1. 下載Oracle VM
2. 下載Ubuntu
3. 新增VM  
   名稱：SaaScourse  
   類型：Linux  
   版本：ubuntu  
   Memory：1024  
   <http://vimeo.com/34754478>
4. 工具列->Devices->Install Guest Additions…  
   ---------------------------------新專案-------------------------------------------------
5. 新專案  
   rails new myrottenpotatoes
6. 新模組  
   rails generate controller welcome
7. 編輯Gemfile, 加入以下code
8. *# use Haml for templates*
9. gem 'haml'
10. *# use Ruby debugger*
11. group **:development**, **:test** **do**
12. gem 'ruby-debug19'
13. **end**
14. 修改Gemfile檔案之後，要執行bundle  
    bundle install
15. *config/routes.rb*新增設定,將首頁導至movies
16. **Myrottenpotatoes::Application**.routes.draw **do**
17. resources **:movies**
18. root **:to** **=>** redirect**(**'/movies'**)**
19. **end**
20. 增加controller class  
    路徑：app/views/controllers  
    def index  
    end
21. 增加view  
    路徑：app/views/movies  
    名稱：index.html.haml
22. 新增migration  
    rails g migration create\_movies
23. **class** CreateMovies **<** **ActiveRecord::Migration**
24. **def** up
25. create\_table 'movies' **do** **|**t**|**
26. t.**string** 'title'
27. t.**string** 'rating'
28. t.text 'description'
29. t.datetime 'release\_date'
30. *# Add fields that let Rails automatically keep track*
31. *# of when movies are added or modified:*
32. t.timestamps
33. **end**
34. **end**
36. **def** down
37. drop\_table 'movies' *# deletes the whole table and all its data!*
38. **end**
39. **end**
40. 執行ＳＱＬ  
    rake db:migrate
41. 新增model  
    路徑：app/models  
    名稱：movie.rb

c**lass** Movie **<** **ActiveRecord::Base**

1. **end**
2. rails console (異動資料庫)
3. *####  Create*
4. starwars = Movie.create!**(:title** **=>** 'Star Wars',
5. **:release\_date** **=>** '25/4/1977', **:rating** **=>** 'PG'**)**
6. *# note that numerical dates follow European format: dd/mm/yyyy*
7. requiem =  Movie.create!**(:title** **=>** 'Requiem for a Dream',
8. **:release\_date** **=>** 'Oct 27, 2000', **:rating** **=>** 'R'**)**
9. *#  Creation using separate 'save' method, used when updating existing records*
10. field = Movie.new**(:title** **=>** 'Field of Dreams',
11. **:release\_date** **=>** '21-Apr-89', **:rating** **=>** 'PG'**)**
12. field.save!
13. field.title = 'New Field of Dreams'
14. *####  Read*
15. pg\_movies = Movie.where**(**"rating = 'PG'"**)**
16. ancient\_movies = Movie.where**(**'release\_date < :cutoff and rating = :rating',
17. **:cutoff** **=>** 'Jan 1, 2000', **:rating** **=>** 'PG'**)**
18. pg\_movies\_2 = Movie.find\_by\_rating**(**'PG'**)**
19. Movie.find**(**3**)**  *# exception if key not found; find\_by\_id returns nil instead*
20. *####  Update*
21. starwars.update\_attributes**(:description** **=>** 'The best space western EVER',
22. **:release\_date** **=>** '25/5/1977'**)**
23. requiem.rating = 'NC-17'
24. requiem.save!
25. *####  Delete*
26. requiem.destroy
27. Movie.where**(**'title = "Requiem for a Dream"'**)**
28. *####  Find returns an enumerable*
29. Movie.find\_all\_by\_rating**(**'PG'**)**.each **do** **|**mov**|**
30. mov.destroy
31. **end**
32. 修改controllers
33. *# This file is app/controllers/movies\_controller.rb*
34. **class** MoviesController **<** ApplicationController
35. **def** index
36. **@movies** = Movie.all
37. **end**
38. **end**
39. 修改views
40. **%**h2 All Movies
42. **%**table*#movies*
43. **%**thead
44. **%**tr
45. **%**th Movie Title
46. **%**th Rating
47. **%**th Release **Date**
48. **%**th More Info
49. **%**tbody
50. **-** **@movies**.each **do** **|**movie**|**
51. **%**tr
52. **%**td= movie.title
53. **%**td= movie.rating
54. **%**td= movie.release\_date
55. **%**td= link\_to "More about #{movie.title}", movie\_path**(**movie**)**
56. 修改layout  
    路徑： app/views/layouts/application.html.erb  
      
      
      
      
      
