

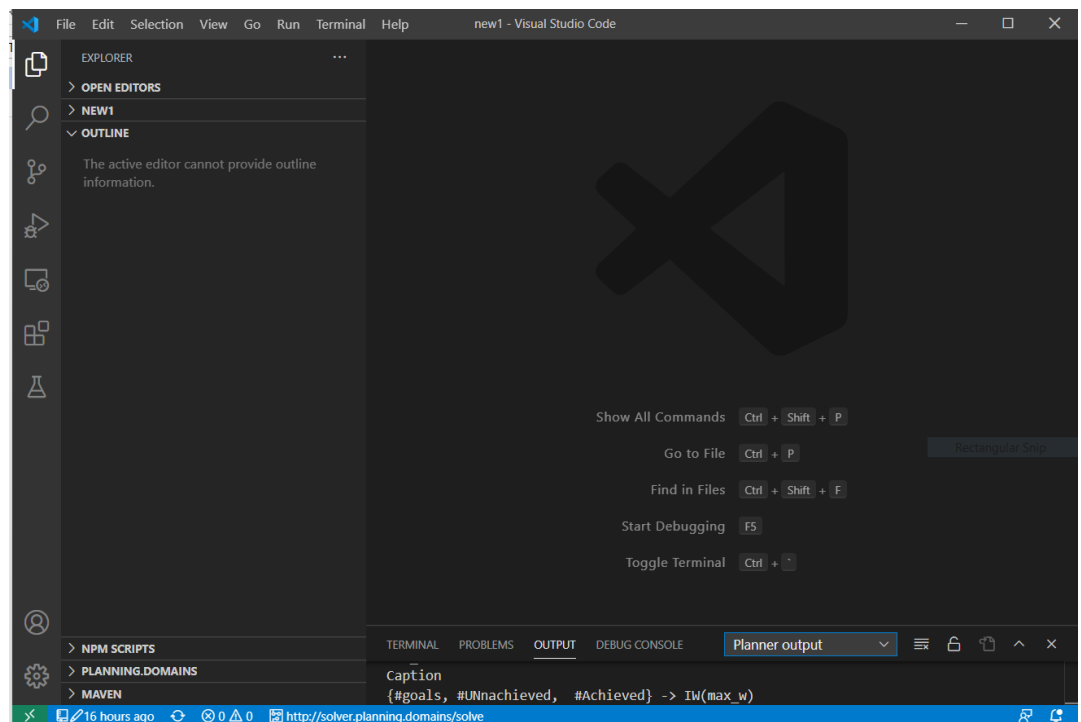
How to write, run and execute PDDL files

In the PAR course, we will use the following software:

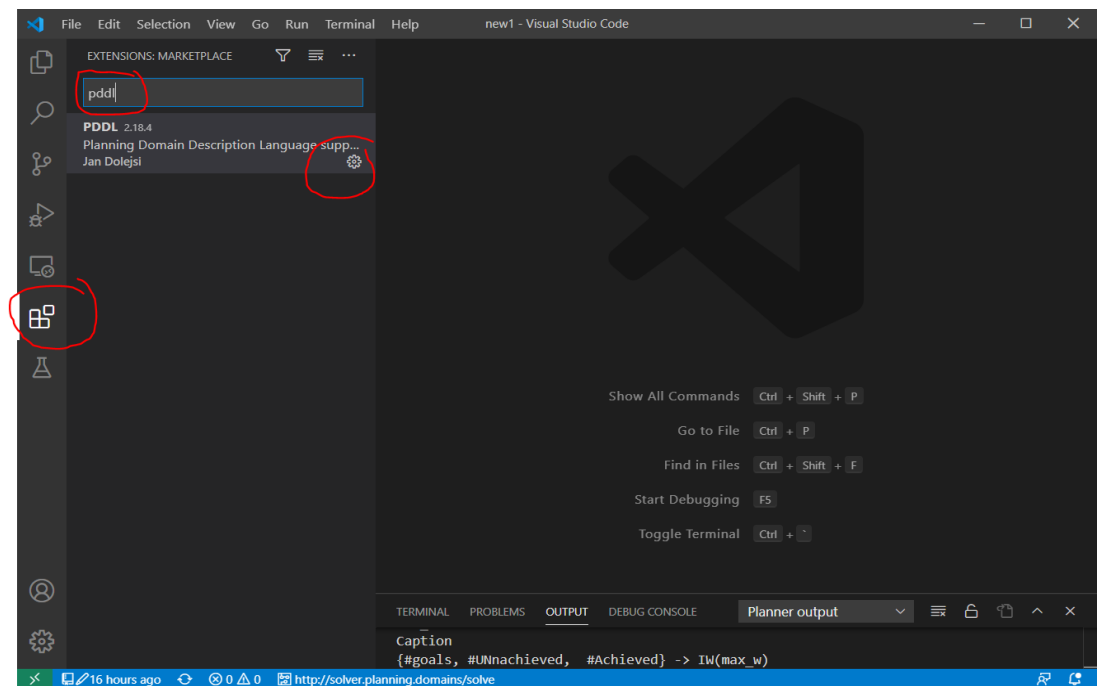
- Visual studio Code with PDDL packages,
<https://marketplace.visualstudio.com/items?itemName=jan-dolejsi.pddl>
- FF (Fast-Forward) Planning
Software: <http://www.ai.mit.edu/courses/16.412J/ff.html>
- Graphplan Planning
Software: <http://www.ai.mit.edu/courses/16.412J/Graphplan.html>
- Online PDDL Editor: <http://editor.planning.domains/>

