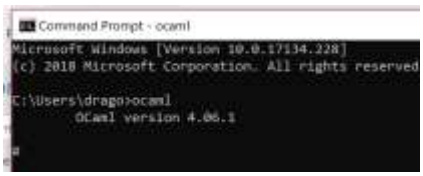


Testarea instalării OCaml

Deschideți un terminal Windows. Apăsați tasta de Windows (⊞) și scrieți `cmd` și apăsați tasta „Enter”.



În terminalul care se deschide scrieți comanda `OCaml`:



O instalare corectă ar trebui să afișeze versiunea de OCaml **și un sigur # pe ultima linie.**

Erori comune

```
C:\Users\drago>ocaml
'ocaml' is not recognized as an internal or external command,
operable program or batch file.
```

Nu ați setat corect variabila de sistem `PATH`. Repetați pașii de setare a variabilei PATH. Asigurați-vă că aveți calea către executabilul OCaml corectă. De asemenea, terminalul de Windows (cel deschis cu `cmd`) trebuie să fie deschis **DUPĂ** setarea variabilei de sistem. Încercați să reporniți sistemul dacă eroarea persistă.

```
C:\Users\drago>ocaml
        OCaml version 4.06.1

File "command line", line 1:
Error: Unbound module Pervasives
```

Nu ați setat corect variabila de sistem OCamlLIB. Repetați pașii de setare a variabilei OCamlLIB. Asigurați-vă că este corectă calea către librăriile OCaml. De asemenea, terminalul de Windows (cel deschis cu `cmd`) trebuie să fie deschis **DUPĂ** setarea variabilei de sistem.

Dacă apar alte probleme, sau nu reușiți să rezolvați una din problemele de mai sus, adresați-vă unui asistent de laborator.

Instalați Visual Studio Code pentru Windows

De ce este necesar?

Un mediu de dezvoltare integrat este foarte folositor pentru productivitatea unui programator. Majoritatea mediilor de dezvoltare moderne oferă o serie de facilități care fac scrierea și depanarea codului mai plăcute și mai eficiente:

- Colorare de sintaxă (cuvintele cheie, valorile, si alte texte cu semnificație specială sunt afișate cu alte culori)
- Sugestii de cod (mediul de dezvoltare oferă o listă de sugestii cu privire la cod valid pe care îl putem scrie)
- Indentare automată (codul va fi automat aliniat conform cu standardele limbajului)
- Compilarea codului din mediul de dezvoltare (mediul de dezvoltare va invoca compilatorul specific limbajului fără să îl rulăm noi manual)
- Depanarea codului (mediul de dezvoltare ne va permite să rulăm programul pas cu pas și să inspectăm starea curentă a programului)

Toate facilitățile de mai sus sunt oferite, printre altele, și de Visual Studio Code.

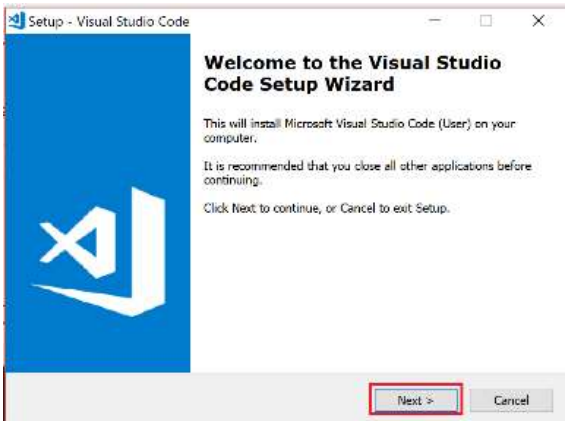
Visual Studio Code este un editor gratuit, dezvoltat de Microsoft, care rulează pe sistemele de operare Windows, Linux și MacOS. Apărut în 2015, a crescut foarte repede în popularitate. În 2018, un sondaj al

utilizatorilor de stackoverflow.com (un site folosit de programatori pentru a dezbate întrebări tehnice) a găsit că 34.9% (din cei 75398 de respondenți la sondaj) folosesc în mod curent acest editor (https://insights.stackoverflow.com/survey/2018/)

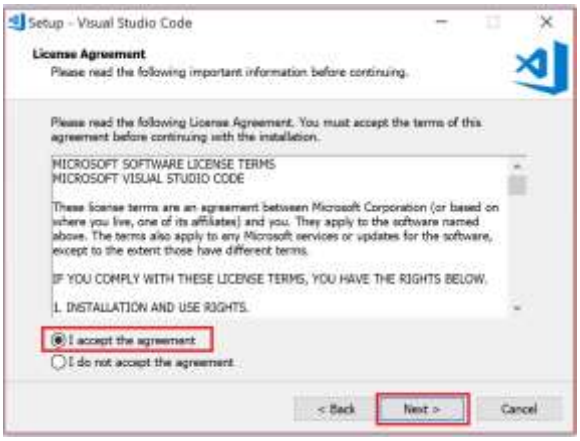
Cum?

