

A Rails keretrendszer Gyakorlat

Kovács Gábor

2018. szeptember 25.

1. Ruby telepítése, Gemek

A Rails keretrendszert egy virtuális környezetben ¹ futó Debian Linuxra ² telepítjük a gyakorlat folyamán. Ez a dokumentáció kiindulási pontként egy ilyen, grafikus felhasználói környezetet is tartalmazó előtelepített operációs rendszert feltételez.

A Rails környezetünk back-endje hosszú távon egy MySQL adatbázis kezelő lesz. Ezt az `apt-get install mysql-server` parancs konzolon történő kiadásával telepíthetjük, a telepítő többször rá fog kérdezni a root felhasználó jelszavára, azt tetszés szerinti értékre állíthatjuk, azonban jegyezzük meg, mert a Rails adatbázis konfigurációs fájljában ezt meg kell adnunk. A gyakorlatra telepített környezeten az egyszerűség kedvéért üres jelszót fogunk használni.

A Rails adatbázis adapterei és néhány további komponens fordításához szükségünk van C és C++ fordítóra, valamint a fejlesztői függvénykönyvtárakra ³, amelyeket az `apt-get install gcc` és `apt-get install g++` parancsok konzolon való kiadásával telepíthetünk.

A Rails környezet telepítéséhez egy Ruby környezet szükséges, ami mellé kiegészítőként telepítjük az RI dokumentációs rendszert és a natív kiegészítések fordításához szükséges Ruby fejlesztői függvénykönyvtárakat. A gyakorlaton a Ruby stabil, 2.3-s verzióját használjuk, amely előfeltétele a tavaly megjelent Rails 5-ös változatának telepítésének. A C függvénykönyvtárakhoz való illesztéshez szükségünk lesz a Ruby header fájlokra is, ezért telepítjük

¹Oracle VirtualBox (<http://www.virtualbox.org>) az otthonra javasolt virtualizációs eszköz, viszont használható a vmware, illetve a parallels is

²Mivel az Ubuntu Linux csomagkezelője megegyezik a Debian Linux disztribúció csomagkezelőjével, a dokumentumban felsorolt csomagnevek egy az egyben átvehetők. Más disztribúciók esetén a csomagok nevei különbözhetnek.

³A Rails egyik függvénykönyvtára miatt szükségünk lesz még a `zlib1g` csomagra is.