    --------增加SHOW---------------------
57. !!! 5
58. **%**html
59. **%**head
60. **%**title Rotten Potatoes!
61. = stylesheet\_link\_tag 'application'
62. = javascript\_include\_tag 'application'
63. = csrf\_meta\_tags
65. **%**body
66. = **yield**
67. 修改controller
68. *# in app/controllers/movies\_controller.rb*
70. **def** show
71. id = params**[:id]** *# retrieve movie ID from URI route*
72. **@movie** = Movie.find**(**id**)** *# look up movie by unique ID*
73. *# will render app/views/movies/show.html.haml by default*
74. **end**
75. 增加views
76. **-***# in app/views/movies/show.html.haml*
78. **%**h2 Details about *#{@movie.title}*
80. **%**ul*#details*
81. **%**li
82. Rating:
83. = **@movie**.rating
84. **%**li
85. Released on:
86. = **@movie**.release\_date.strftime**(**"%B %d, %Y"**)**
88. **%**h3 Description:
90. **%p***#description= @movie.description*
92. = link\_to 'Back to movie list', movies\_path
93. 修改CSS樣式
94. */\* Simple CSS styling for RottenPotatoes app \*/*
95. */\* Add these lines to app/assets/stylesheets/application.css \*/*
97. html, body {
98. **margin**: 0;
99. **padding**: 0;
100. **background**: White;
101. **color**: DarkSlateGrey;
102. **font-family**: Tahoma, Verdana, sans-serif;
103. **font-size**: 10pt;
104. }
105. div#main {
106. **margin**: 0;
107. **padding**: 0 20px 20px;
108. }
109. a {
110. **background**: transparent;
111. **color**: maroon;
112. **text-decoration**: underline;
113. **font-weight**: bold;
114. }
115. h1 {
116. **color**: maroon;
117. **font-size**: 150%;
118. **font-style**: italic;
119. **display**: block;
120. **width**: 100%;
121. **border-bottom**: 1px solid DarkSlateGrey;
122. }
123. h1.title {
124. **margin**: 0 0 1em;
125. **padding**: 10px;
126. **background-color**: orange;
127. **color**: white;
128. **border-bottom**: 4px solid gold;
129. **font-size**: 2em;
130. **font-style**: normal;
131. }
132. table#movies {
133. **margin**: 10px;
134. **border-collapse**: collapse;
135. **width**: 100%;
136. **border-bottom**: 2px solid **black**;
137. }
138. table#movies th {
139. **border**: 2px solid white;
140. **font-weight**: bold;
141. **background-color**: wheat;
142. }
143. table#movies th, table#movies td {
144. **padding**: 4px;
145. **text-align**: **left**;
146. }
147. #notice, #warning {
148. **background**: rosybrown;
149. **margin**: 1em 0;
150. **padding**: 4px;
151. }
152. form label {
153. **display**: block;
154. **line-height**: 25px;
155. **font-weight**: bold;
156. **color**: maroon;
157. }
158. 首頁增加“新增電影”連結
159. **-***# add to end of index.html.haml*
161. = link\_to 'Add new movie', new\_movie\_path
162. 在 controller增加class new
163. **def** new
164. *# default: render 'new' template*
165. **end**
166. 設計new頁面  
     路徑：app/views/movies/new.html.haml  
       
       
       
       
       
     ------------新增create相關----------------------------
167. **%**h2 Create New Movie
169. = form\_tag movies\_path, **:method** **=>** **:post** **do**
171. = label **:movie**, **:title**, 'Title'
172. = text\_field **:movie**, **:title**
174. = label **:movie**, **:rating**, 'Rating'
175. = **select** **:movie**, **:rating**, **[**'G','PG','PG-13','R','NC-17'**]**
177. = label **:movie**, **:release\_date**, 'Released On'
178. = date\_select **:movie**, **:release\_date**
180. = submit\_tag 'Save Changes'
181. 新增create view  
     路徑：app/views/movies/create.html.haml
182. 新增create至controller  
       
       
       
       
       
       
       
       
     ------------新增flash相關----------------------------
183. *# in movies\_controller.rb*
184. **def** create
185. **@movie** = Movie.create!**(**params**[:movie])**
186. redirect\_to movies\_path
187. **end**
188. add flash
189. *# in movies\_controller.rb*
190. **def** create
191. **@movie** = Movie.create!**(**params**[:movie])**
192. flash**[:notice]** = "#{@movie.title} was successfully created."
193. redirect\_to movies\_path
194. **end**
195. 修改layouut  
     路徑： app/views/layouts/application.html.erb
196. **-***# this goes just inside %body:*
197. **-** **if** flash**[:notice]**
198. *#notice.message= flash[:notice]*
199. **-** **elsif** flash**[:warning]**
200. *#warning.message= flash[:warning]*
201. 修改notice的底色  
     路徑： app/views/assets/stylesheets/application.css
202. 新增連結：編輯，返回,delete

**-** = button\_to 'Delete', movie\_path**(**@movie**)**, **:method** **=>** **:delete**, **:confirm** **=>** 'Are you sure?'