Mergeți la pagina de descărcare Visual Studio Code (https://code.visualstudio.com/download) și descărcați versiunea corectă pentru sistemul vostru (64/32 de biți).

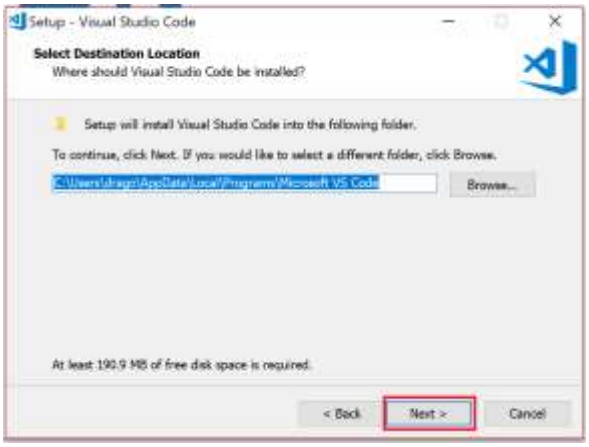
După ce s-a descărcat programul, rulați-l. Primul pas din instalare este doar de bun venit. Puteți să apăsați „Next”.



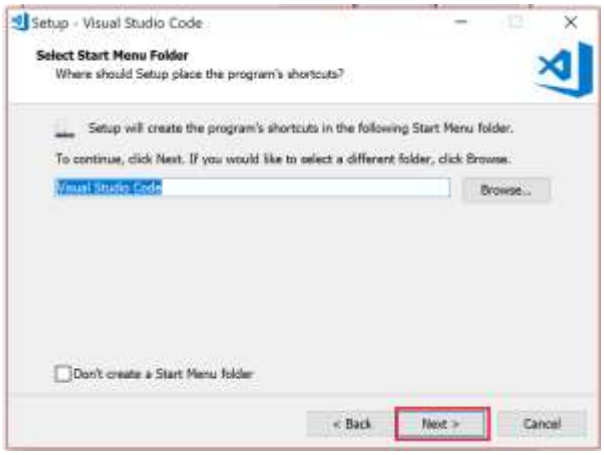
Următorul pas conține condițiile legale. Dacă doriți să folosiți programul, va trebui să acceptați condițiile de utilizare și să apăsați butonul „Next”.



Următorul pas vă permite să modificați calea unde se instalează Visual Studio Code, recomandat este să lăsați valoarea implicită. Apăsați butonul „Next”.

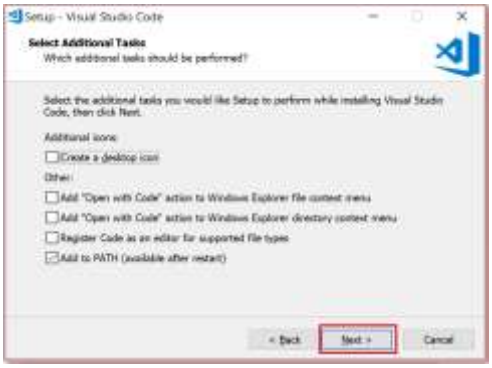


Următorul pas configurează intrarea în meniul de start pentru Visual Studio Code. Din nou, lăsăm opțiunile implicite și apăsăm butonul „Next”.

