

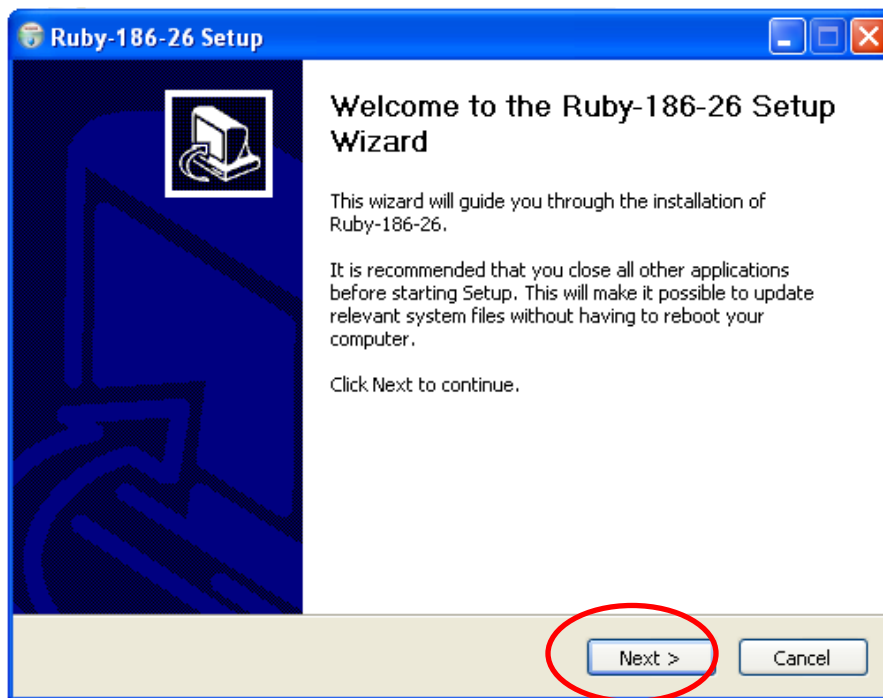
Part 1 of 2: Install and Use Rails & Hobo with SQLite as the Database Engine

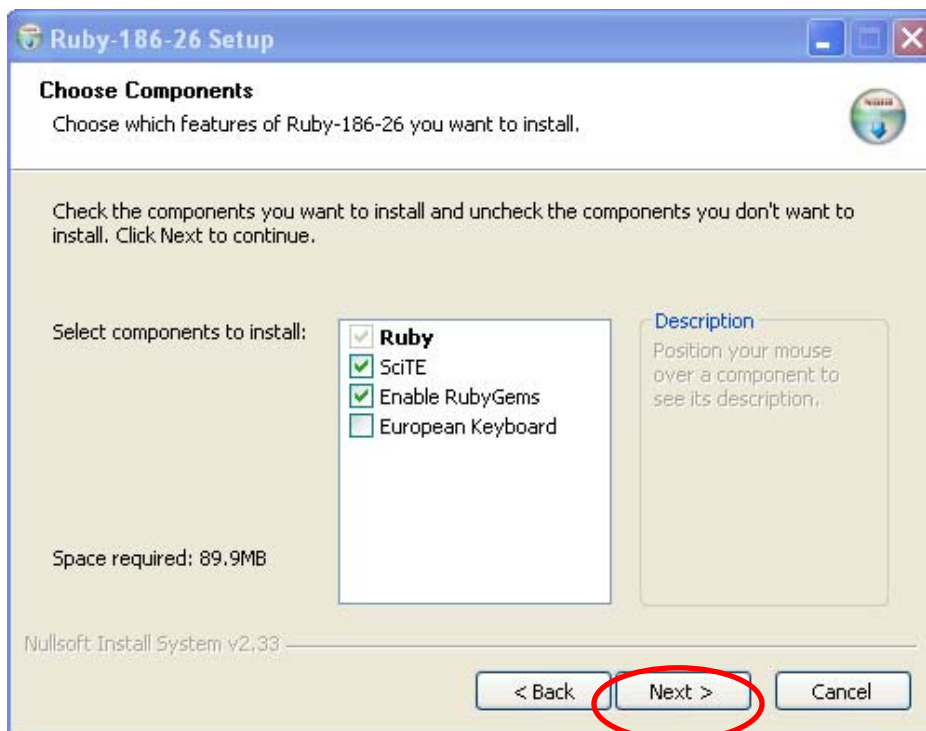
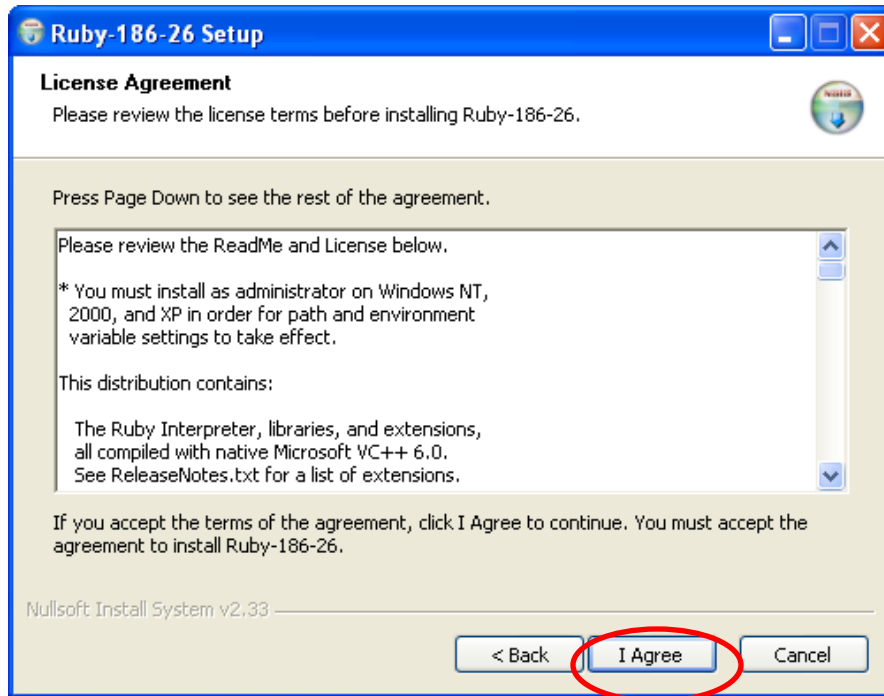
1. Download the latest stable Ruby version as a Windows installer executable file `ruby186-26.exe` (as of 2008-03-31) from rubyforge.org:

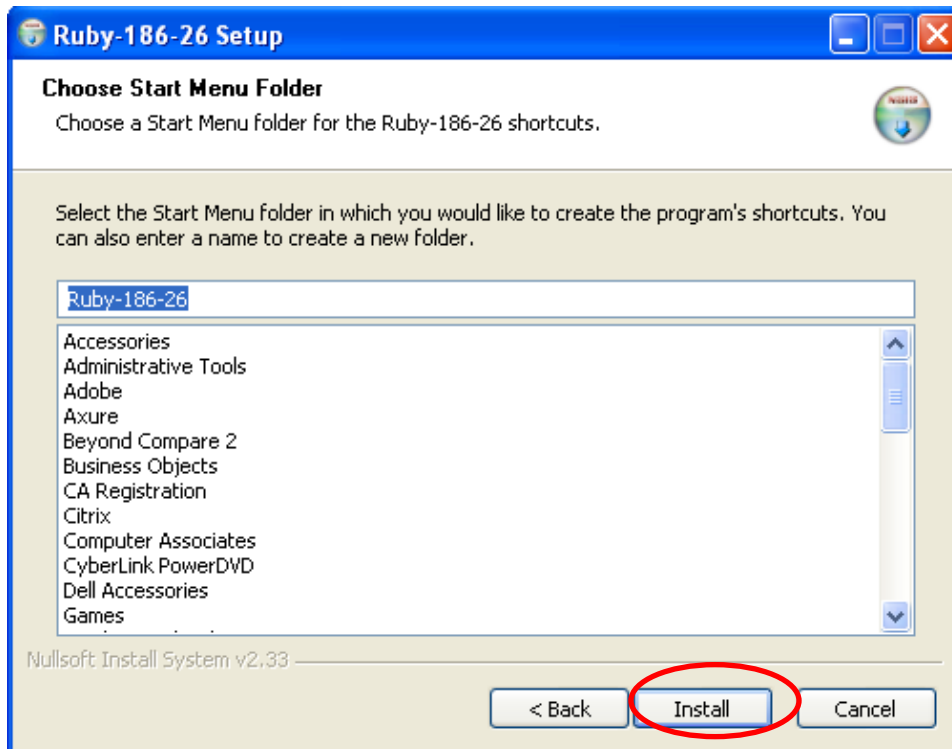
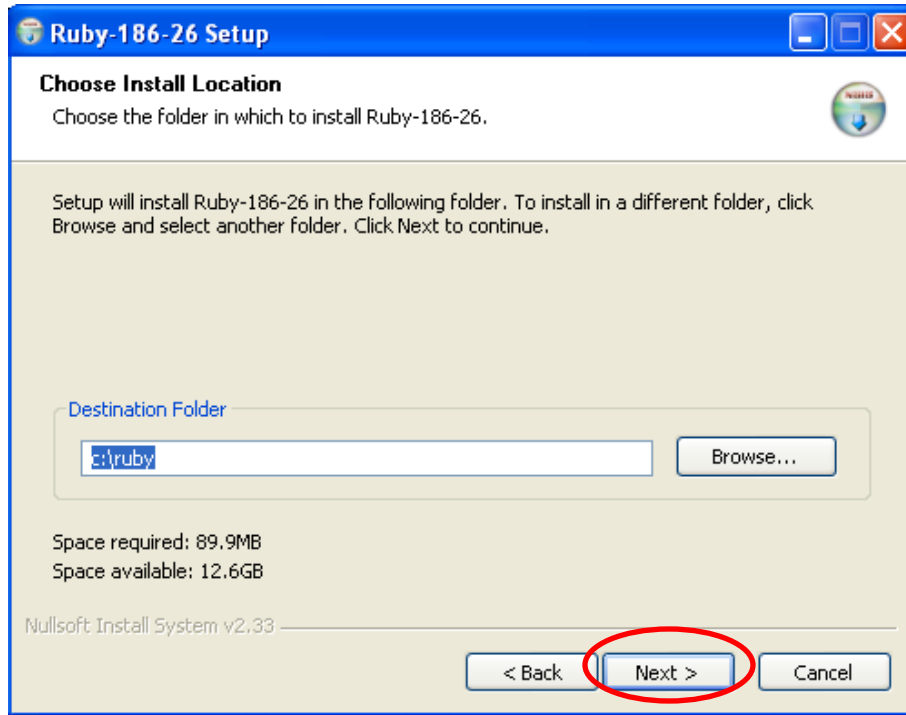
<http://rubyforge.org/projects/rubyinstaller/>

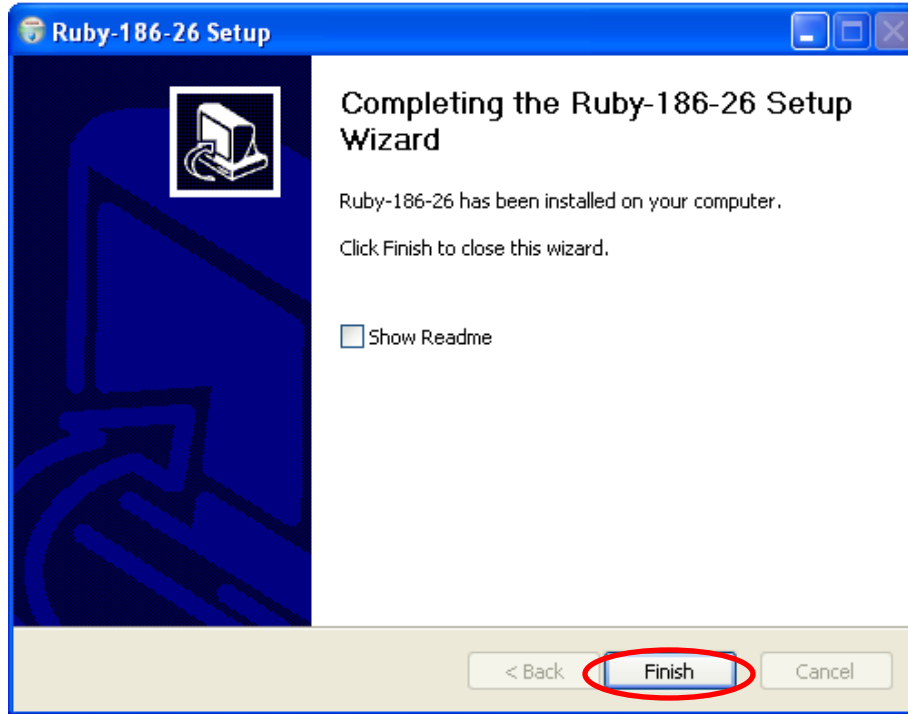
Latest File Releases				
Package	Version	Date	Notes / Monitor	Download
One-Click Installer - Windows	1.8.6-26 Final Release	December 14, 2007		Download