1. **-***# modify last 2 lines of app/views/movies/show.html.haml to:*
2. = link\_to 'Edit info', edit\_movie\_path**(**@movie**)**
3. = link\_to 'Back to movie list', movies\_path
4. edit controller
5. **def** destroy
6. **@movie** = Movie.find**(**params**[:id])**
7. **@movie**.destroy
8. flash**[:notice]** = "Movie '#{@movie.title}' deleted."
9. redirect\_to movies\_path
10. **end**
11. *# in movies\_controller.rb*
13. **def** edit
14. **@movie** = Movie.find params**[:id]**
15. **end**
17. **def** update
18. **@movie** = Movie.find params**[:id]**
19. **@movie**.update\_attributes!**(**params**[:movie])**
20. flash**[:notice]** = "#{@movie.title} was successfully updated."
21. redirect\_to movie\_path**(**@movie**)**
22. **end**
23. add edit.html.haml  
    ----------------Cucumber-------------------------
24. 為了測試，修改Gemfile
25. *# add to end of Gemfile*
26. group **:test**, **:development** **do**
27. gem 'cucumber-rails'
28. gem 'cucumber-rails-training-wheels' *# some pre-fabbed step definitions*
29. gem 'database\_cleaner' *# to clear Cucumber's test database between runs*
30. gem 'capybara'         *# lets Cucumber pretend to be a web browser*
31. gem 'launchy'          *# a useful debugging aid for user stories*
32. **end**
33. 修改Gemfile檔案之後，要執行bundle  
    bundle install
34. 安裝cucumber  
    rails generate cucumber:install capybara  
    rails generate cucumber\_rails\_training\_wheels:install
35. 初始化測試資料庫  
    rake db:test:prepare  
    -----影片範例-----------  
    http://vimeo.com/34754747
36. 新增feature  
    路徑：features/step\_definitions/Addmovie.feature
37. Feature: User can manually add movie
39. Scenario: Add a movie
40. Given I am on the RottenPotatoes home page
41. When I follow "Add new movie"
42. Then I should be on the Create New Movie page
43. When I fill in "Title" with "Men In Black"
44. And I select "PG-13" from "Rating"
45. And I press "Save Changes"
46. Then I should be on the RottenPotatoes home page
47. And I should see "Men In Black"
48. 修改path.rb
49. 執行cucumber  
    ----------5-6 新增查詢test case-----------------  
    影片：  
    <http://vimeo.com/34754766>  
    ----------ch 6 TDD---------------------------  
    影片  
    http://vimeo.com/34754856
50. 安裝環境  
    1) 修改gemfile   
     在group:test下新增  
    gem ‘rspec-rails  
    u﷽﷽﷽﷽﷽﷽﷽﷽﷽﷽﷽﷽Test’  
    gem 'ZenTest'  
    2) 執行bundle install  
    3) 在專案app資料夾路徑下執行  
    rails g rspec:install
51. 新增檔案movies\_controller\_spec.rb  
    路徑：/spec/controllers
52. **require** 'spec\_helper'
54. describe MoviesController **do**
55. describe 'searching TMDb' **do**
56. it 'should call the model method that performs TMDb search'
57. it 'should select the Search Results template for rendering'
58. it 'should make the TMDb search results available to that template'
59. **end**
60. **end**
61. 執行  
    rspec -c -fn spec/controllers/movies\_controller\_spec.rb
62. 在movies\_controller\_spec.rb增加prnding
63. 執行autotest
64. 在movies\_controller\_spec.rb增加flurk
65. 執行  
    rspec -c -fn spec/controllers/movies\_controller\_spec.rb  
    ---------------------------------------------  
    練習測試影片  
    <http://vimeo.com/34754876>
66. 在movies\_controller.rb的def search\_tmdb加上  
    **@movies** = Movie.find\_in\_tmdb**(**params**[:search\_terms])**
67. 將movies\_controller\_spec.rb改為  
    -----------ch 6-4-----------------------
68. **require** 'spec\_helper'
70. describe MoviesController **do**
71. describe 'searching TMDb' **do**
72. it 'should call the model method that performs TMDb search' **do**
73. fake\_results = **[**mock**(**'movie1'**)**, mock**(**'movie2'**)]**
74. Movie.should\_receive**(:find\_in\_tmdb)**.with**(**'hardware'**)**.
75. and\_return**(**fake\_results**)**
76. post **:search\_tmdb**, **{**:search\_terms **=>** 'hardware'**}**
77. **end**
78. it 'should select the Search Results template for rendering'
79. it 'should make the TMDb search results available to that template'
80. **end**
81. **end**
82. 將 movies\_controller\_spec.rb改為
83. **require** 'spec\_helper'
85. describe MoviesController **do**
86. describe 'searching TMDb' **do**
87. it 'should call the model method that performs TMDb search' **do**