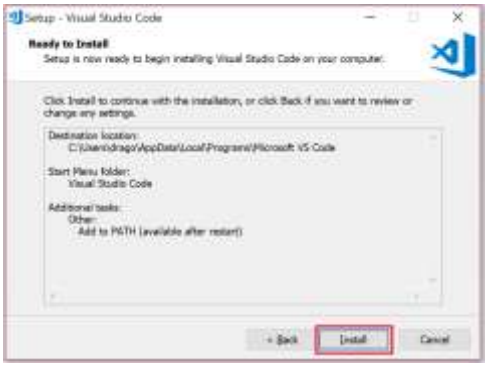


Pasul următor permite configurarea icoanelor create, configurarea fișierelor deschise de editor, precum și configurarea integrării Visual Studio Code cu „Explorer” (adică dacă în meniul de context din Explorer apar opțiuni legate de Visual Sudio Code). Pentru a simplifica accesul la program, putem bifa opțiunea

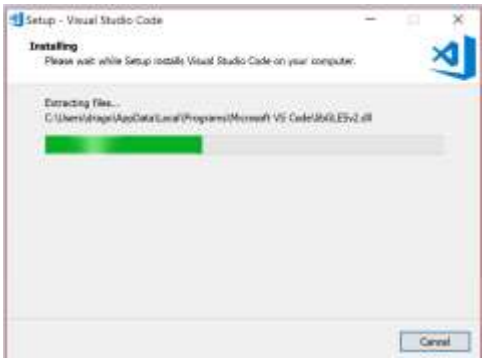
„Create a desktop icon”, care va determina programul de instalare să creeze o scurtătură pe desktop la Visual Studio Code. După ce am selectat opțiunile dorite, apăsăm butonul „Next”.



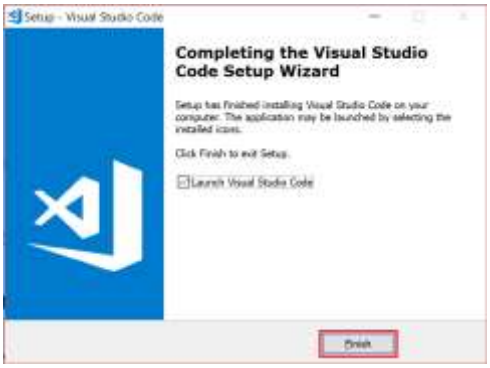
În acest pas, vedem un sumar al opțiunilor selectate. Verificăm că opțiunile sunt cele dorite și apăsăm pe butonul „Install”, care va porni instalarea propriu-zisă a programului.



Așteptăm să se finalizeze instalarea.



Ultimul pas ne anunță dacă Visual Studio Code a fost instalat cu succes. Apăsăm butonul „Finish” pentru a finaliza instalarea. În acest moment, ar trebui să se și deschidă programul instalat.



Instalați extensia pentru OCaml în Visual Studio Code

De ce este necesar?

Un editor nu poate oferi suport pentru toate limbajele de programare existente. Visual Studio Code vine cu suport integrat pentru Javascript și Typescript și oferă posibilitatea de a instala extensii (sau plug-in, ce e o extensie, citiți aici) care adaugă suport pentru alte limbaje. Pentru a avea o experiență de dezvoltare productivă, va trebui să instalăm o extensie pentru OCaml.

Cum?

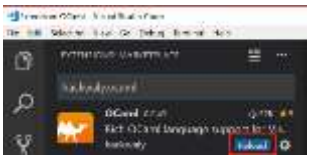
Deschideți Visual Studio Code și faceți click pe butonul pentru extensii:

.

În panelul care a apărut, în câmpul pentru căutare introduceți textul: `hackwaly.OCaml`

Apăsați pe butonul de „Install” al primului rezultat din listă.

După ce extensia a fost instalată, butonul „Install” se va transforma în butonul „Reload”. Pentru a încărca noua extensie în editor. Apăsați acest buton:



Reveniți la panelul pentru explorarea de fișiere.

Configurarea sugestiilor și erorilor

De ce este necesar?

După cum am menționat, un mediu de dezvoltare poate să ne ajute foarte mult în scrierea de cod corect, atât prin sublinierea codului incorect cât și prin sugestii cu privire la codul pe care putem să îl scrie (cunoscut în domeniu sub denumirile de „code completion” și „inteli-sense”).

