

Nachdem das Skriptediting in TurboCad auf Ruby umgestellt wurde, muss man sich nun nach der entsprechend Entwicklungsplattform umsehen. Viele Skripte (*.rb) in TurboCad sind für Sketchup geschrieben. „Sketchup“ wird allerdings auf die TurboCad Anwendung „umgeleitet“. In den Skripten der TurboCad-Ruby Engine sind teilweise 3/4 der Codezeilen auskommentiert. Was man davon halten soll bleibt freigestellt. Da wurde was mit der heißen Nadel zusammengestellt. Nichts desto trotz, wie mache ich jetzt in Ruby??

Zur Kontaktaufnahme mit Ruby habe ich folgende Tutorials aus Youtube geladen:
(Vielen dank an Autor Lampe)

Name	Erw.	Größe
[..]		<DIR>
Ruby Tutorial 1_ Installation und erstes Projekt	mp4	31.304.446
Ruby Tutorial 2_ Zahlen in ruby	mp4	19.293.229
Ruby Tutorial 3_ Strings als Zeichenketten	mp4	32.499.754
Ruby Tutorial 4_ Variablen und Typ umwandlung !	mp4	42.141.560
Ruby Tutorial 5_ Von der Tastatur lesen mit gets	mp4	23.886.614
Ruby Tutorial 6_ Wahrheitswerte also booleans	mp4	17.353.197
Ruby Tutorial 7_ die if schleife	mp4	22.248.368
Ruby Tutorial 8_ die while schleife	mp4	24.961.467
Ruby Tutorial 9_ upto downto	mp4	21.874.448
Ruby Tutorial 10_ case	mp4	18.389.656
Ruby Tutorial 11_ Array , Arrays	mp4	31.978.174
Ruby Tutorial 12_ Hashes	mp4	21.297.071
Ruby Tutorial 13_ regex Regular Expressions	mp4	22.764.478

Tutorial 1 zeigt die Installation von Netbeans für Ruby.

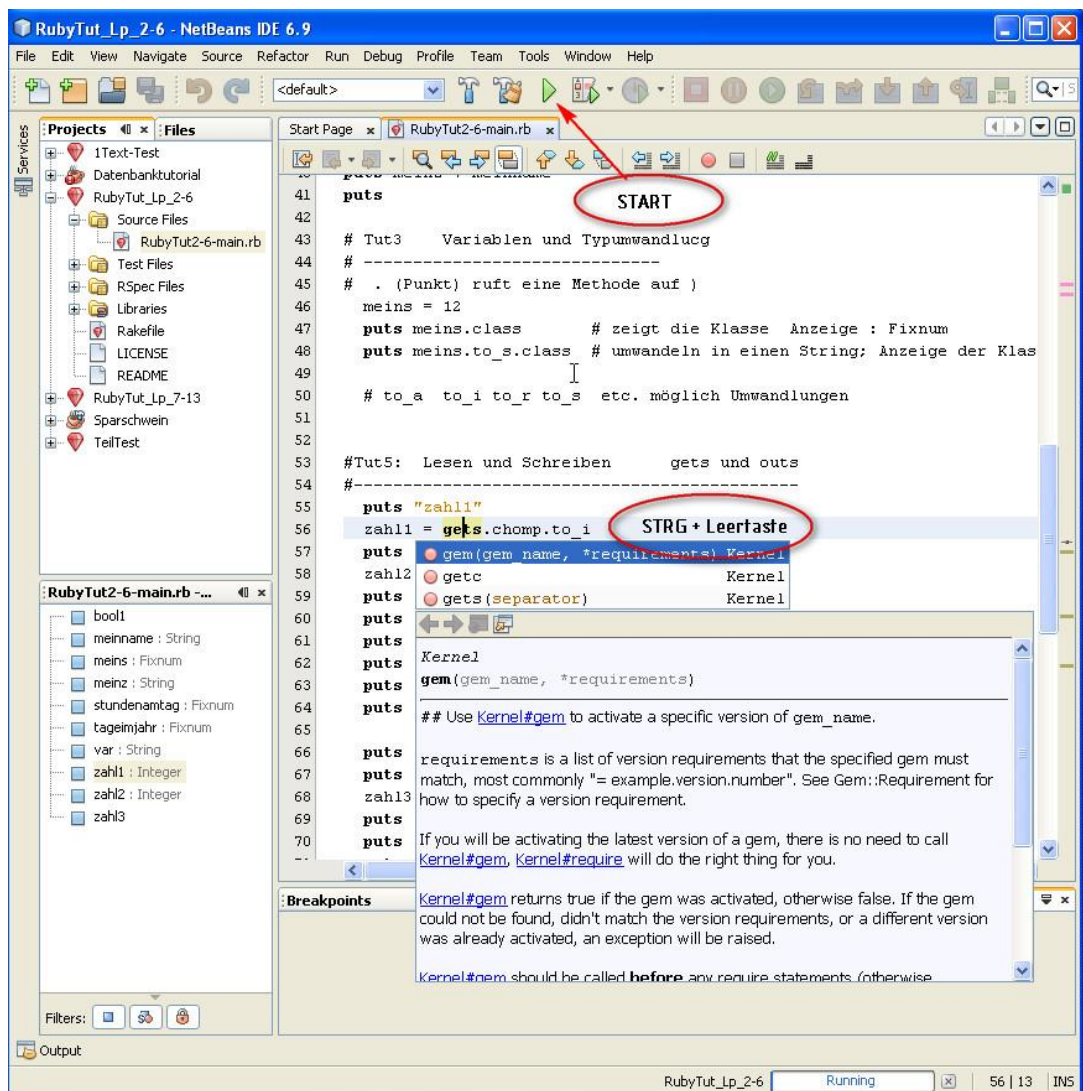
Die letzte Version mit Ruby gibt es hier:
<http://netbeans.org/downloads/6.9.1/index.html>