Using Visual studio Code for writing and executing the PDDL files
([setup-overview](#))

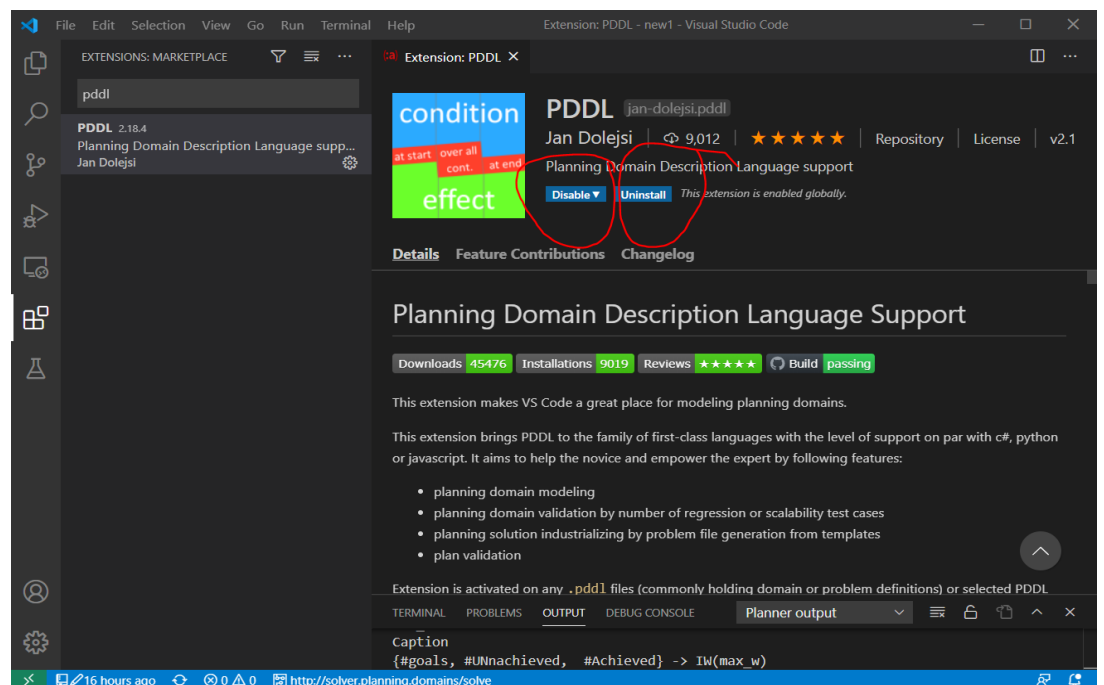
- 1- Visual studio code installation (Windows), please follow this [link](#)
 - 2- Visual studio code installation (Linux), please follow this [link](#)
 - 3- Visual studio code installation (MacOS), please follow this [link](#)
- Install the **PDDL Extension**, or follow the following
 - Open VS code