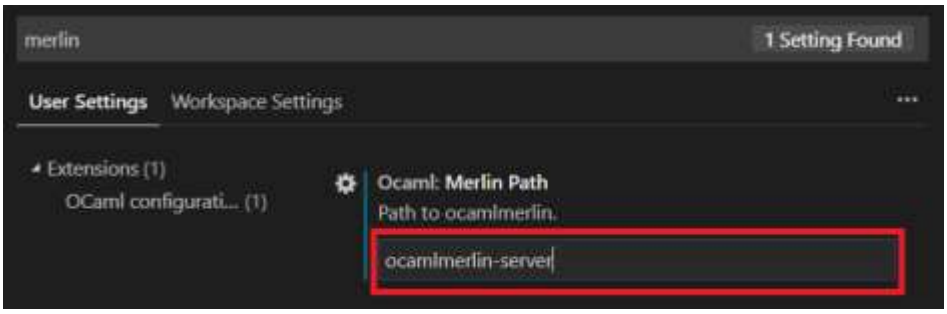
Visual Studio Code nu are inclusă această funcționalitate în editor, ci trebuie să folosim o extensie (pe care l-am instalat deja) împreună cu un program extern (Merlin, pe care l-am instalat deja cu managerul de pachete OPAM). În acest pas, vom lega programul extern de mediul de dezvoltare.

Cum?

Mergeți la `File → Preferences → Settings`

În panelul care se deschide, scrieți în câmpul de căutare textul `merlin`.

Modificați setarea înlocuind textul `OCamlmerlin` cu `OCamlmerlin-server` :



Scurtături din tastatura / Keyboard shortcut

De ce este necesar?

De multe ori vrem să rulăm aceeași comandă de mai multe ori, fără să o căutăm. Pentru aceasta, putem folosi scurtături de la tastatură (keyboard shortcuts).

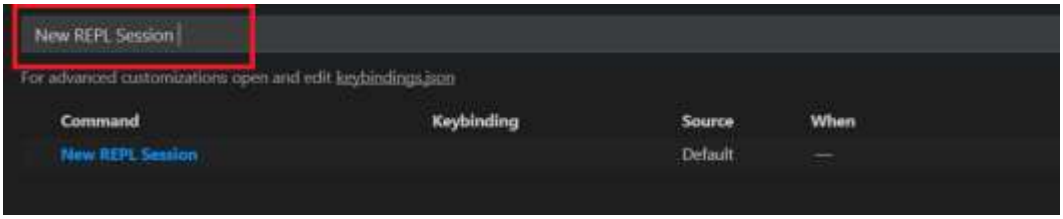
În cazul nostru, comenzile de care avem nevoie pentru OCaml, care nu au deja scurtături asociate, sunt:

- New REPL Session (care deschide un nou terminal cu interpretorul OCaml pornit);
- Send File to REPL Session (care trimite conținutul fișierului curent spre a fi interpretat de interpretorul OCaml).

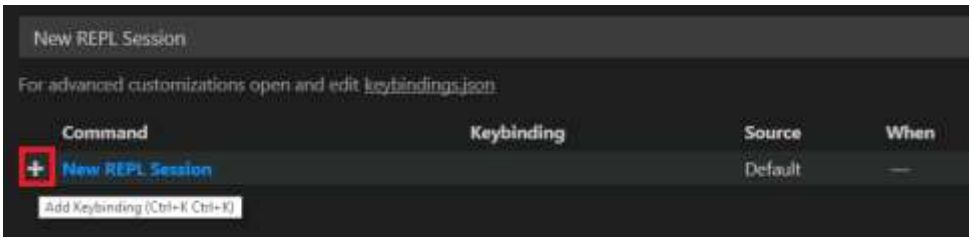
Cum?

Mergeți la `File → Preferences → Keyboard Shortcuts`

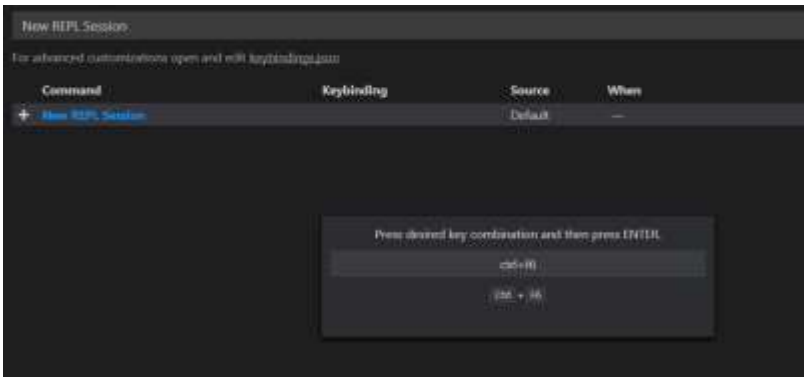
În panelul care se deschide, scrieți în câmpul de căutare textul `New REPL Session`.