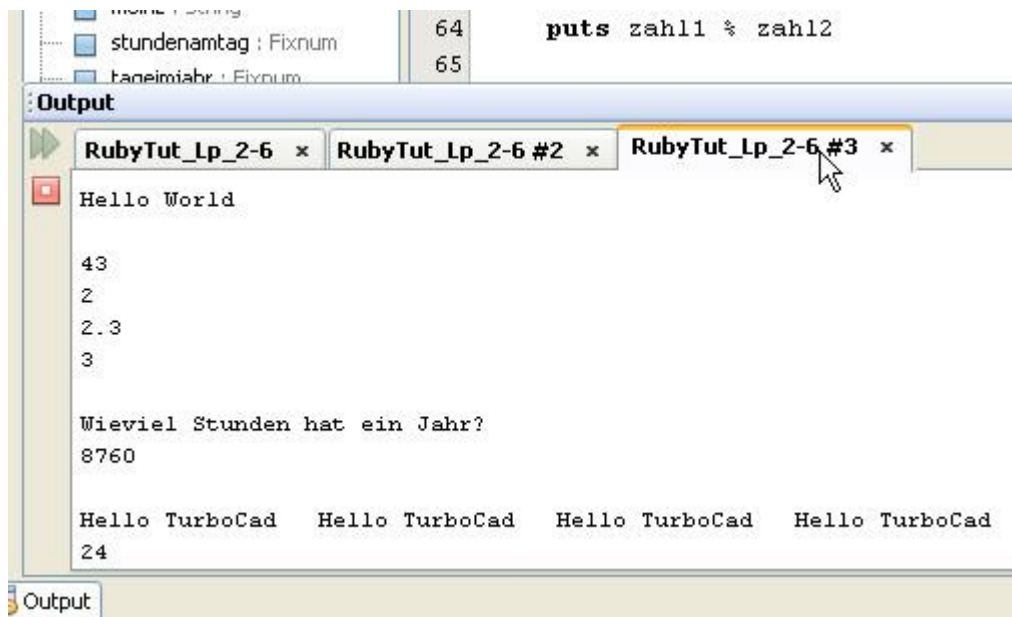
Da ich ohnehin NETBEANS auf meinem Rechner hatte, konnte ich noch für die Version 6.9 das RUBY-Plugin laden.

!! Schade: Neuere Versionen unterstützen RUBY nicht mehr.