2. Run `ruby186-26.exe` and install in the folder `c:\Ruby`.





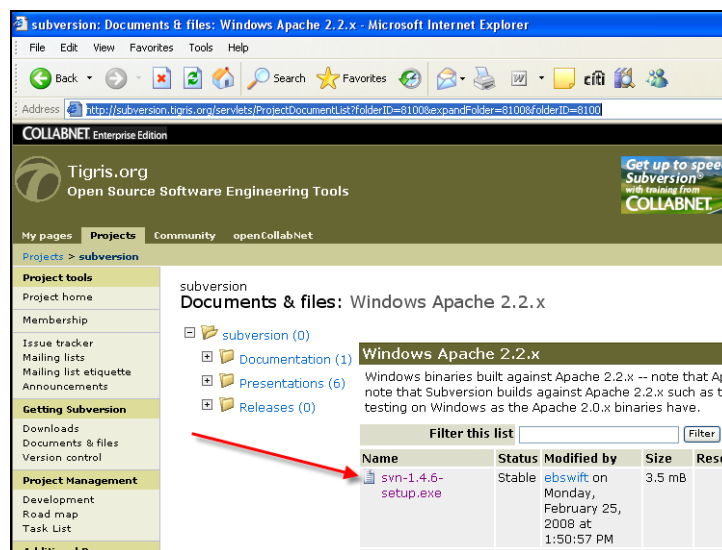




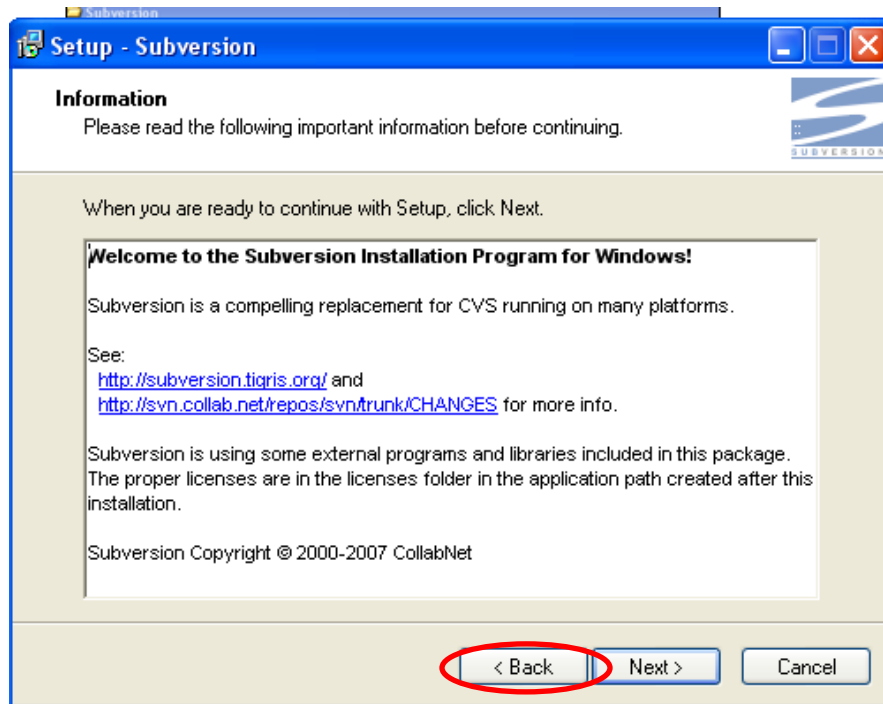
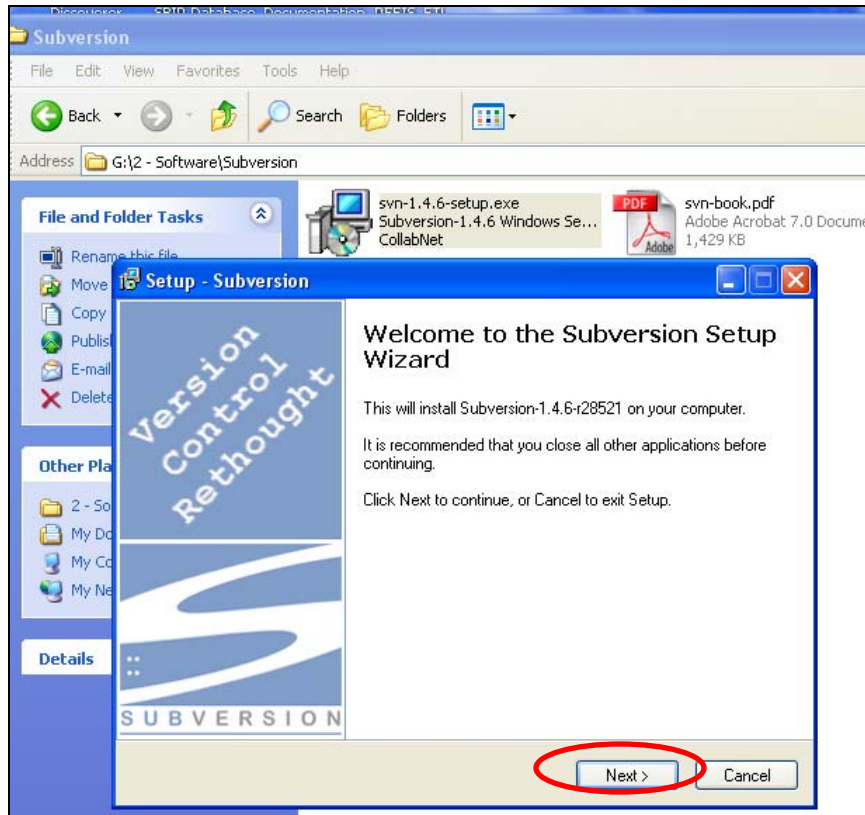
3. Download and Install the Subversion (SVN) version control client for Windows. (Hobo makes automatic calls using SVN to obtain other Ruby plugins (e.g., will_paginate) when you create a new application project.)

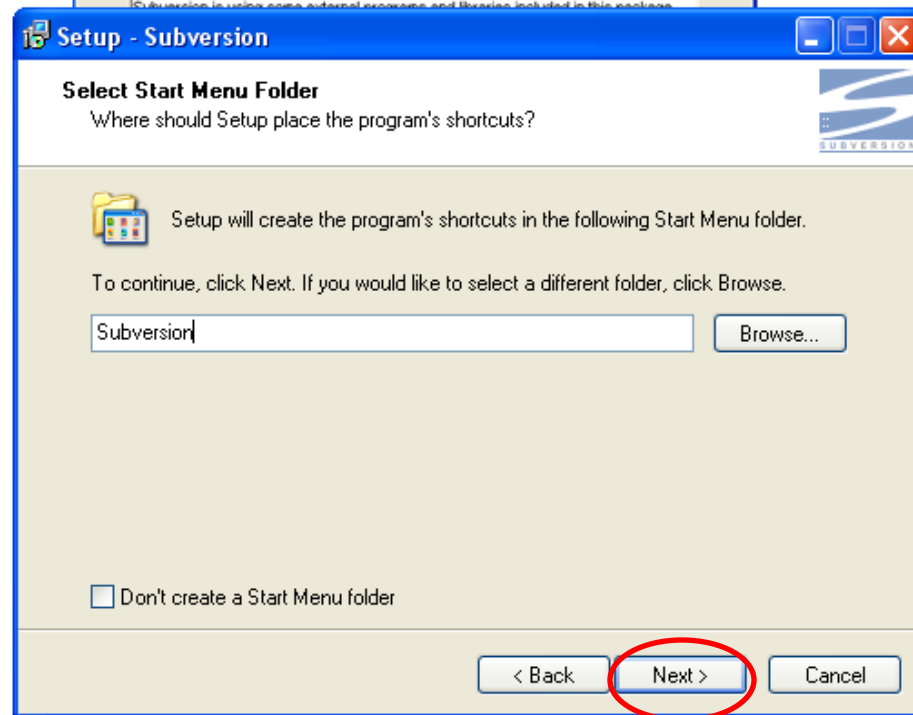
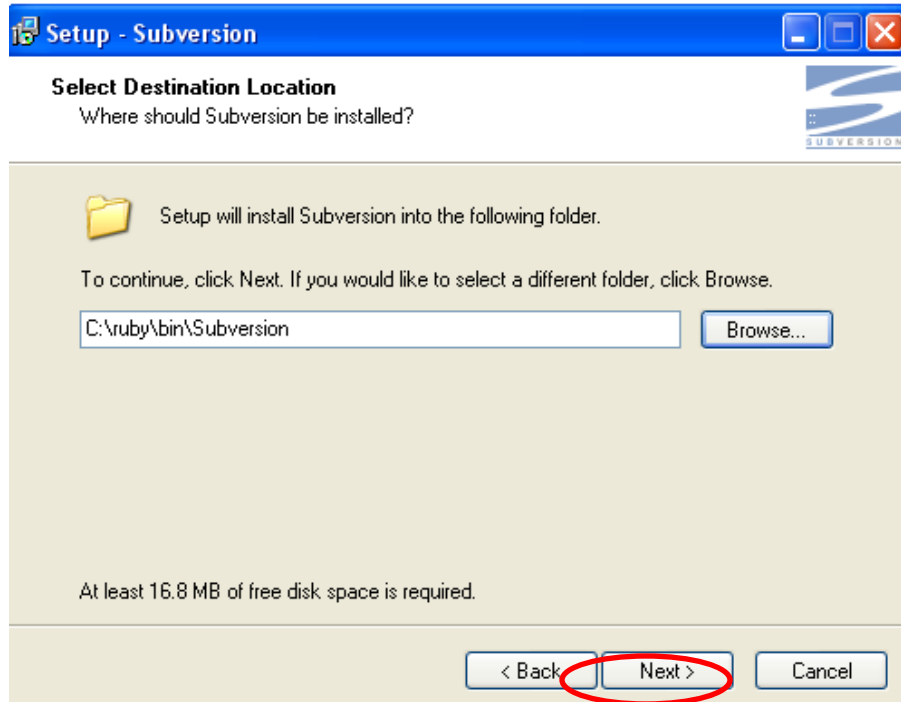
You can obtain the download using this URL:

<http://subversion.tigris.org/servlets/ProjectDocumentList?folderID=8100&expandFolder=8100&folderID=8100>



After you have downloaded the self-installing exe, double click to start the installation:

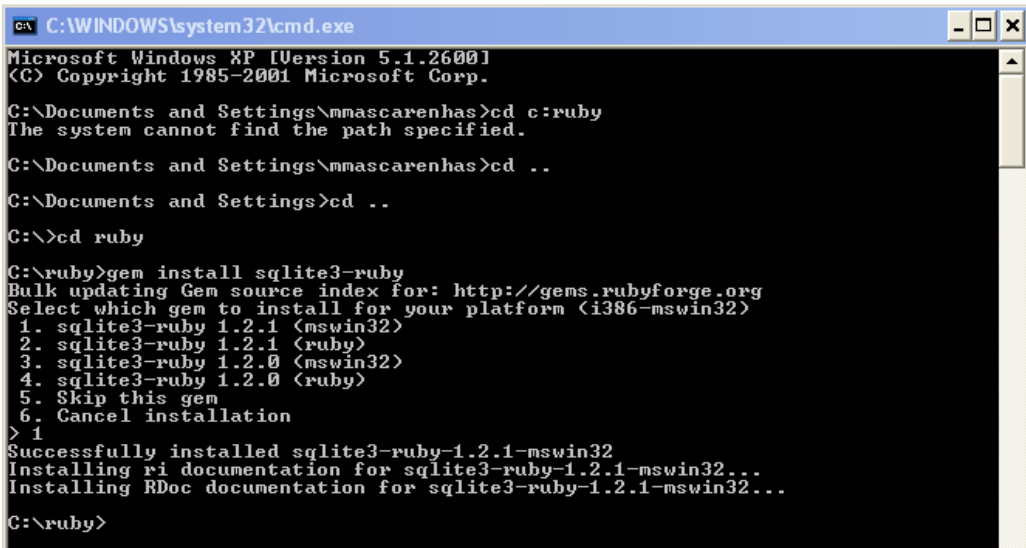




4. Install the **SQLite3** for Ruby gem using the “gem” command:

Open up the dos prompt (Start>Programs>Accessories>Command Prompt) and run the following from the c:\ruby folder.

```
C:\ruby> gem install sqlite3-ruby
```