Mergeți cu cursorul peste linia unde scrie „New REPL Session” și apăsați pe butonul „+” care va apărea:



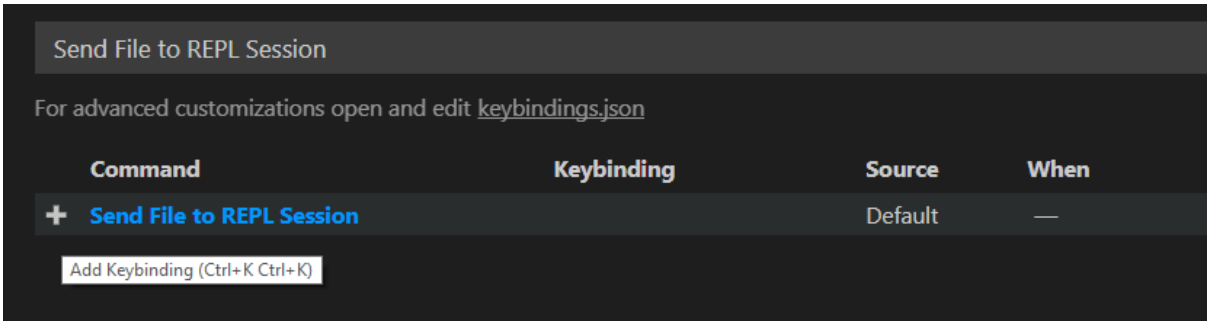
Va apărea o fereastră nouă. Pe tastatură, apăsați simultan pe butoanele „Ctrl” și „F6” pentru a atribui scurtătura Ctrl+F6 acestei comenzi:



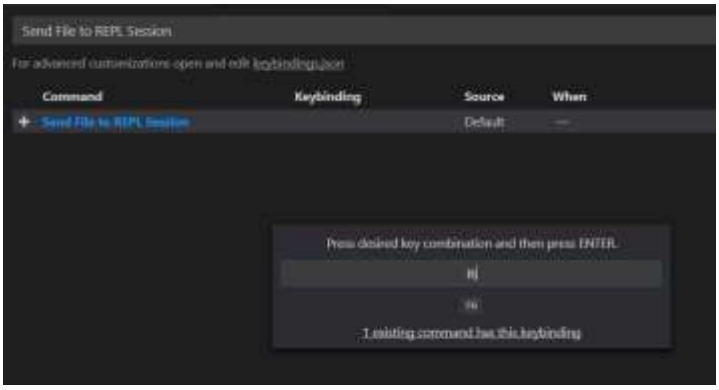
Apăsați „Enter” pentru a salva scurtătura. Ar trebui să apară noua scurtătură lângă numele comenzii.

Asemănător, căutăm comanda „Send File to REPL Session”.

Apăsăm pe butonul „+” de lângă comanda „Send File to REPL Session”.



Va apărea o fereastră nouă. Pe tastatură apăsați butonul „F6” pentru a atribui scurtătura F6 acestei comenzi:



Apăsați tasta „Enter” pentru a salva scurtătura. Ar trebui să apară noua scurtătură lângă numele comenzii:

Notă: Dacă comenzile nu apar când le căutați, asigurați-vă că aveți instalată extensia. Dacă aveți extensia instalată, reporniți Visual Studio Code. Daca în continuare nu apar comenzile, creați un fișier nou .ml și încercați din nou.

Folosirea Visual Studio Code

Crearea unui proiect nou

Pentru a putea să editām cod, trebuie mai întâi să selectăm un director (folder), care va conține fișierele pe care le vom crea. Pentru a face acest lucru, deschideți Visual Studio Code și mergeți în meniu la `File → Open Folder...`

În fereastra care se deschide, creați directorul care va conține codul. Este preferabil să creați unul nou, nu să folosiți unul existent, deoarece, pe lângă fișierele cu cod, vor apărea și alte fișiere conexe și folosirea unui director existent poate să facă dificilă identificarea fișierelor relevante.