azok Linux csomagját is. Ezt a következő paranccsal telepíthetjük rendszer-gazdaként:

```
kovacs@debian:~# ruby -v
bash: ruby: command not found
kovacs@debian:~# sudo bash
[sudo] password for kovacs:
root@debian:/home/kovacs# su -
root@debian:~# apt-get install ruby ruby-dev ri
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  fonts-lato libgmp-dev libgmpxx4ldbl libruby2.3 rake ruby-did-you-mean ruby-
  -minitest
  ruby-net-telnet ruby-power-assert ruby-test-unit ruby2.3 ruby2.3-dev ruby2
  .3-doc
  rubygems-integration
Suggested packages:
  gmp-doc libgmp10-doc libmpfr-dev bundler
The following NEW packages will be installed:
  fonts-lato libgmp-dev libgmpxx4ldbl libruby2.3 rake ri ruby ruby-dev ruby-
  did-you-mean
  ruby-minitest ruby-net-telnet ruby-power-assert ruby-test-unit ruby2.3
  ruby2.3-dev
  ruby2.3-doc rubygems-integration
0 upgraded, 17 newly installed, 0 to remove and 20 not upgraded.
Need to get 0 B/11.6 MB of archives.
After this operation, 61.9 MB of additional disk space will be used.
Do you want to continue? [Y/n]
Selecting previously unselected package fonts-lato.
(Reading database ... 220739 files and directories currently installed.)
Preparing to unpack .../00-fonts-lato_2.0-1_all.deb ...
Unpacking fonts-lato (2.0-1) ...
Selecting previously unselected package libgmpxx4ldbl:amd64.
Preparing to unpack .../01-libgmpxx4ldbl_2%3a6.1.2+dfsg-1_amd64.deb ...
Unpacking libgmpxx4ldbl:amd64 (2:6.1.2+dfsg-1) ...
Selecting previously unselected package libgmp-dev:amd64.
Preparing to unpack .../02-libgmp-dev_2%3a6.1.2+dfsg-1_amd64.deb ...
Unpacking libgmp-dev:amd64 (2:6.1.2+dfsg-1) ...
Selecting previously unselected package rubygems-integration.
Preparing to unpack .../03-rubygems-integration_1.11_all.deb ...
Unpacking rubygems-integration (1.11) ...
Selecting previously unselected package ruby-did-you-mean.
Preparing to unpack .../04-ruby-did-you-mean_1.0.0-2_all.deb ...
Unpacking ruby-did-you-mean (1.0.0-2) ...
Selecting previously unselected package ruby-minitest.
Preparing to unpack .../05-ruby-minitest_5.9.0-1_all.deb ...
Unpacking ruby-minitest (5.9.0-1) ...
Selecting previously unselected package ruby-net-telnet.
Preparing to unpack .../06-ruby-net-telnet_0.1.1-2_all.deb ...
Unpacking ruby-net-telnet (0.1.1-2) ...
Selecting previously unselected package ruby-power-assert.
Preparing to unpack .../07-ruby-power-assert_0.3.0-1_all.deb ...
Unpacking ruby-power-assert (0.3.0-1) ...
Selecting previously unselected package ruby-test-unit.
Preparing to unpack .../08-ruby-test-unit_3.1.7-2_all.deb ...
Unpacking ruby-test-unit (3.1.7-2) ...
Selecting previously unselected package libruby2.3:amd64.
Preparing to unpack .../09-libruby2.3_2.3.3-1+deb9u2_amd64.deb ...
Unpacking libruby2.3:amd64 (2.3.3-1+deb9u2) ...
Selecting previously unselected package ruby2.3.
```

```

Preparing to unpack .../10-ruby2.3_2.3.3-1+deb9u2_amd64.deb ...
Unpacking ruby2.3 (2.3.3-1+deb9u2) ...
Selecting previously unselected package ruby.
Preparing to unpack .../11-ruby_1%3a2.3.3_amd64.deb ...
Unpacking ruby (1:2.3.3) ...
Selecting previously unselected package rake.
Preparing to unpack .../12-rake_10.5.0-2_all.deb ...
Unpacking rake (10.5.0-2) ...
Selecting previously unselected package ruby2.3-doc.
Preparing to unpack .../13-ruby2.3-doc_2.3.3-1+deb9u2_all.deb ...
Unpacking ruby2.3-doc (2.3.3-1+deb9u2) ...
Selecting previously unselected package ri.
Preparing to unpack .../14-ri_1%3a2.3.3_all.deb ...
Unpacking ri (1:2.3.3) ...
Selecting previously unselected package ruby2.3-dev:amd64.
Preparing to unpack .../15-ruby2.3-dev_2.3.3-1+deb9u2_amd64.deb ...
Unpacking ruby2.3-dev:amd64 (2.3.3-1+deb9u2) ...
Selecting previously unselected package ruby-dev:amd64.
Preparing to unpack .../16-ruby-dev_1%3a2.3.3_amd64.deb ...
Unpacking ruby-dev:amd64 (1:2.3.3) ...
Setting up ruby2.3-doc (2.3.3-1+deb9u2) ...
Setting up fonts-lato (2.0-1) ...
Setting up ruby-did-you-mean (1.0.0-2) ...
Processing triggers for libc-bin (2.24-11+deb9u3) ...
Setting up ruby-net-telnet (0.1.1-2) ...
Setting up rubygems-integration (1.11) ...
Processing triggers for man-db (2.7.6.1-2) ...
Setting up libgmpxx4ldbl:amd64 (2:6.1.2+dfsg-1) ...
Setting up ruby-minitest (5.9.0-1) ...
Processing triggers for fontconfig (2.11.0-6.7+b1) ...
Setting up ruby-power-assert (0.3.0-1) ...
Setting up libgmp-dev:amd64 (2:6.1.2+dfsg-1) ...
Setting up ruby-test-unit (3.1.7-2) ...
Setting up rake (10.5.0-2) ...
Setting up libruby2.3:amd64 (2.3.3-1+deb9u2) ...
Setting up ruby2.3 (2.3.3-1+deb9u2) ...
Setting up ruby2.3-dev:amd64 (2.3.3-1+deb9u2) ...
Setting up ruby-dev:amd64 (1:2.3.3) ...
Setting up ruby (1:2.3.3) ...
Setting up ri (1:2.3.3) ...
Processing triggers for libc-bin (2.24-11+deb9u3) ...
root@debian:~# apt-get install zlib1g-dev
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  zlib1g-dev
0 upgraded, 1 newly installed, 0 to remove and 20 not upgraded.
Need to get 0 B/205 kB of archives.
After this operation, 424 kB of additional disk space will be used.
Selecting previously unselected package zlib1g-dev:amd64.
(Reading database ... 237374 files and directories currently installed.)
Preparing to unpack .../zlib1g-dev_1%3a1.2.8.dfsg-5_amd64.deb ...
Unpacking zlib1g-dev:amd64 (1:1.2.8.dfsg-5) ...
Processing triggers for man-db (2.7.6.1-2) ...
Setting up zlib1g-dev:amd64 (1:1.2.8.dfsg-5) ...

