

Run

```
In[*]:= SetDirectory["C:\\drorbn\\AcademicPensieve\\Projects\\Theta"];
$PageWidth = 6.4;
Make["Implementation.tex", "Theta.nb", Hold[nb2tex["Theta",
  "Implementation", PDFFolder -> "Snips/Implementation", PDFWidth -> $PageWidth]]];
Make["Theta.pdf", {"Theta.tex", "defs.tex", "Implementation.tex"},
Hold[Run@
  "\\C:\\Users\\drorb\\AppData\\Local\\Programs\\MiKTeX\\miktex\\bin\\x64\\pdflatex.exe
  \" Theta.tex";
Run@
  "\\C:\\Users\\drorb\\AppData\\Local\\Programs\\MiKTeX\\miktex\\bin\\x64\\pdflatex.exe
  \" Theta.tex"
]]
Making {Implementation.tex} ...
Making {Theta.pdf} ...
Out[*]=
0
```

A la carte Run

```
(Alt) In[*]:=
Run@
  "\\C:\\Users\\drorb\\AppData\\Local\\Programs\\MiKTeX\\miktex\\bin\\x64\\pdflatex.exe\"
  Theta.tex"
(Alt) Out[*]=
0
```

Make

```

In[*]:= Make::usage =
  "Make[target, sources, Hold[action]] makes a target, or a list of targets, given
  sources, or a list of sources, in the style of the unix 'make' command.";
Make[target_String, sources_, action_Hold] :=
  Make[Evaluate@{target}, sources, action];
Make[targets_, source_String, action__Hold] :=
  Make[targets, Evaluate@{source}, action];
Make[targets_List, sources_List, action_Hold] := Module[{},
  If[
    (And @@ ((FileType[#] != None) & /@ sources)) &&
    Or[
      Or @@ ((FileType[#] === None) & /@ targets),
      Min[AbsoluteTime[FileDate[#]] & /@ targets] <
      Max[AbsoluteTime[FileDate[#]] & /@ sources]
    ],
    Print["Making ", targets, " ..."];
    ReleaseHold[action]
  ]
];

```

nb2tex

As in <http://drorbn.net/AcademicPensieve/Projects/nb2tex/>.

```

In[*]:= SetOptions[$FrontEndSession, PrintingStyleEnvironment -> "Working"];
nb2tex[nb_String, opts__Rule] := nb2tex[nb, nb, opts];

```

```

In[*]:= nb2tex[nb_String, tex_String, opts___Rule] := Module[
  {notebook, PDFCounter = 0, type,
   tag, pdfname, cells, cell, c, c1, texfiles = {}, TeXOut,
   PDFFolder = PDFFolder /. {opts} /. PDFFolder → nb
  },
  nb2tex$TeXFileName = tex <> ".tex";
  nb2tex$PDFWidth = PDFWidth /. {opts} /. PDFWidth → 6.5;
  TeXOut[s_String] := (texfiles = texfiles ∪ {nb2tex$TeXFileName};
   WriteString[nb2tex$TeXFileName, s]);
  notebook = NotebookGet[NotebookOpen@FileNameJoin[{Directory[], nb <> ".nb"}]];
  If[FileType[PDFFolder] === None, CreateDirectory[PDFFolder]];
  DeleteFile /@ FileNames["*.pdf", PDFFolder];
  cells = Cases[notebook, c_Cell /; Length[c] ≥ 2, ∞];
  Do[
    type = cell[[2]];
    tag = CellTags /. Cases[cell, _Rule] /. CellTags → "";
    Which[
      type == "Text" ^ tag == "tex", TeXOut[
        StringReplace[cell[[1]], {"'" → "'", "\"" → "\""}] <> "\n\n",
        StringMatchQ[tag, "pdf:" ~~ ___], (
          pdfname = PDFFolder <> "/" <> StringDrop[tag, 4] <> type <> ".pdf";
          Export[pdfname, Join[cell, Cell[PageWidth → 80 nb2tex$PDFWidth / 0.75]]];
          c1 = "c:\\drorbn\\bin\\cpdf.exe -scale-page \"0.75 0.75\" " <>
            pdfname <> " -o " <> pdfname;
          Close@OpenRead["!" <> c1];
        ),
      StringMatchQ[tag, "pdf" ~~ ___], (
          pdfname = PDFFolder <> "/" <> ToString[++PDFCounter] <> ".pdf";
          Export[pdfname, Join[cell, Cell[PageWidth → 80 nb2tex$PDFWidth / 0.75]]];
          c1 = "c:\\drorbn\\bin\\cpdf.exe -scale-page \"0.75 0.75\" " <>
            pdfname <> " -o " <> pdfname;
          Close@OpenRead["!" <> c1];
          TeXOut[StringReplace[
            "\\nbpdfXXXType{pdfname}\n",
            {"XXX" → StringDrop[tag, 3], "Type" → type, "pdfname" → pdfname}
          ]]
        ),
      type == "Input" ^ tag == "exec", ToExpression[cell[[1]],
      True, Null
    ],
    {cell, cells}
  ];
  Close /@ texfiles;
]

```