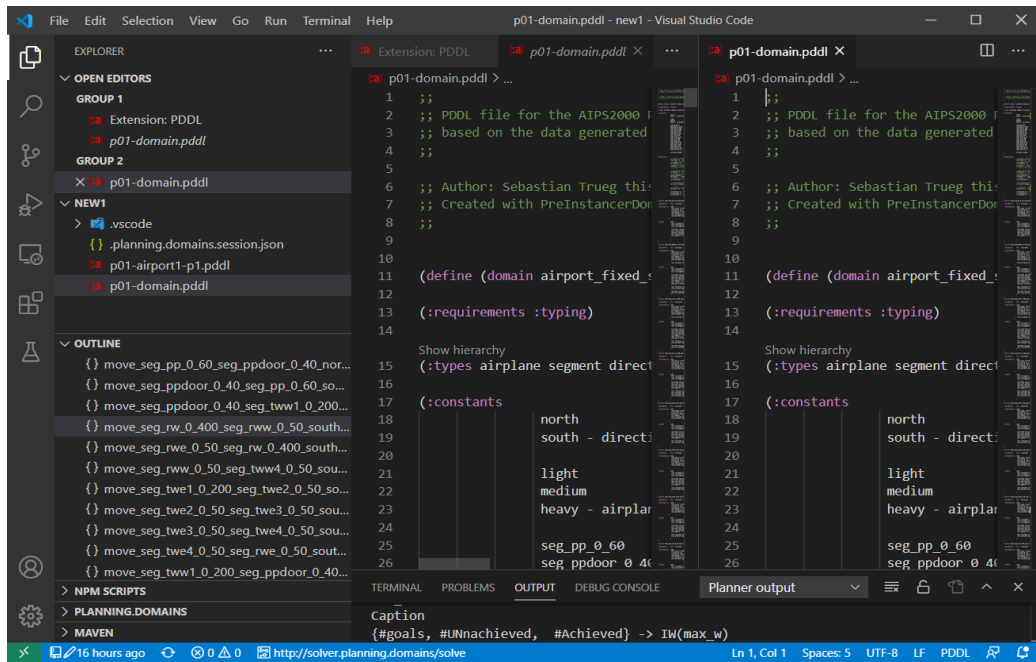
- Open extensions on the left toolbar and write PDDL, then you can install PDDL extension easily.



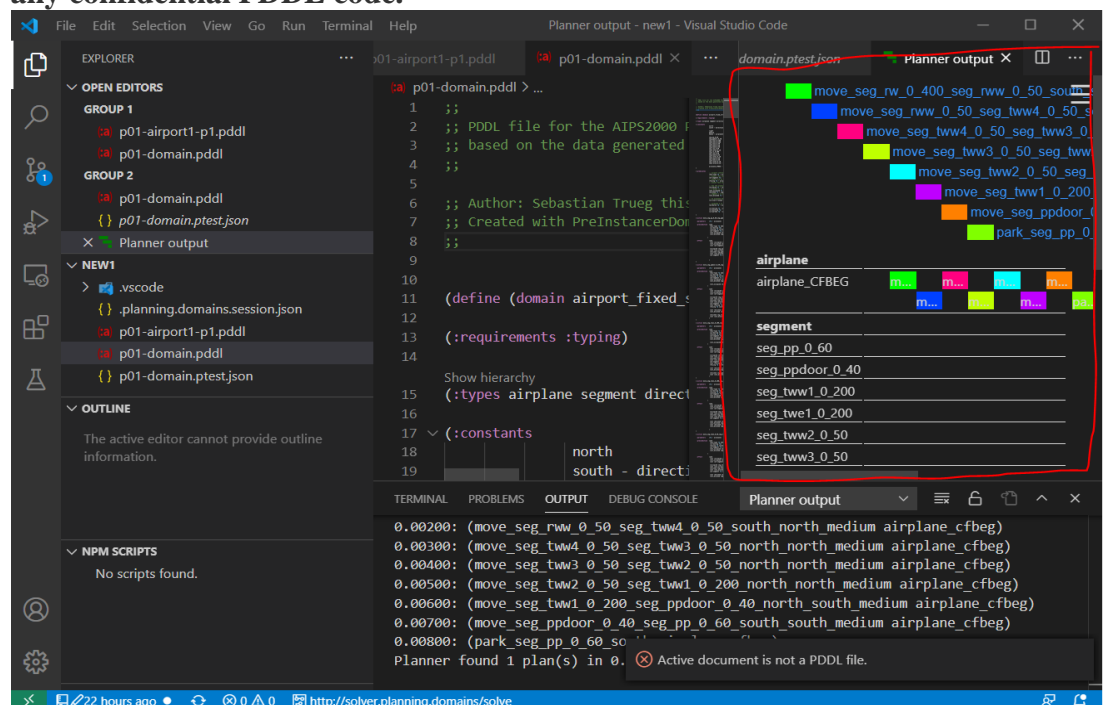
- Open PDDL extension to enable/disable/uninstall