```

Ezután ellenőrizzük, hogy elérhető-e a ruby értelmező és a gem Ruby csomagkezelő, és nézzük meg az előre telepített Ruby API-k listáját.

```

kovacsg@debian:~# ruby -v

```

```

ruby 2.3.3 p222 (2016-11-21) [x86_64-linux-gnu]
kovacs@debian:~# gem list

*** LOCAL GEMS ***

bigdecimal (1.2.8)
did_you_mean (1.0.0)
io-console (0.4.5)
json (1.8.3)
minitest (5.9.0)
net-telnet (0.1.1)
power_assert (0.2.7)
psych (2.1.0)
rake (10.5.0)
rdoc (4.2.1)
test-unit (3.1.7)

```

A következő lépés a Ruby dokumentációgeneráló függvénykönyvtárának telepítése, amit a `gem` Ruby csomagkezelővel teszünk meg. A lépést kihagyva a Rails keretrendszer dokumentációjának telepítése sikertelen lenne, ami mindazonáltal nem nagy tragédia lévén annak telepítését amúgy is előszere-ttel kikapcsoljuk a `gem --no-rdoc --no-ri` kapcsolóival. Először azonban hibára futottunk, mert az egyik szükséges C fejlesztői függvénykönyvtár nem volt elérhető.

```

root@debian:~# gem install rails
Fetching: concurrent-ruby-1.0.5.gem (100%)
Successfully installed concurrent-ruby-1.0.5
Fetching: i18n-1.1.0.gem (100%)
Successfully installed i18n-1.1.0
Fetching: thread_safe-0.3.6.gem (100%)
Successfully installed thread_safe-0.3.6
Fetching: tzinfo-1.2.5.gem (100%)
Successfully installed tzinfo-1.2.5
Fetching: activesupport-5.2.1.gem (100%)
Successfully installed activesupport-5.2.1
Fetching: rack-2.0.5.gem (100%)
Successfully installed rack-2.0.5
Fetching: rack-test-1.1.0.gem (100%)
Successfully installed rack-test-1.1.0
Fetching: mini_portile2-2.3.0.gem (100%)
Successfully installed mini_portile2-2.3.0
Fetching: nokogiri-1.8.4.gem (100%)
Building native extensions. This could take a while...
Successfully installed nokogiri-1.8.4
Fetching: crass-1.0.4.gem (100%)
Successfully installed crass-1.0.4
Fetching: loofah-2.2.2.gem (100%)
Successfully installed loofah-2.2.2
Fetching: rails-html-sanitizer-1.0.4.gem (100%)
Successfully installed rails-html-sanitizer-1.0.4
Fetching: rails-dom-testing-2.0.3.gem (100%)
Successfully installed rails-dom-testing-2.0.3
Fetching: builder-3.2.3.gem (100%)
Successfully installed builder-3.2.3
Fetching: erubi-1.7.1.gem (100%)
Successfully installed erubi-1.7.1
Fetching: actionview-5.2.1.gem (100%)
Successfully installed actionview-5.2.1