Choose option 1 when prompted (mswin32)

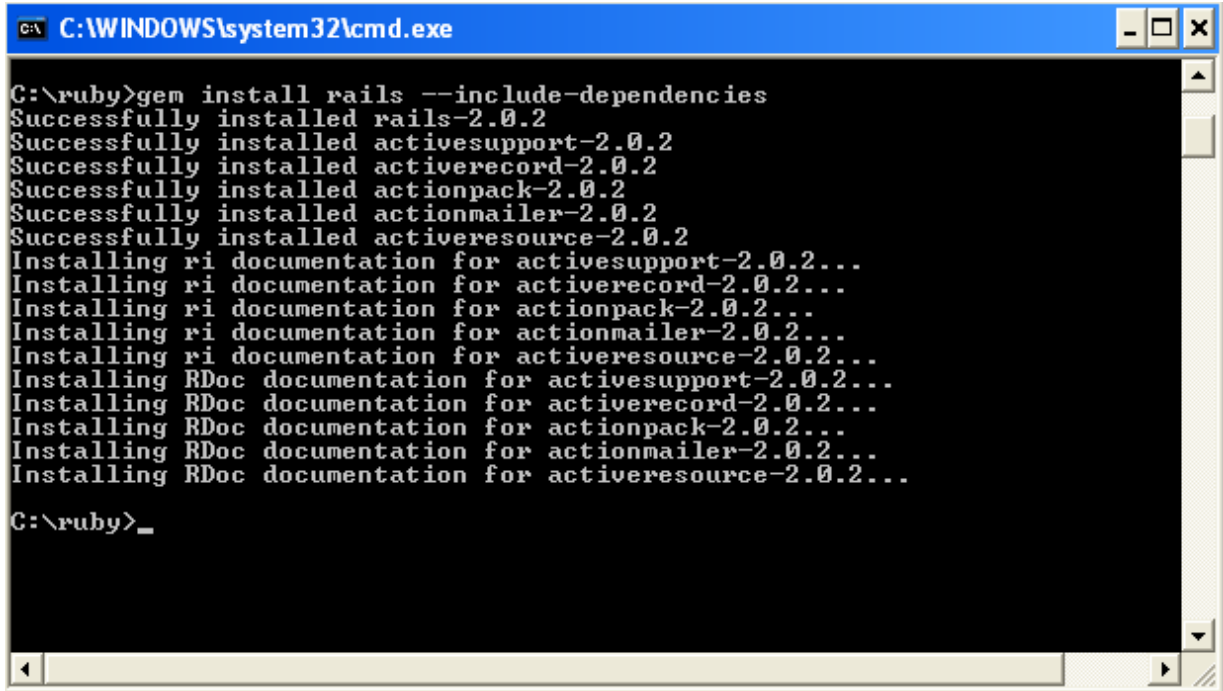
5. MS Windows PCs also require the **sqlite3.dll**. Download this from <http://www.sqlite.org/download.html> place it the “c:\ruby\bin” folder.

Unzip the downloaded file and place the **sqlite3.dll** under **c:\ruby\bin**

Precompiled Binaries For Windows	
sqlite-3_5_7.zip (209.56 KIB)	A command-line program for accessing and modifying SQLite databases. See the documentation for additional information.
tclsqlite-3_5_7.zip (278.66 KIB)	Bindings for Tcl/Tk . You can import this shared library into either tclsh or wish to get SQLite database access from Tcl/Tk. See the documentation for details.
sqllitedll-3_5_7.zip (207.88 KIB)	This is a DLL of the SQLite library without the TCL bindings. The only external dependency is MSVCRT.DLL.
sqlite3_analyzer-3_5_4.zip (540.03 KIB)	An analysis program for database files compatible with SQLite version 3.5.4 and later.

6. Install the latest version of Rails:

```
C:\ruby> gem install rails --include-dependencies
```



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

C:\ruby>gem install rails --include-dependencies
Successfully installed rails-2.0.2
Successfully installed activesupport-2.0.2
Successfully installed activerecord-2.0.2
Successfully installed actionpack-2.0.2
Successfully installed actionmailer-2.0.2
Successfully installed activeresource-2.0.2
Installing ri documentation for activesupport-2.0.2...
Installing ri documentation for activerecord-2.0.2...
Installing ri documentation for actionpack-2.0.2...
Installing ri documentation for actionmailer-2.0.2...
Installing ri documentation for activeresource-2.0.2...
Installing RDoc documentation for activesupport-2.0.2...
Installing RDoc documentation for activerecord-2.0.2...
Installing RDoc documentation for actionpack-2.0.2...
Installing RDoc documentation for actionmailer-2.0.2...
Installing RDoc documentation for activeresource-2.0.2...

C:\ruby>_
```

7. Install the Hobo Support gem:

```
C:\ruby> gem install hobosupport
```

8. Install the Hobo gem:

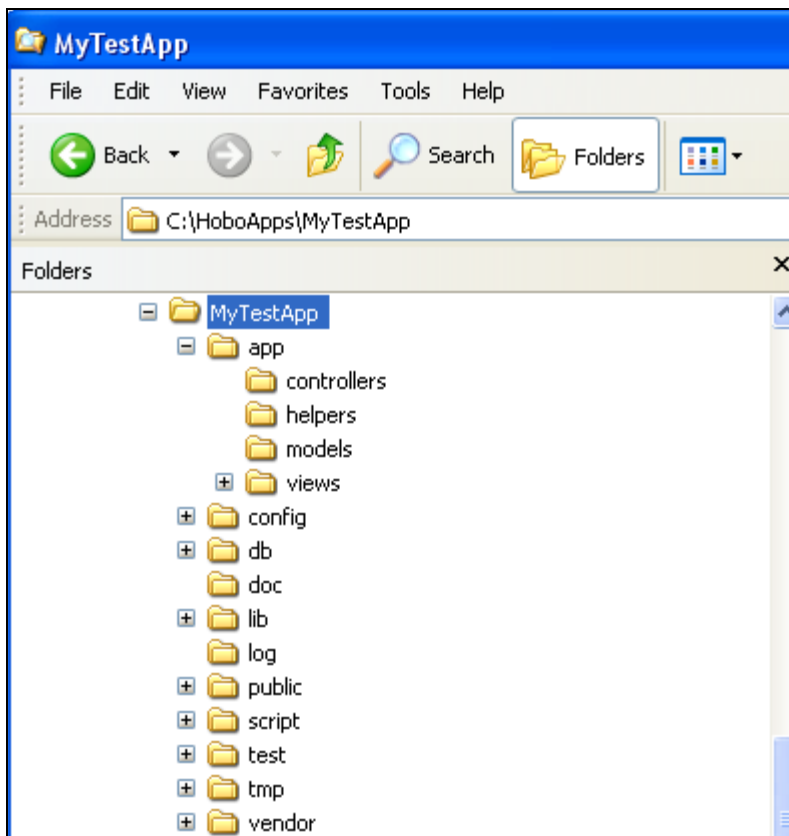
```
C:\ruby> gem install hobo
```

9. Create a master application folder (e.g., **HoboApps**) on your drive of choice (e.g., C: This folder will contain all your applications.

At the command prompt, create a new Hobo application—for example, **MyTestApp**. Without specifying a database, SQLite will be used as default:

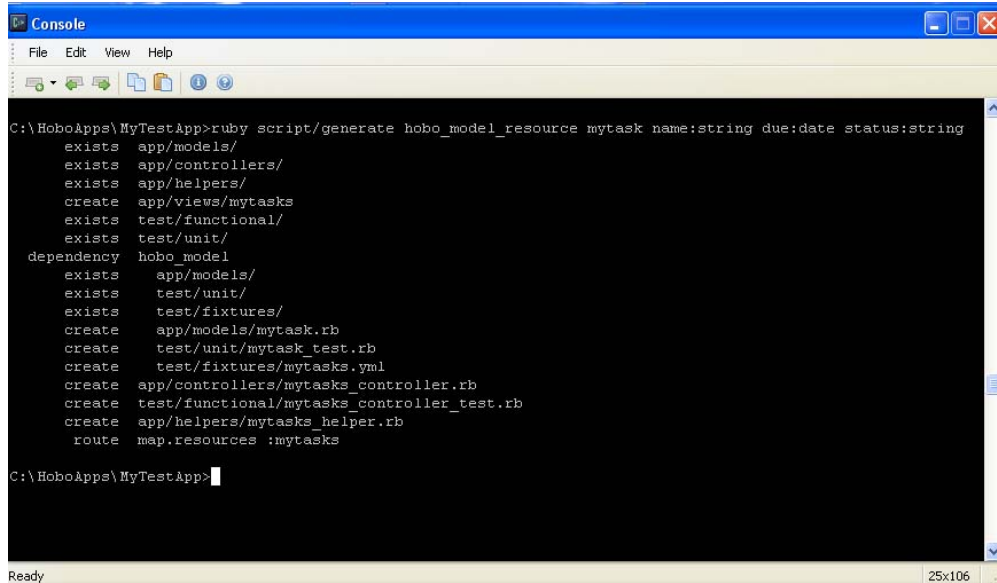
```
C:\HoboApps> Hobo MyTestApp
```

The following folders will be created under the Application folder.



10. Now prepare a simple data model using the **hobo_model_resource** command:

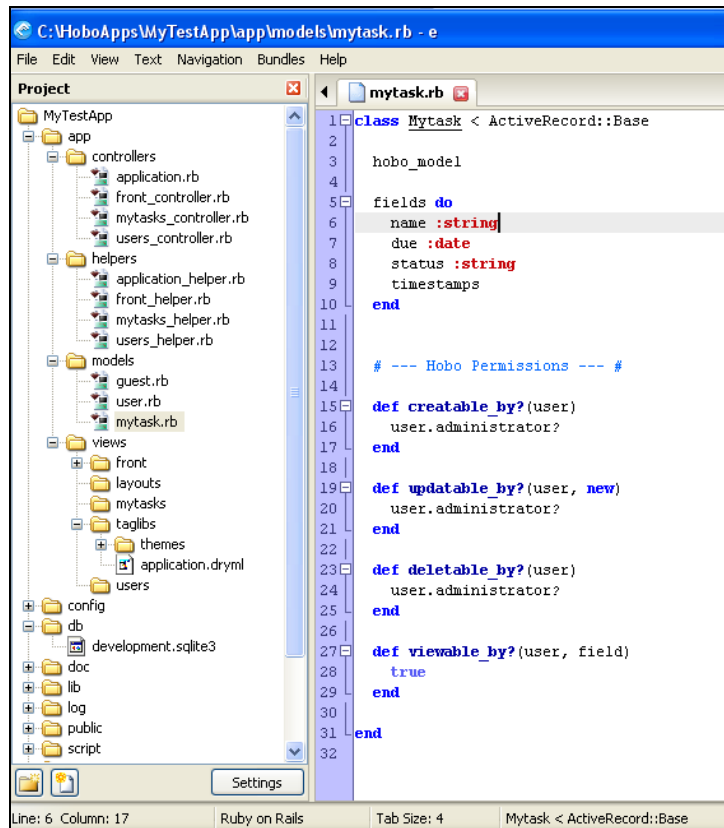
C:\HoboApps\MyTestApp> ruby script/generate hobo_model_resource mytask name:string due:date status:string



```
C:\HoboApps\MyTestApp>ruby script/generate hobo_model_resource mytask name:string due:date status:string
exists app/models/
exists app/controllers/
exists app/helpers/
create app/views/mytasks
exists test/functional/
exists test/unit/
dependency hobo_model
exists app/models/
exists test/unit/
exists test/fixtures/
create app/models/mytask.rb
create test/unit/mytask_test.rb
create test/fixtures/mytasks.yml
create app/controllers/mytasks_controller.rb
create test/functional/mytasks_controller_test.rb
create app/helpers/mytasks_helper.rb
route map.resources :mytasks

C:\HoboApps\MyTestApp>
```

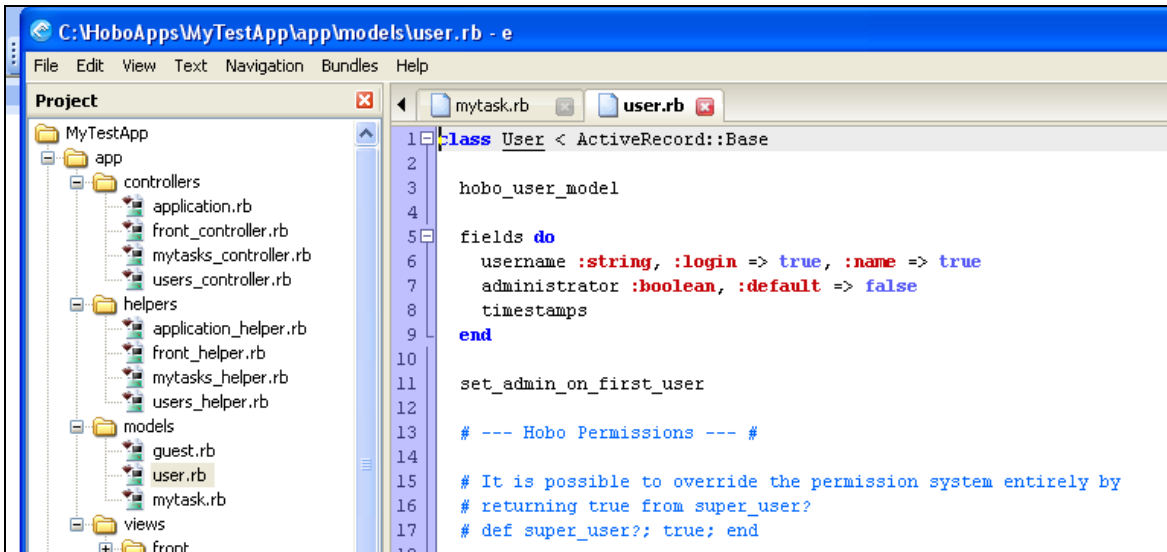
Notice what is created automatically:



```
C:\HoboApps\MyTestApp\app\models\mytask.rb - e
File Edit View Text Navigation Bundles Help
Project
MyTestApp
  app
    controllers
      application.rb
      front_controller.rb
      mytasks_controller.rb
      users_controller.rb
    helpers
      application_helper.rb
      front_helper.rb
      mytasks_helper.rb
      users_helper.rb
    models
      guest.rb
      user.rb
      mytask.rb
    views
      front
      layouts
      mytasks
      taglibs
      themes
      application.dryml
      users
  config
  db
  development.sqlite3
  doc
  lib
  log
  public
  script