- Open the *domain.pddl* file and type domain. The auto-completion suggests to insert the entire structure of the domain file. Use the Tab and Enter keys to skip through the placeholders and make your selections.

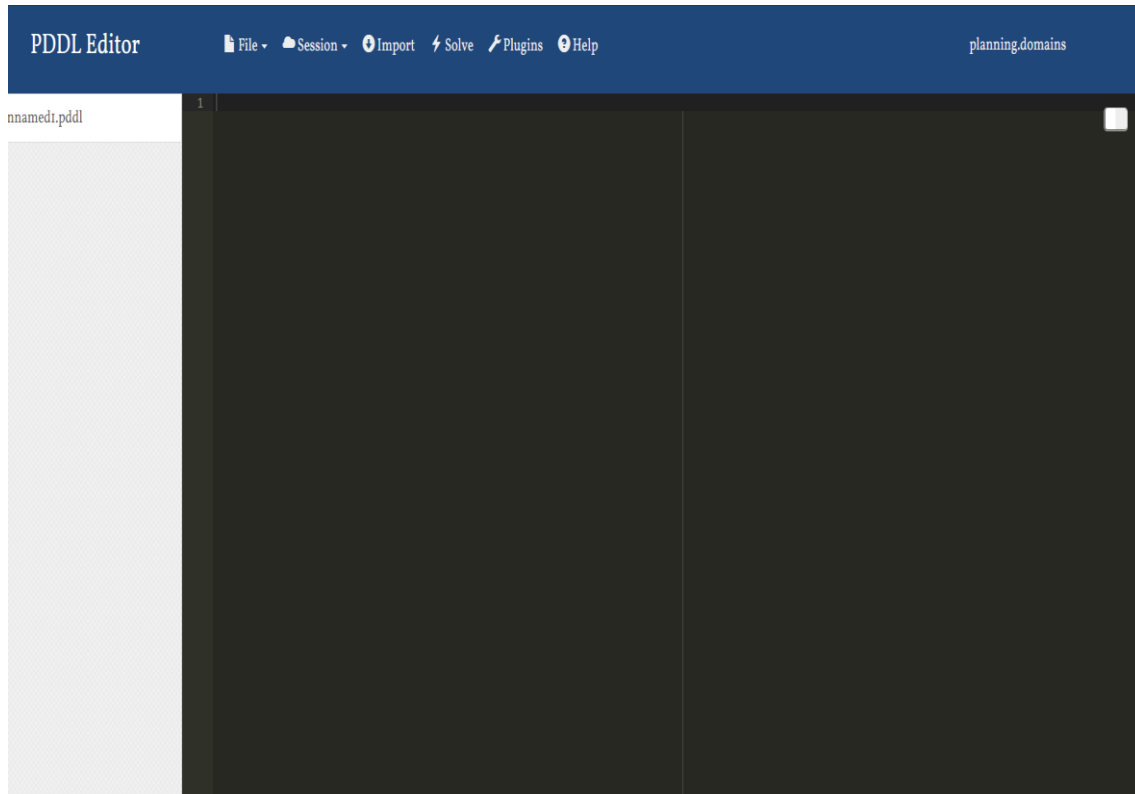


- Open the *problem.pddl* file and type problem. The auto-completion suggests to insert the entire structure of the problem file. Make sure that the (domain name) here matches the name selected in the domain file.