```

```

Fetching: actionpack-5.2.1.gem (100%)
Successfully installed actionpack-5.2.1
Fetching: activemodel-5.2.1.gem (100%)
Successfully installed activemodel-5.2.1
Fetching: arel-9.0.0.gem (100%)
Successfully installed arel-9.0.0
Fetching: activerecord-5.2.1.gem (100%)
Successfully installed activerecord-5.2.1
Fetching: globalid-0.4.1.gem (100%)
Successfully installed globalid-0.4.1
Fetching: activejob-5.2.1.gem (100%)
Successfully installed activejob-5.2.1
Fetching: mini_mime-1.0.1.gem (100%)
Successfully installed mini_mime-1.0.1
Fetching: mail-2.7.0.gem (100%)
Successfully installed mail-2.7.0
Fetching: actionmailer-5.2.1.gem (100%)
Successfully installed actionmailer-5.2.1
Fetching: nio4r-2.3.1.gem (100%)
Building native extensions. This could take a while...
Successfully installed nio4r-2.3.1
Fetching: websocket-extensions-0.1.3.gem (100%)
Successfully installed websocket-extensions-0.1.3
Fetching: websocket-driver-0.7.0.gem (100%)
Building native extensions. This could take a while...
Successfully installed websocket-driver-0.7.0
Fetching: actioncable-5.2.1.gem (100%)
Successfully installed actioncable-5.2.1
Fetching: mimemagic-0.3.2.gem (100%)
Successfully installed mimemagic-0.3.2
Fetching: marcel-0.3.3.gem (100%)
Successfully installed marcel-0.3.3
Fetching: activestorage-5.2.1.gem (100%)
Successfully installed activestorage-5.2.1
Fetching: thor-0.20.0.gem (100%)
Successfully installed thor-0.20.0
Fetching: method_source-0.9.0.gem (100%)
Successfully installed method_source-0.9.0
Fetching: raities-5.2.1.gem (100%)
Successfully installed raities-5.2.1
Fetching: bundler-1.16.5.gem (100%)
Successfully installed bundler-1.16.5
Fetching: sprockets-3.7.2.gem (100%)
Successfully installed sprockets-3.7.2
Fetching: sprockets-rails-3.2.1.gem (100%)
Successfully installed sprockets-rails-3.2.1
Fetching: rails-5.2.1.gem (100%)
Successfully installed rails-5.2.1
Parsing documentation for nokogiri-1.8.4
Installing ri documentation for nokogiri-1.8.4
Parsing documentation for crass-1.0.4
Installing ri documentation for crass-1.0.4
Parsing documentation for loofah-2.2.2
Installing ri documentation for loofah-2.2.2
Parsing documentation for rails-html-sanitizer-1.0.4
Installing ri documentation for rails-html-sanitizer-1.0.4
Parsing documentation for rails-dom-testing-2.0.3
Installing ri documentation for rails-dom-testing-2.0.3
Parsing documentation for builder-3.2.3
Installing ri documentation for builder-3.2.3
Parsing documentation for erubi-1.7.1
Installing ri documentation for erubi-1.7.1