88. fake\_results = **[**mock**(**'movie1'**)**, mock**(**'movie2'**)]**
89. Movie.should\_receive**(:find\_in\_tmdb)**.with**(**'hardware'**)**.
90. and\_return**(**fake\_results**)**
91. post **:search\_tmdb**, **{**:search\_terms **=>** 'hardware'**}**
92. **end**
93. it 'should select the Search Results template for rendering' **do**
94. fake\_results = **[**mock**(**'Movie'**)**, mock**(**'Movie'**)]**
95. Movie.stub**(:find\_in\_tmdb)**.and\_return**(**fake\_results**)**
96. post **:search\_tmdb**, **{**:search\_terms **=>** 'hardware'**}**
97. response.should render\_template**(**'search\_tmdb'**)**
98. **end**
99. it 'should make the TMDb search results available to that template'
100. **end**
101. **end**
102. 將 movies\_controller\_spec.rb改為（使用before）
103. **require** 'spec\_helper'
105. describe MoviesController **do**
106. describe 'searching TMDb' **do**
107. before **:each** **do**
108. **@fake\_results** = **[**mock**(**'movie1'**)**, mock**(**'movie2'**)]**
109. **end**
110. it 'should call the model method that performs TMDb search' **do**
111. Movie.should\_receive**(:find\_in\_tmdb)**.with**(**'hardware'**)**.
112. and\_return**(**@fake\_results**)**
113. post **:search\_tmdb**, **{**:search\_terms **=>** 'hardware'**}**
114. **end**
115. it 'should select the Search Results template for rendering' **do**
116. Movie.stub**(:find\_in\_tmdb)**.and\_return**(**@fake\_results**)**
117. post **:search\_tmdb**, **{**:search\_terms **=>** 'hardware'**}**
118. response.should render\_template**(**'search\_tmdb'**)**
119. **end**
120. it 'should make the TMDb search results available to that template'
121. **end**
122. **end**
123. 將 movies\_controller\_spec.rb改為（增加after）  
     ------------ch 6-5--------------------------------
124. **require** 'spec\_helper'
126. describe MoviesController **do**
127. describe 'searching TMDb' **do**
128. before **:each** **do**
129. **@fake\_results** = **[**mock**(**'movie1'**)**, mock**(**'movie2'**)]**
130. **end**
131. it 'should call the model method that performs TMDb search' **do**
132. Movie.should\_receive**(:find\_in\_tmdb)**.with**(**'hardware'**)**.
133. and\_return**(**@fake\_results**)**
134. post **:search\_tmdb**, **{**:search\_terms **=>** 'hardware'**}**
135. **end**
136. describe 'after valid search' **do**
137. before **:each** **do**
138. Movie.stub**(:find\_in\_tmdb)**.and\_return**(**@fake\_results**)**
139. post **:search\_tmdb**, **{**:search\_terms **=>** 'hardware'**}**
140. **end**
141. it 'should select the Search Results template for rendering' **do**
142. response.should render\_template**(**'search\_tmdb'**)**
143. **end**
144. it 'should make the TMDb search results available to that template' **do**
145. assigns**(:movies)**.should == **@fake\_results**
146. **end**
147. **end**
148. **end**
149. **end**
150. 新增 spec/models/movie\_spec.rb
151. *# spec/models/movie\_spec.rb:*
153. describe Movie **do**
154. fixtures **:movies**
155. it 'should include rating and year in full name' **do**
156. movie = movies**(:milk\_movie)**
157. movie.name\_with\_rating.should == 'Milk (R)'
158. **end**
159. **end**
160. 新增spec/factories/movie.rb
161. *# spec/factories/movie.rb*
163. FactoryGirl.define **do**
164. factory **:movie** **do**
165. title 'A Fake Title' *# default values*
166. rating 'PG'
167. release\_date **{** 10.years.ago **}**
168. **end**
169. **end**
170. 修改spec/models/movie\_spec.rb
171. *# spec/models/movie\_spec.rb*
173. describe Movie **do**
174. it 'should include rating and year in full name' **do**
175. movie = FactoryGirl.build**(:movie**, **:title** **=>** 'Milk', **:rating** **=>** 'R'**)**
176. movie.name\_with\_rating.should == 'Milk (R)'
177. **end**
178. **end**
180. *# or if you mix in FactoryGirl's syntax methods (see FactoryGirl README):*
182. describe Movie **do**
183. subject **{** create **:movie**, **:title** **=>** 'Milk', **:rating** **=>** 'R' **}**
184. its **:name\_with\_rating** **{** should == 'Milk (R)' **}**
185. **end**
186. 新增gemfile  
     gem 'factory\_girl\_rails'  
       
       
     ---------ch7-1 --------------------
187. ch7-1 Partials