- When prompted to install the VAL (i.e. Validator) tools, follow the instructions. This will bring a PDDL parser and plan validation utilities to your experience.
- When you are ready to run the planner on your domain and problem files (both must be open in the editor), invoke the planner via context menu on one of the file text content, or via the Alt + P shortcut. The [planning.domains](#) solver will be used, so do not send any confidential PDDL code.

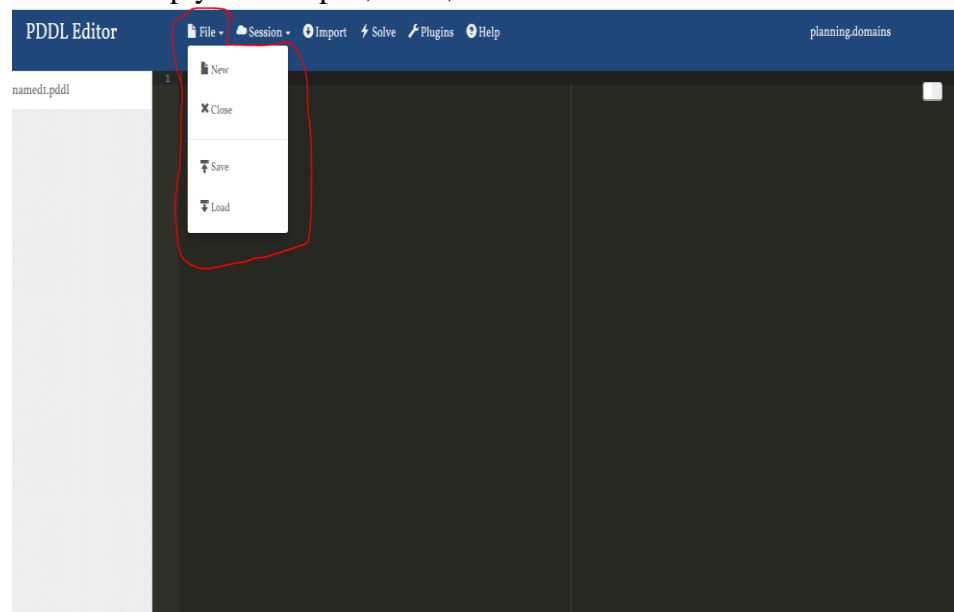


- Configure your own PDDL planner by following [instructions](#).