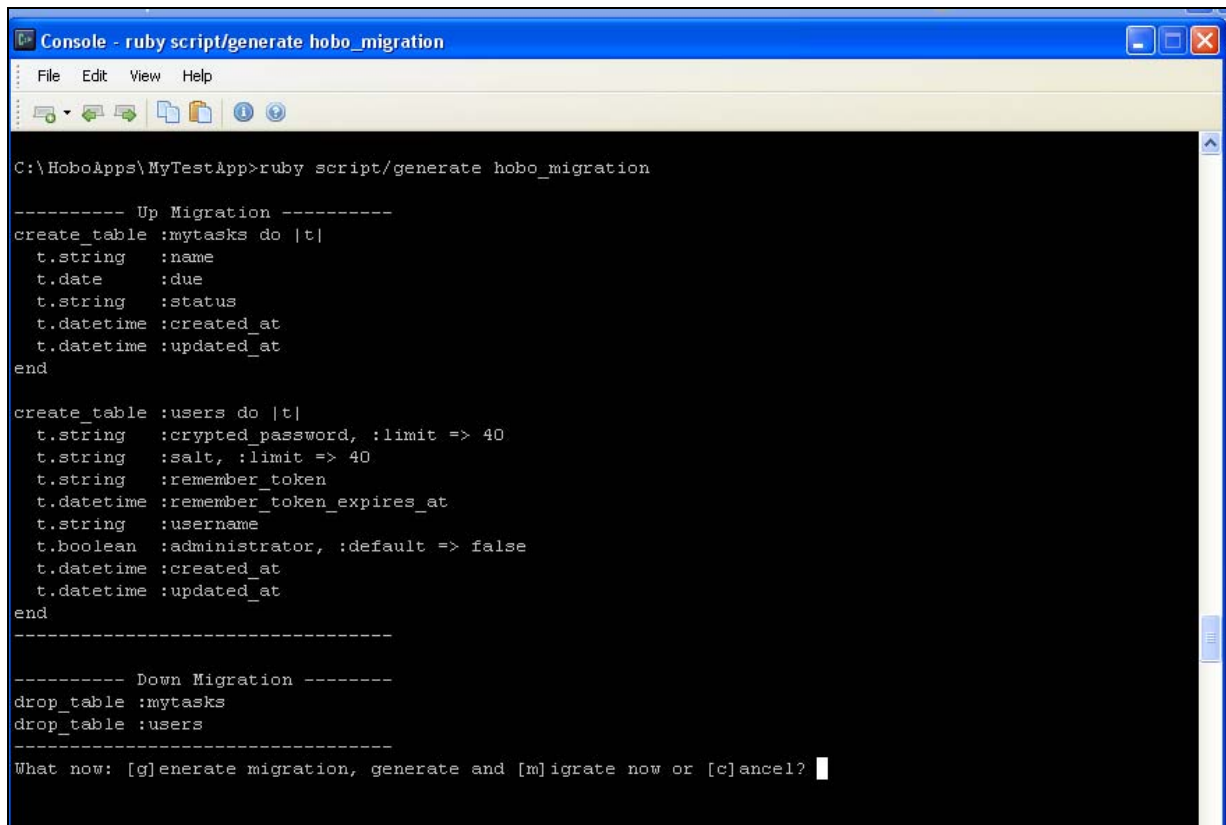
1 class Mytask < ActiveRecord::Base
2
3   hobo_model
4
5   fields do
6     name :string
7     due :date
8     status :string
9     timestamps
10  end
11
12  # --- Hobo Permissions --- #
13
14  def creatable_by?(user)
15    user.administrator?
16  end
17
18  def updatable_by?(user, new)
19    user.administrator?
20  end
21
22  def deletable_by?(user)
23    user.administrator?
24  end
25
26  def viewable_by?(user, field)
27    true
28  end
29
30 end
31
32
```

The Hobo user model was also created automatically by Hobo:



11. Generate the hobo_migration for these models:

```
C:\HoboApps\MyTestApp> ruby script/generate hobo_migration
```



Select the "g" (generate) option and press <ENTER> to select the default migration file name.

12. Now implement the migration using the “rake” command:

C:\HoboApps\MyTestApp> rake db:migrate

```
----- Down Migration -----
drop_table :mytasks
drop_table :users
-----

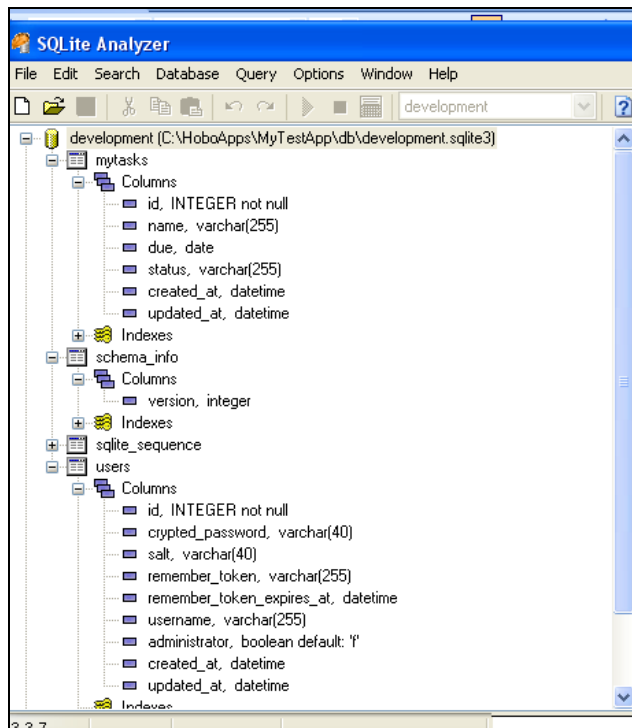
What now: [g]enerate migration, generate and [m]igrate now or [c]ancel? g

Migration filename:
(you can type spaces instead of '_' -- every little helps)
Filename [hobo migration_1]:
  create db/migrate
  create db/migrate/001_hobo_migration_1.rb

C:\HoboApps\MyTestApp>rake db:migrate
(in C:/HoboApps/MyTestApp)
== 1 HoboMigration1: migrating =====
-- create_table(:mytasks)
-> 0.0000s
-- create_table(:users)
-> 0.0160s
== 1 HoboMigration1: migrated (0.0160s) =====

C:\HoboApps\MyTestApp>
```

This will create the tables in the development SQLite database. You can use a tool like SQLite Analyzer to see a detailed view of what was created:



13. You could run the app now with the default WeBrick server, but Mongrel is a better performer. Install the server gem:

C:\Ruby `gem install mongrel --include-dependencies`

```

C:\WINDOWS\system32\cmd.exe
C:\>cd ruby
C:\ruby>gem install mongrel --include-dependencies
Need to update 1 gems from http://gems.rubyforge.org
.
complete
Select which gem to install for your platform <i386-mswin32>
1. mongrel 1.1.4 (x86-mswin32-60)
2. mongrel 1.1.4 (java)
3. mongrel 1.1.4 (ruby)
4. mongrel 1.1.3 (ruby)
5. mongrel 1.1.3 (i386-mswin32)
6. mongrel 1.1.3 (java)
7. Skip this gem
8. Cancel installation
> 1
Successfully installed mongrel-1.1.4-x86-mswin32-60
Successfully installed gem_plugin-0.2.3
Successfully installed cgi_multipart_eof_fix-2.5.0
Installing ri documentation for mongrel-1.1.4-x86-mswin32-60...
Installing ri documentation for gem_plugin-0.2.3...
Installing ri documentation for cgi_multipart_eof_fix-2.5.0...
Installing RDoc documentation for mongrel-1.1.4-x86-mswin32-60...
Installing RDoc documentation for gem_plugin-0.2.3...
Installing RDoc documentation for cgi_multipart_eof_fix-2.5.0...
* C:\ruby>
    
```

- Change your directory in your command prompt to root of the new application, e.g.,

C:\HoboApps\MyTestApp>

In the command prompt, type “ruby script/server”:

C:\HoboApps\MyTestApp> `ruby script/server`

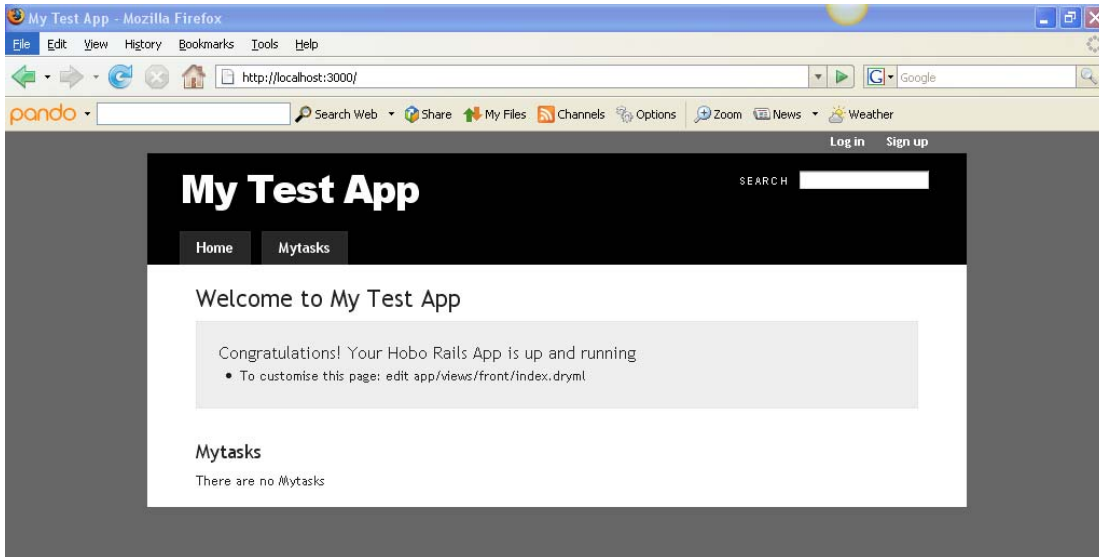
Now Ruby will boot the Mongrel server by default:

```

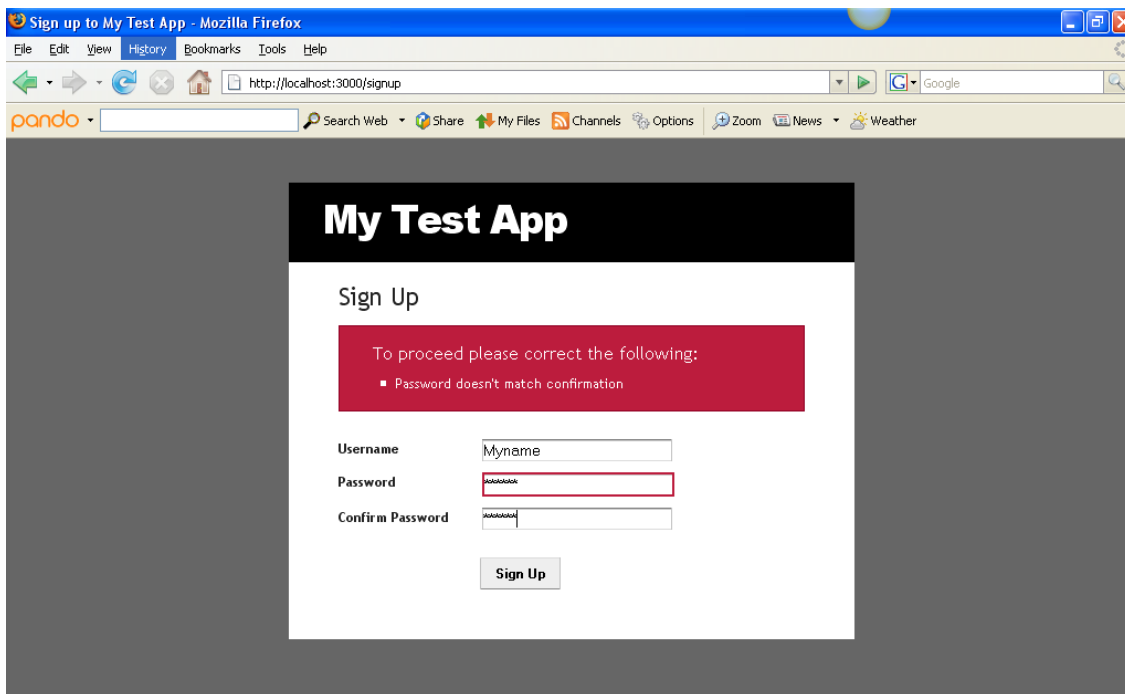
Console - ruby script/server
File Edit View Help
C:\HoboApps\MyTestApp>ruby script/server
=> Booting Mongrel (use 'script/server webrick' to force WEBrick)
=> Rails application starting on http://0.0.0.0:3000
=> Call with -d to detach
=> Ctrl-C to shutdown server
** Starting Mongrel listening at 0.0.0.0:3000
** Starting Rails with development environment...
** Rails loaded.
** Loading any Rails specific GemPlugins
** Signals ready. INT => stop (no restart).
** Mongrel 1.1.4 available at 0.0.0.0:3000
** Use CTRL-C to stop.
-[4;36;1mSQL (0.000000)-[0m -[0;1m SELECT name
FROM sqlite_master
WHERE type = 'table' AND NOT name = 'sqlite_sequence'
-[0m
    