Denumiți directorul cu un nume sugestiv, de exemplu Lab1:

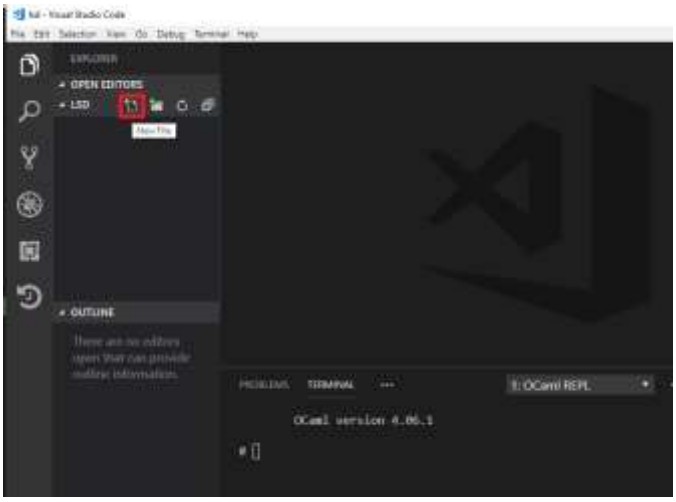
Asigurați-vă că directorul nou creat este selectat (este marcat cu culoare albastru deschis) și apăsați butonul „`Select Folder`”.

În acest moment, Visual Studio Code va afișa conținutul directorului (dacă ați creat un director nou, conținutul este gol). Verificați că numele directorului apare în fereastra editorului:

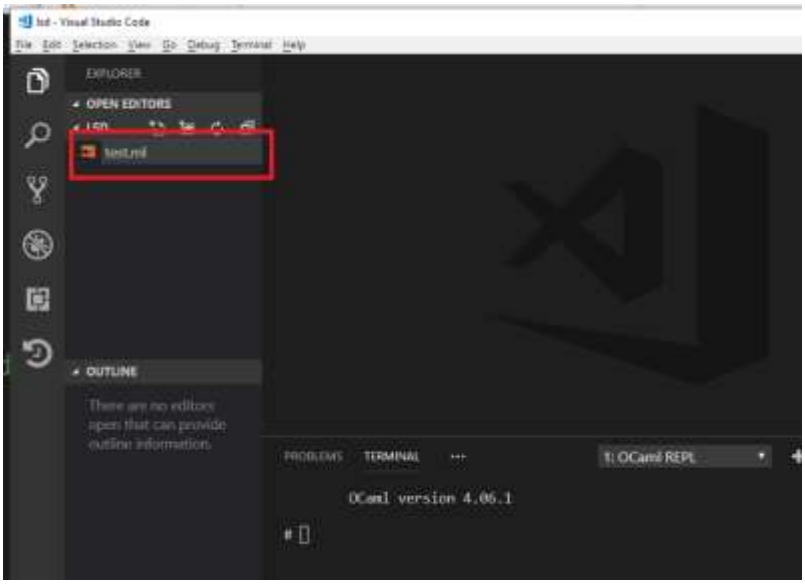
Crearea și salvarea unui fișier

După ce am urmat pașii de a crea un proiect nou , putem să creăm un fișier `.ml` nou.

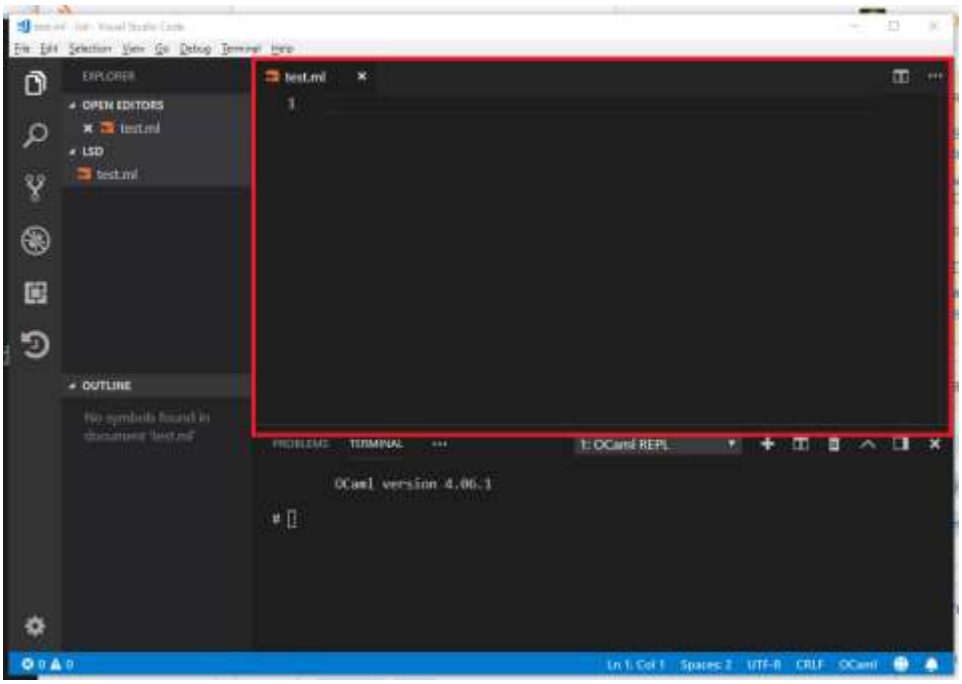
Pentru a face acest lucru, mergem cu cursorul peste numele directorului, și va apărea butonul de creare a unui fișier:



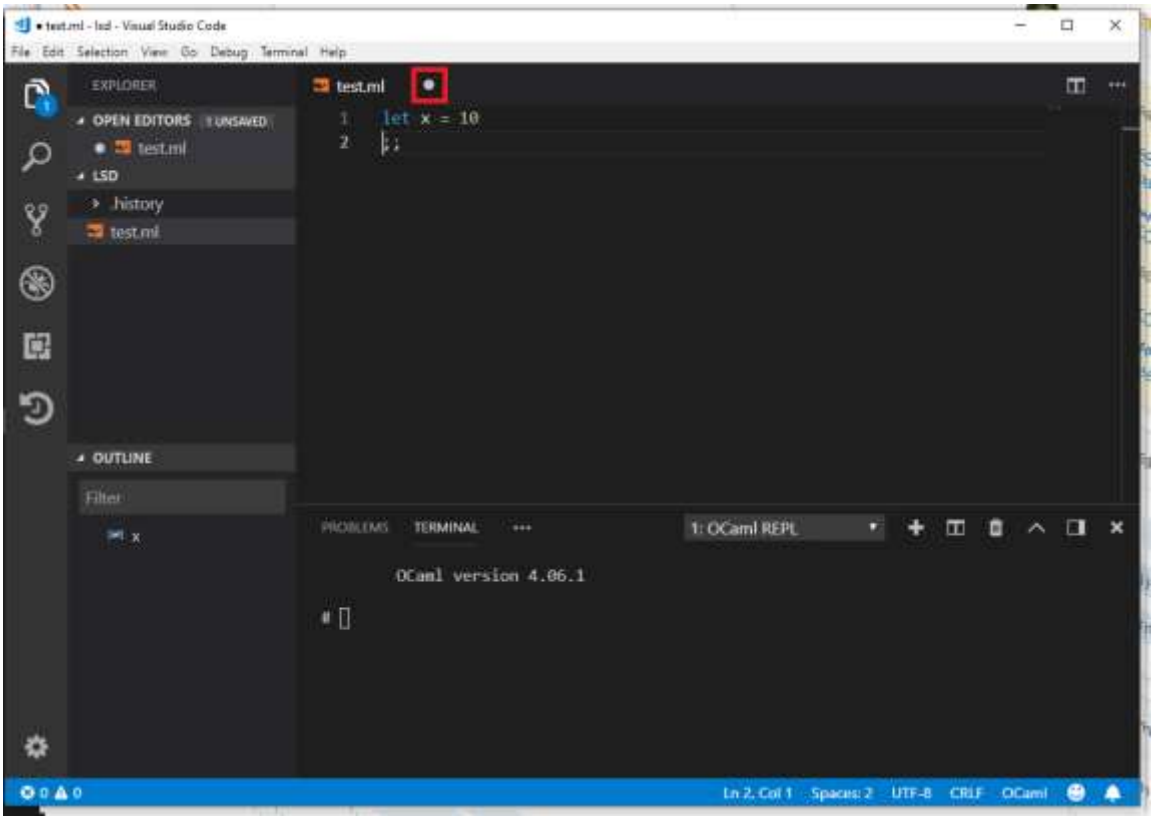
În câmpul text care apare, scrieți numele fișierului. Puteți să alegeți orice nume, dar el trebuie să se termine cu extensia `.ml`.