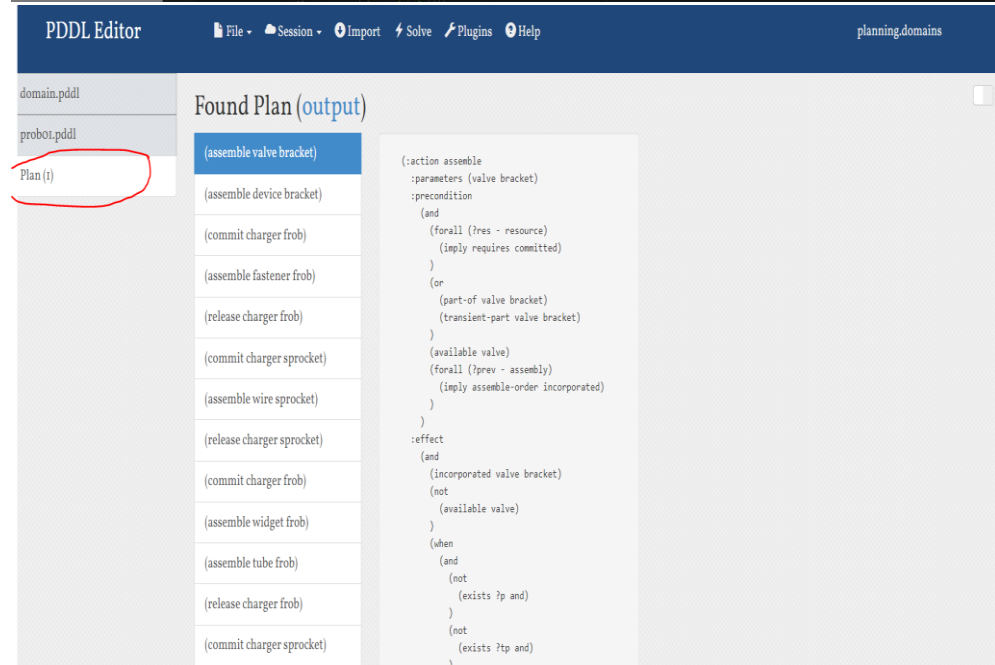
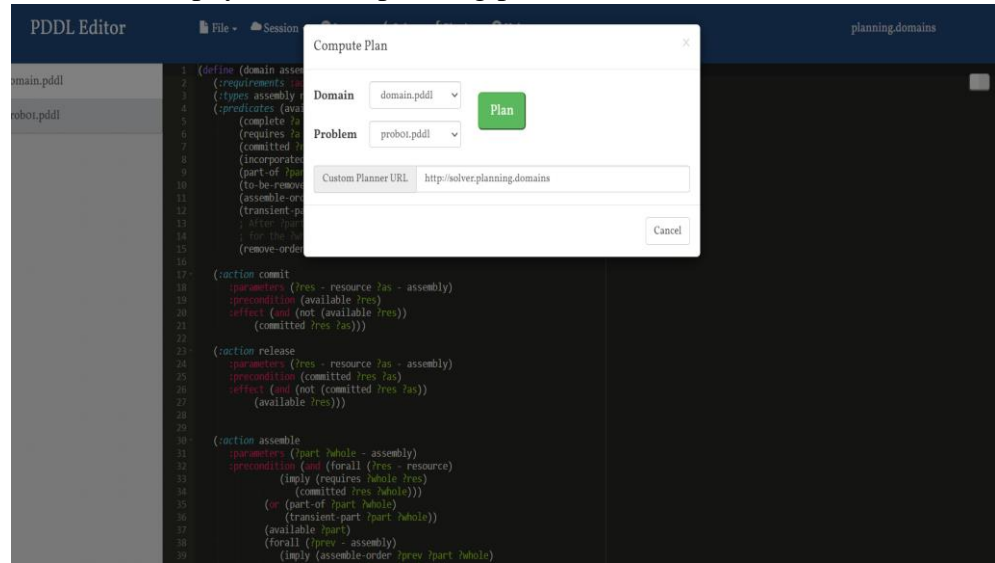
Using [Online PDDL Editor](#) for writing and excusing PDDL file