```

15. Load your browser and enter the URL accessing the default port: <http://localhost:3000>

Notice that Hobo uses the application folder name as the default title:

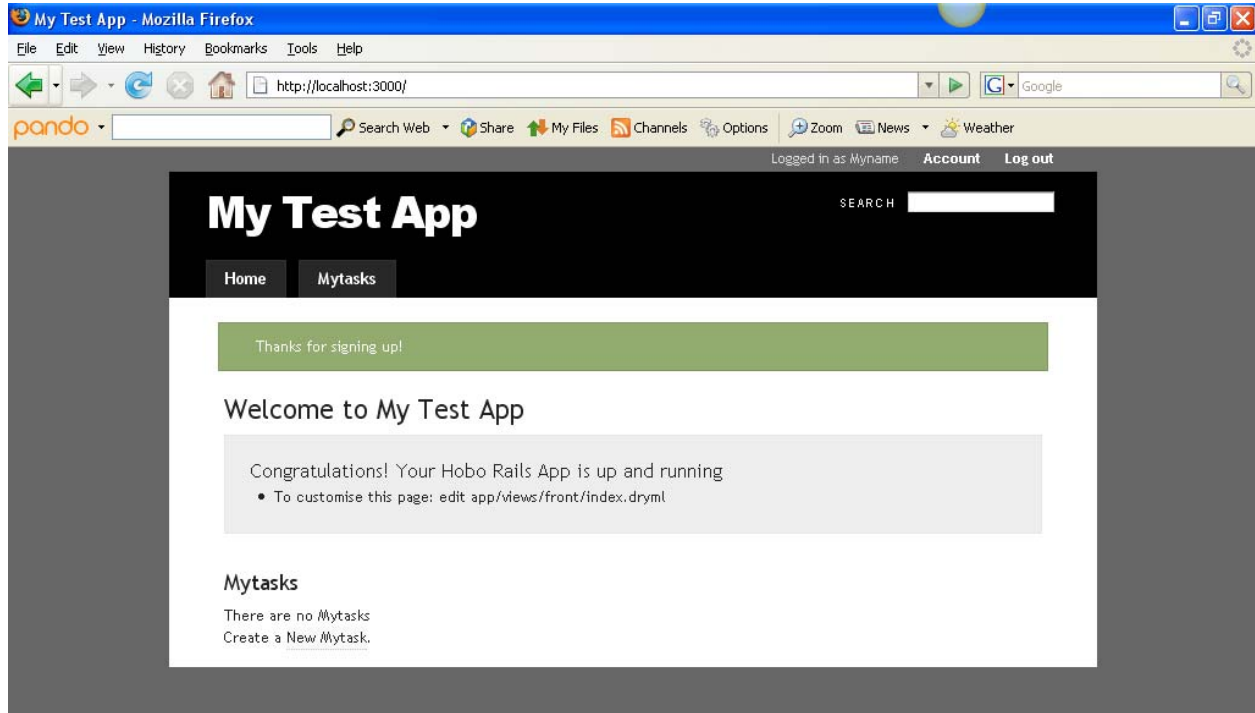


Choose the “Sign up” option. The first user to sign up will be given admin rights.

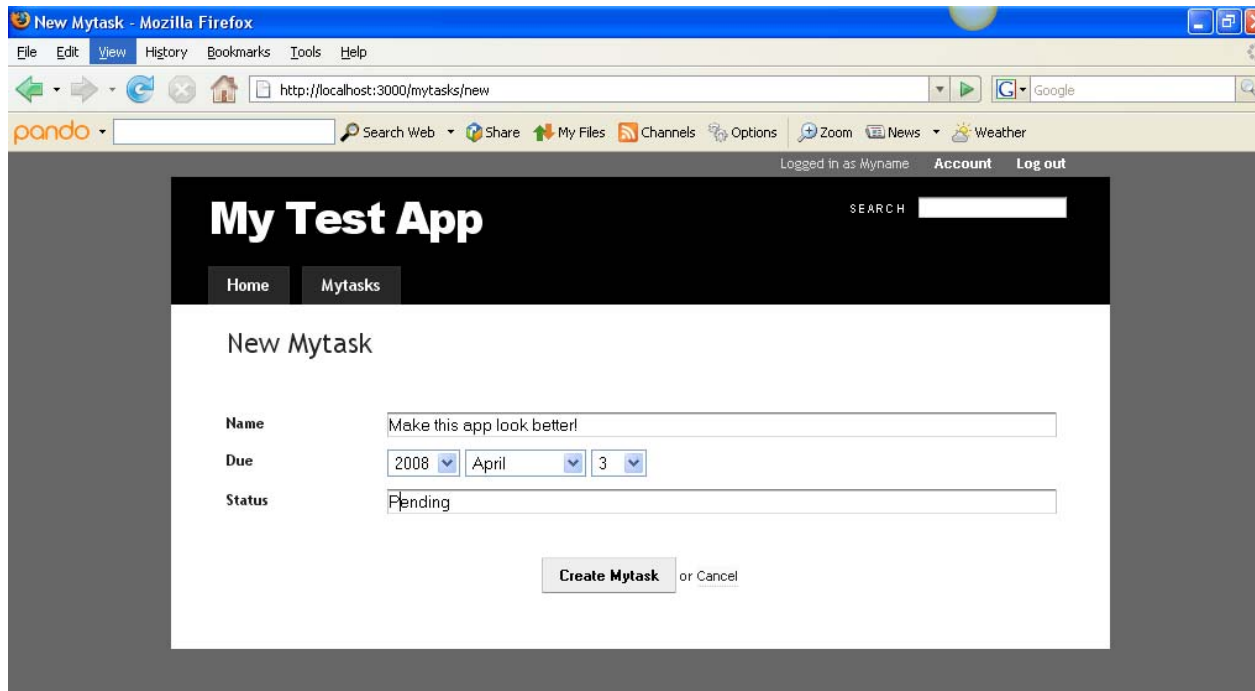


Password confirmation is also included:

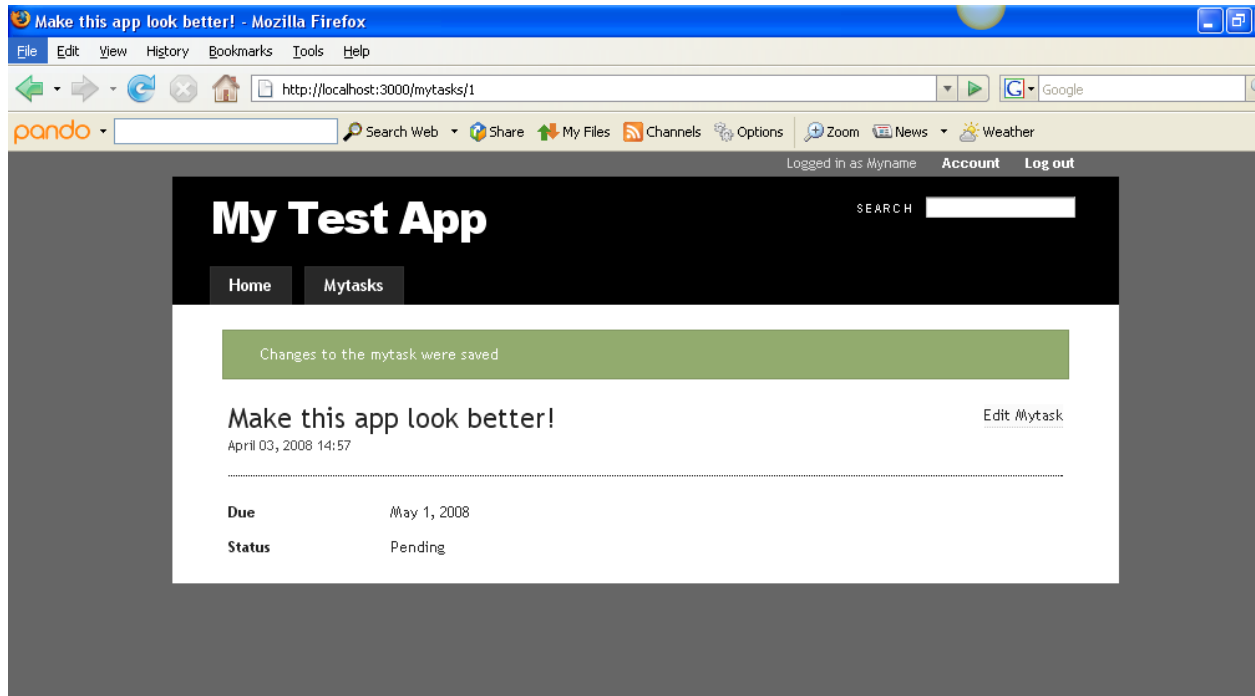
After successful log in:



Create a MyTask record:



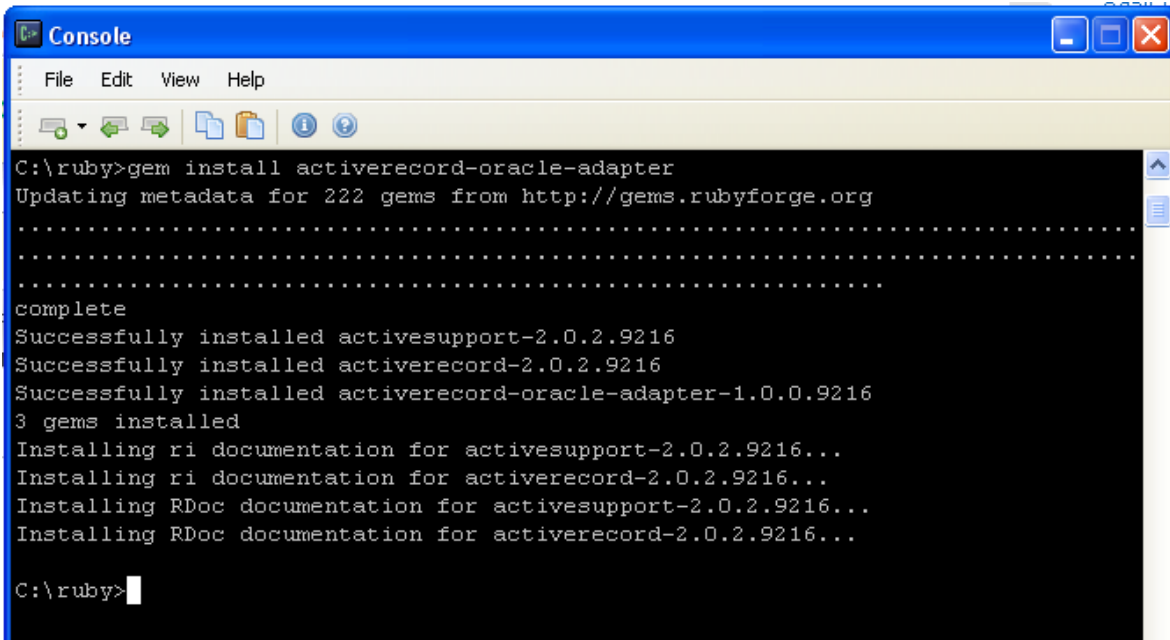
After creating an entry:



Part 2 of 2: What You Need to Use Oracle as the Database Engine

1. Install the Oracle adapter for ActiveRecord:

```
C:\Ruby gem install activerecord-oracle-adapter
```



```
Console
File Edit View Help
C:\ruby>gem install activerecord-oracle-adapter
Updating metadata for 222 gems from http://gems.rubyforge.org
.....
.....
complete
Successfully installed activesupport-2.0.2.9216
Successfully installed activerecord-2.0.2.9216
Successfully installed activerecord-oracle-adapter-1.0.0.9216
3 gems installed
Installing ri documentation for activesupport-2.0.2.9216...
Installing ri documentation for activerecord-2.0.2.9216...
Installing RDoc documentation for activesupport-2.0.2.9216...
Installing RDoc documentation for activerecord-2.0.2.9216...
C:\ruby>
```

2. Create a new Hobo application (for e.g.: MyOracleTestApp) with the database flag for using Oracle.

```
C:\HoboApps> Hobo -d oracle MyOracleTestApp
```

This will create the database.yml file (in c:\HoboApps\MyOracleTestApp\config\) with the correct parameters to fill in:



```
database.yml *
1 # Oracle/OCI 8i, 9, 10g
2 #
3 # Requires Ruby/OCI8:
4 # http://rubyforge.org/projects/ruby-oci8/
5 #
6 # Specify your database using any valid connection syntax, such as a
7 # tnsnames.ora service name, or a sql connect url string of the form:
8 #
9 # //host:[port][/]service name]
10 #
11
12 development:
13   adapter: oracle
14   database: MYDB
15   username: username
16   password: Password
17
```

See Oracle's [Using the Easy Connect Naming Method](#) documentation for the Easy Connect syntax. Alternatively, if the “c:/instantclient/tnsnames.ora” file contains:

```
MYDB =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = mymachine.mydomain)(PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = MYDB)
    )
  )
```

You can simply use “MYDB” when referring to your database. (See the database.yml entries in the screenshot above.)

Further details about using Oracle and Rails (e.g., creating a TNSNAMES.ORA file that points to your Oracle database instance) can be found at this URL:

<http://www.oracle.com/technology/pub/articles/saternos-ror-faq.html#4-3-3>