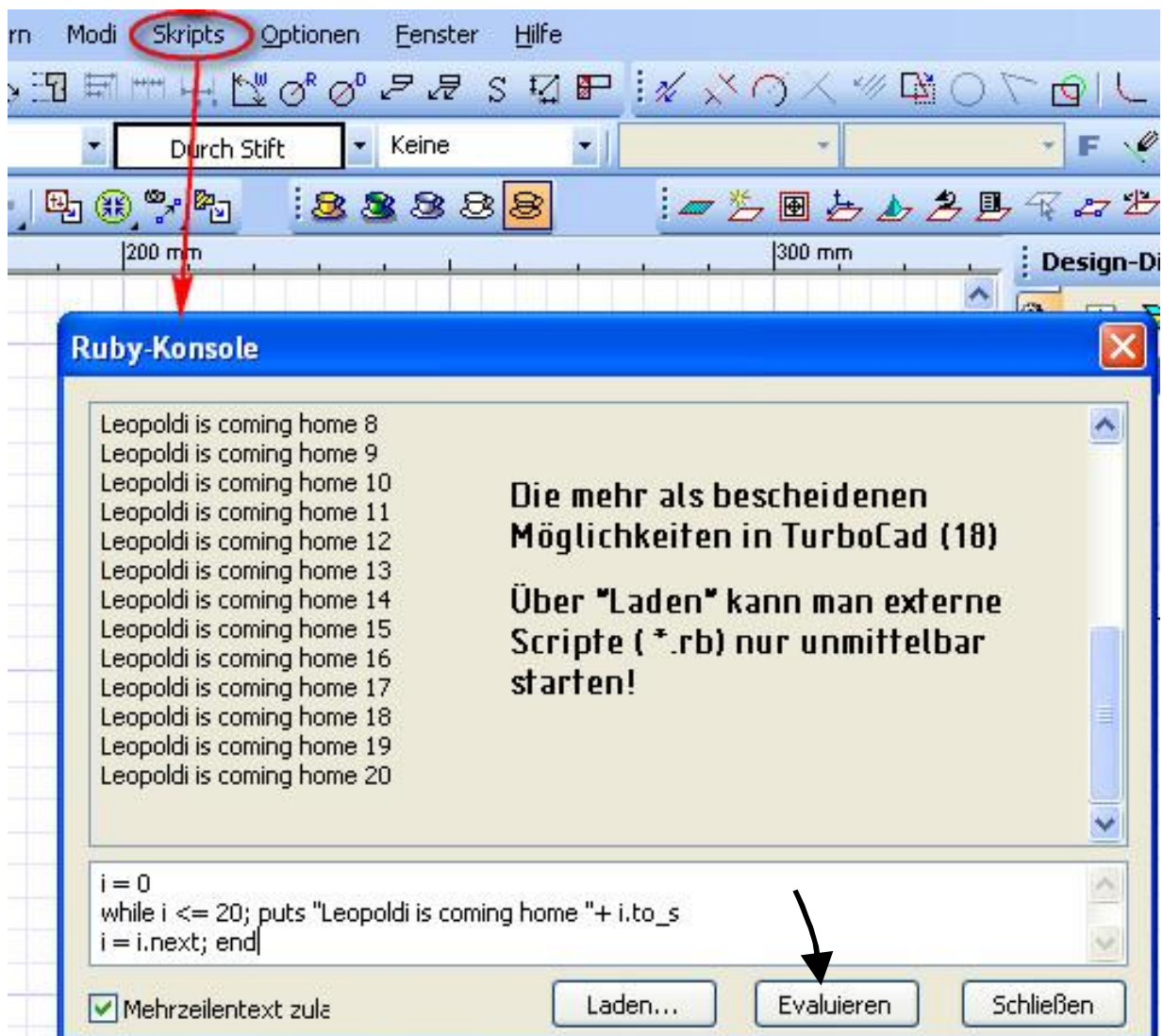
Ich habe die Tut-Videos abgespielt und in Netbeans 6.9/Ruby nachvollzogen. Das hat gut funktioniert. Lediglich das Debugging hat bei mir nicht funktioniert. Da blick ich *noch* nicht durch.

Auch die möglichen Breakpoints funktionieren nicht, aber da kann man ein gets als Haltepunkt einsetzen.



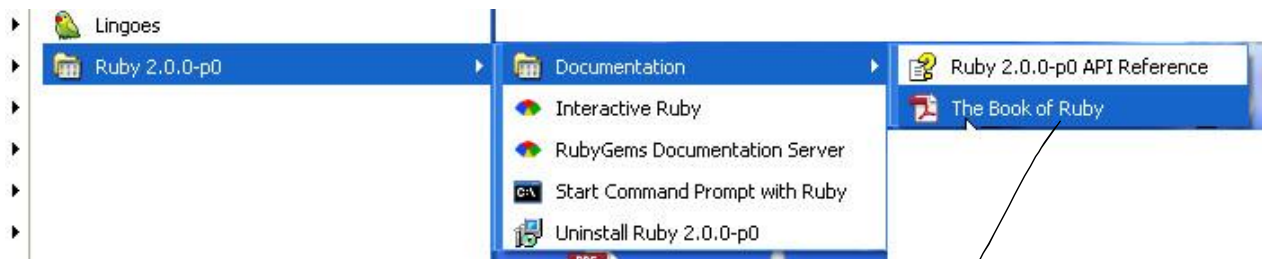


In TurboCad selbst gibt es die RUBY-Konsole.