După ce am scris numele dorit, apăsăm tasta „Enter” și fișierul nou creat se va deschide. Acum putem să scriem cod în acest fișier:



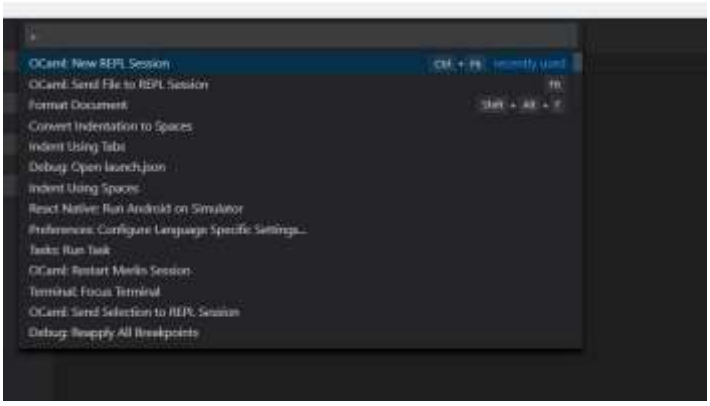
După ce am terminat de editat fișierul, putem apăsa concomitent tastele „Ctrl” și „S” pentru a salva modificările. Dacă un fișier conține modificări nesalvate, va apărea un cerc lângă numele fișierului:



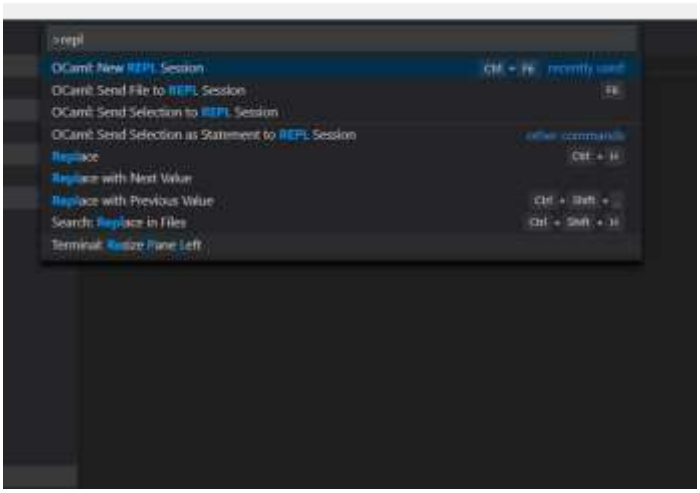
Paleta de comenzi

Visual Studio Code dispune o mulțime de comenzi pe care le putem executa în editor. Pe lângă comenzile care le oferă implicit editorul (de exemplu, comanda „Salvare fișier”), extensiile instalate pot să adauge și ele comenzi (de exemplu, extensia pentru OCaml adaugă comanda „Deschide interpretor”). Pentru a accesa aceste comenzi trebuie să deschidem paleta de comenzi.

Pentru a deschide paleta de comenzi trebuie să apăsăm simultan tastele Ctrl+Shift+P. În parte de sus a editorului, se va deschide paleta de comenzi:



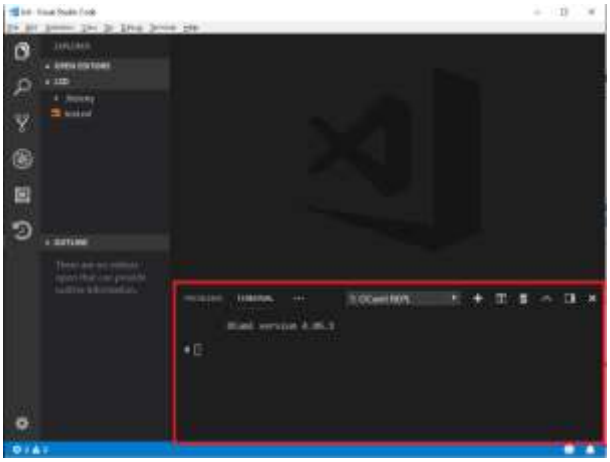
După ce paleta s-a deschis, putem să începem să căutăm o comandă. Spre exemplu, putem căuta comanda „New REPL Session”. Nu trebuie să scriem întreaga denumire a comenzii, ci ajunge să scriem „REPL”:



Putem să selectăm comanda pe care o dorim folosind săgețile sus/jos. Când am găsit comanda dorită, apăsăm tasta „Enter” și comanda va fi executată. Pentru a deschide interpretorul OCaml, selectăm comanda „New REPL Session”.

Deschiderea interpretorului OCaml

Dacă am urmat pașii de mai sus, în partea de jos a editorului ar fi trebuit să apară interpretorul OCAML:



Dacă facem click în zona interpretorului, putem să scriem comenzi care să fie executate de acesta.