- From File tap you can open, close, save and load a PDDL file

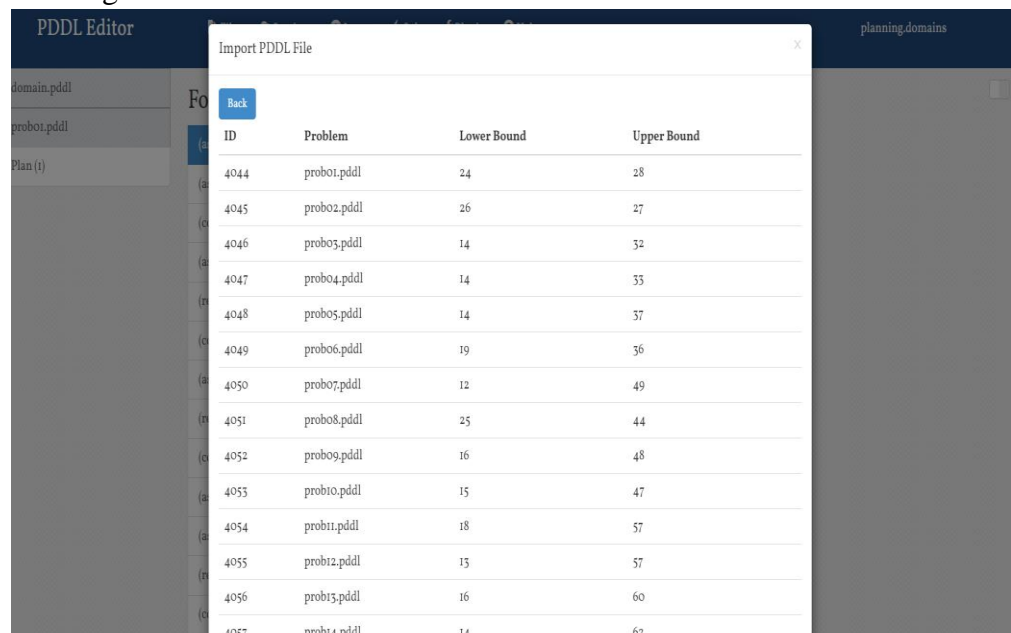


- From Solve tap, you run the planning problem



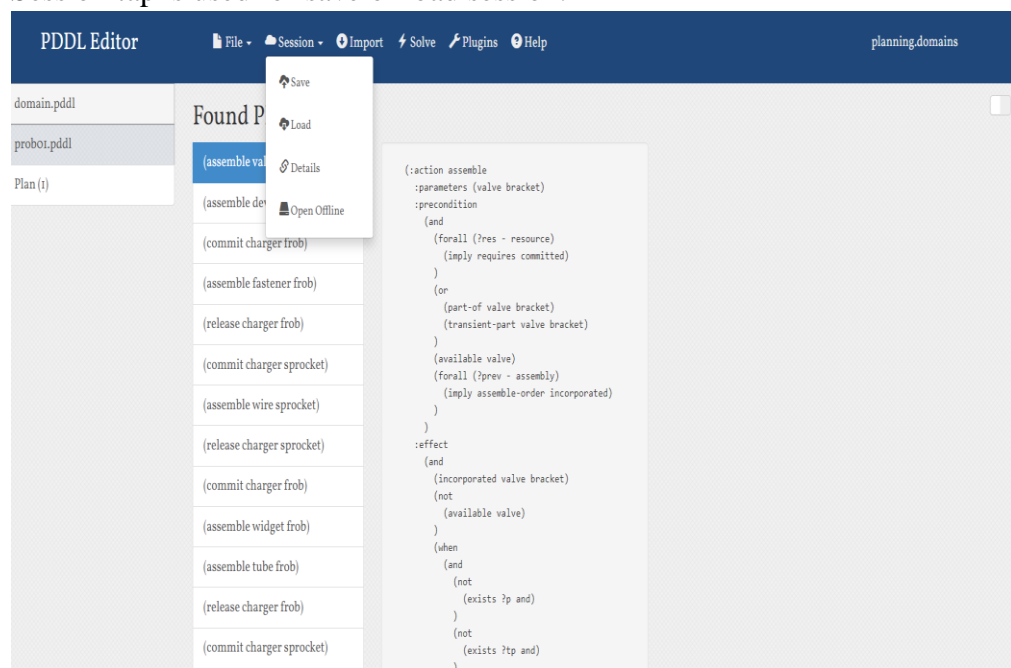
-

- Import tap is used for find a solution of standard automated planning challenges.



ID	Problem	Lower Bound	Upper Bound
4044	probo1.pddl	24	28
4045	probo2.pddl	26	27
4046	probo3.pddl	14	32
4047	probo4.pddl	14	33
4048	probo5.pddl	14	37
4049	probo6.pddl	19	36
4050	probo7.pddl	12	49
4051	probo8.pddl	25	44
4052	probo9.pddl	16	48
4053	probo10.pddl	15	47
4054	probo11.pddl	18	57
4055	probo12.pddl	15	57
4056	probo13.pddl	16	60
4057	probo14.pddl	14	62

- Session tap is used for save or load session.