Dazu gehören eine API- Reference, ein Handbuch und eine Konsole „Interactive Ruby“ (irb)

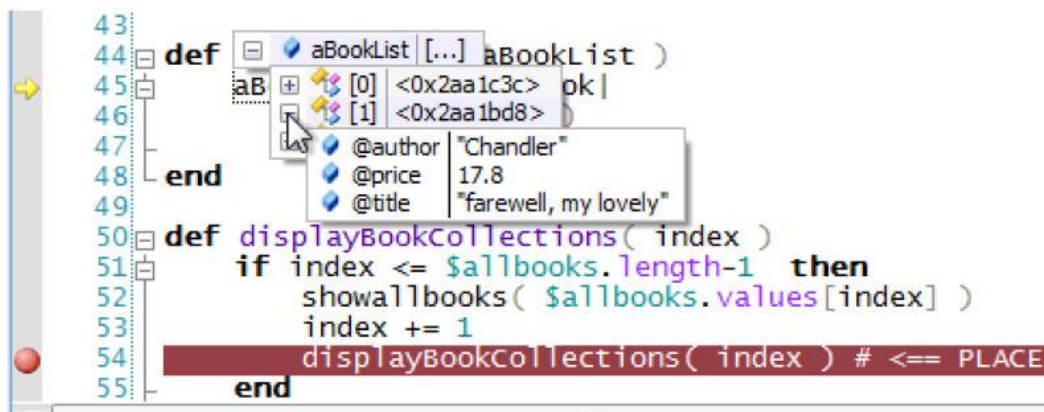
START|ALLE PROGRAMME



THE BOOK OF RUBY gehört zu ruby Installation

GRAPHICAL DEBUGGERS

For serious debugging, I would strongly recommend a graphical debugger. For example the debugger in the **Ruby In Steel IDE** allows you to set **breakpoints** and **watchpoints** by clicking the margin of the editor. It lets you monitor the **values** of selected **'watch variables'** or **all local variables in separate docked windows**. It maintains a 'callstack' of all method calls leading to the current point of execution and allows you to navigate 'backwards' through the callstack to view the changing values of variables. It also has full 'drill-down' expansion of variables to allow you to expand Arrays and hashes and look 'inside' complex objects. These capabilities go well beyond the features of the standard Ruby debugger.

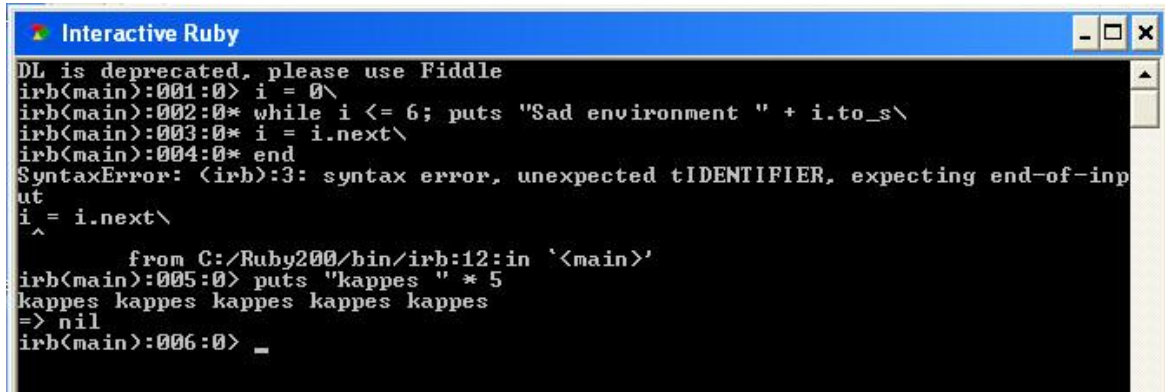


Empfohlen wird hier als IDE „Ruby in Steel“ welches auch unter Microsoft VisualStudio2010 installiert werden kann. (Kosten ~250\$)

Das scheint absolut professionell zu sein. (Kosten ~250\$)