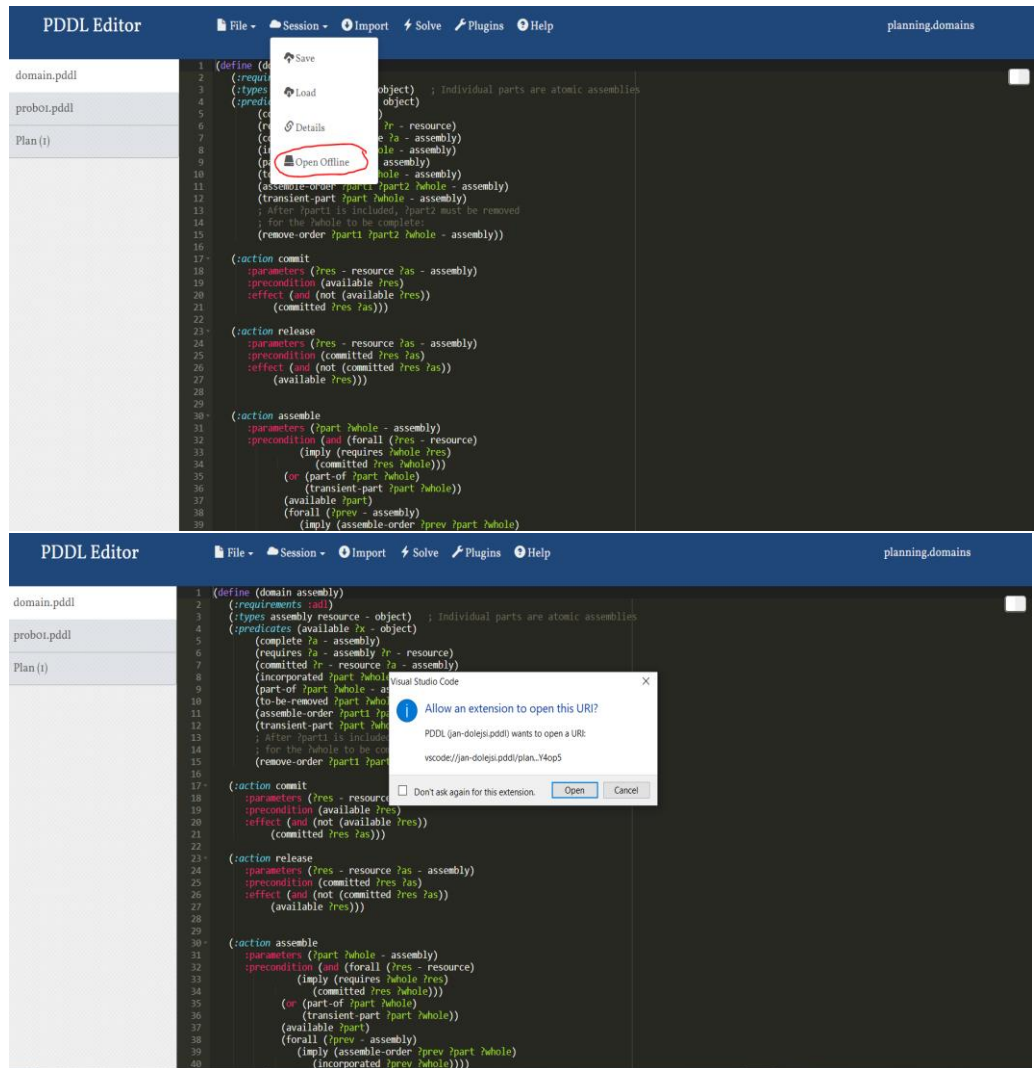
The 'Session' menu is open, showing options: Save, Load, (assemble valve), Details, and Open Offline. The background shows a list of planning problems on the left and a PDDL problem definition on the right.

```

(:action assemble
:parameters (valve bracket)
:precondition
  (and
    (forall (?res - resource)
      (imply requires committed)
    )
    (or
      (part-of valve bracket)
      (transient-part valve bracket)
    )
    (available valve)
    (forall (?prev - assembly)
      (imply assemble-order incorporated)
    )
  )
:effect
  (and
    (incorporated valve bracket)
    (not
      (available valve)
    )
  )
  (when
    (and
      (not
        (exists ?p and)
      )
      (not
        (exists ?tp and)
      )
    )
  )
)

```

- Using open offline in Session tap, firstly you need to save the session and then open offline will allow you to download the project to your computer and open VS code, if you are already installed on your PC. You need to select your local folder to save the project.



Now it will be opened with VS code.