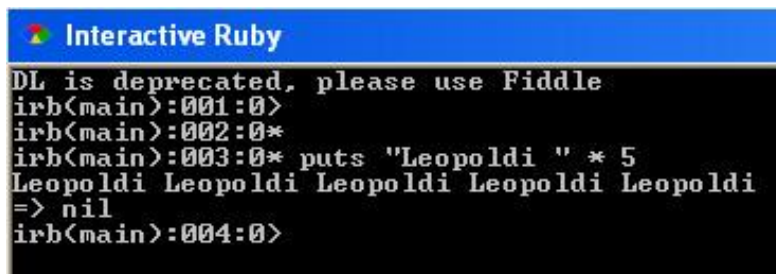
Als Entwicklungsumgebung allerdings ungeeignet:

funktionierte nicht:



```
Interactive Ruby
DL is deprecated, please use Fiddle
irb(main):001:0> i = 0\
irb(main):002:0> while i <= 6; puts "Sad environment " + i.to_s\
irb(main):003:0> i = i.next\
irb(main):004:0> end
SyntaxError: (irb):3: syntax error, unexpected tIDENTIFIER, expecting end-of-input
irb(main):005:0> puts "kappes " * 5
kappes kappes kappes kappes kappes
=> nil
irb(main):006:0> _
```

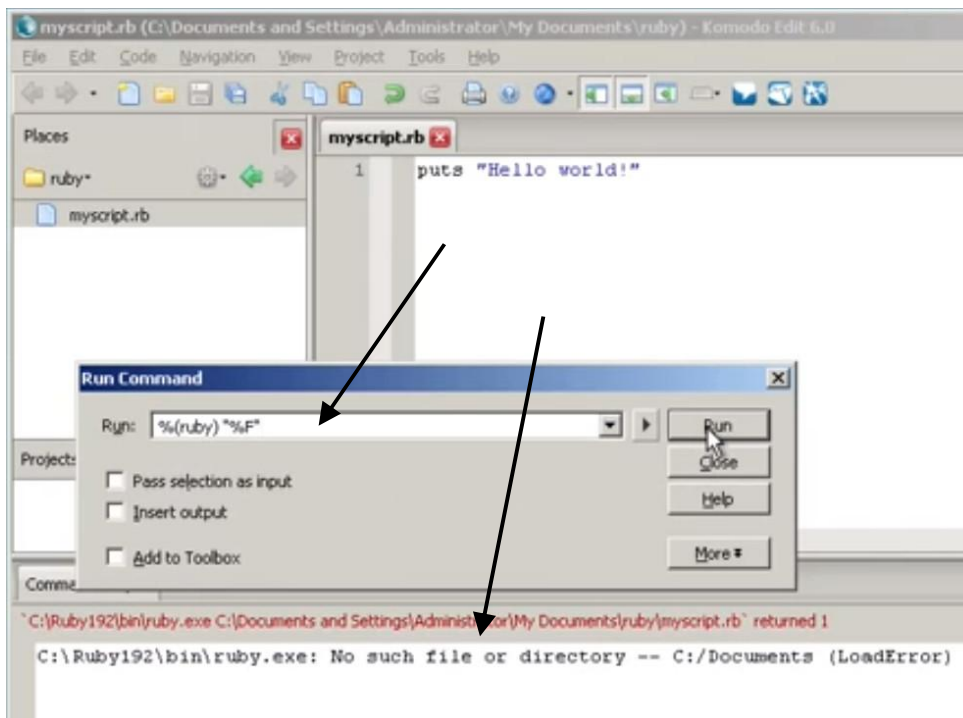
hat funktioniert:



```
Interactive Ruby
DL is deprecated, please use Fiddle
irb(main):001:0>
irb(main):002:0>
irb(main):003:0> puts "Leopoldi " * 5
Leopoldi Leopoldi Leopoldi Leopoldi Leopoldi
=> nil
irb(main):004:0>
```

Zu Eingabe eines Ruby-Skriptes kann man jeden beliebigen Editor nehmen. (PSPad, Notepad++, etc.)

Außer dem Syntax-Highlighters ist aber auch „Intellisense“ für den Einsteiger wertvoll. Dies bietet, dafür mit anderen Abstrichen (z.B. Run) der Editor „Komodo Edit“ (z.Zt. Version8)



an die Ausgabe le einstellen.

Resumee: Für alle genannten Beispiel gibt es in Youtube Videos. Die sollte man sich unbedingt anschauen bevor man sich für irgend etwas entscheidet.

In meiner Ruby-Lernphase werde ich weiter Netbeans 6.9.1 nutzen.