```

```

Parsing documentation for actionview-5.2.1
Installing ri documentation for actionview-5.2.1
Parsing documentation for actionpack-5.2.1
Installing ri documentation for actionpack-5.2.1
Parsing documentation for activemodel-5.2.1
Installing ri documentation for activemodel-5.2.1
Parsing documentation for arel-9.0.0
Installing ri documentation for arel-9.0.0
Parsing documentation for activerecord-5.2.1
Installing ri documentation for activerecord-5.2.1
Parsing documentation for globalid-0.4.1
Installing ri documentation for globalid-0.4.1
Parsing documentation for activejob-5.2.1
Installing ri documentation for activejob-5.2.1
Parsing documentation for mini_mime-1.0.1
Installing ri documentation for mini_mime-1.0.1
Parsing documentation for mail-2.7.0
Installing ri documentation for mail-2.7.0
Parsing documentation for actionmailer-5.2.1
Installing ri documentation for actionmailer-5.2.1
Parsing documentation for nio4r-2.3.1
Installing ri documentation for nio4r-2.3.1
Parsing documentation for websocket-extensions-0.1.3
Installing ri documentation for websocket-extensions-0.1.3
Parsing documentation for websocket-driver-0.7.0
Installing ri documentation for websocket-driver-0.7.0
Parsing documentation for actioncable-5.2.1
Installing ri documentation for actioncable-5.2.1
Parsing documentation for mimemagic-0.3.2
Installing ri documentation for mimemagic-0.3.2
Parsing documentation for marcel-0.3.3
Installing ri documentation for marcel-0.3.3
Parsing documentation for activestorage-5.2.1
Installing ri documentation for activestorage-5.2.1
Parsing documentation for thor-0.20.0
Installing ri documentation for thor-0.20.0
Parsing documentation for method_source-0.9.0
Installing ri documentation for method_source-0.9.0
Parsing documentation for railties-5.2.1
Installing ri documentation for railties-5.2.1
Parsing documentation for bundler-1.16.5
Installing ri documentation for bundler-1.16.5
Parsing documentation for sprockets-3.7.2
Installing ri documentation for sprockets-3.7.2
Parsing documentation for sprockets-rails-3.2.1
Installing ri documentation for sprockets-rails-3.2.1
Parsing documentation for rails-5.2.1
Installing ri documentation for rails-5.2.1
Done installing documentation for nokogiri, crass, loofah, rails-html-
  sanitizer, rails-dom-testing, builder, erubi, actionview, actionpack,
  activemodel, arel, activerecord, globalid, activejob, mini_mime, mail,
  actionmailer, nio4r, websocket-extensions, websocket-driver, actioncable
  , mimemagic, marcel, activestorage, thor, method_source, railties,
  bundler, sprockets, sprockets-rails, rails after 60 seconds
31 gems installed

```

A Rails rendszerünk konfigurációja a következő lesz a félév során. Kétféle webszervert használunk, a fejlesztéshez a Puma-t, míg az éles rendszerhez az Apache2-t. Adatbáziskezelőből szintén kétfélét nézünk meg, a beágyazott SQLite3-at és a kliens-szerver alapú MySQL-t. A következőkben ezek

illesztéséhez szükséges adapterek fordítását lehetővé tevő C++ és Ruby függvénykönyvtárakat telepítjük.

Railshez egy opcionális, ám az éles rendszerek szempontjából annál hasznosabb komponens az Apache webserverral való integrációt lehetővé tevő Passenger plugin.

```
root@debian:~# gem install passenger
Fetching: passenger-5.3.4.gem (100%)
Building native extensions. This could take a while...
Successfully installed passenger-5.3.4
Parsing documentation for passenger-5.3.4
Installing ri documentation for passenger-5.3.4
Done installing documentation for passenger after 50 seconds
1 gem installed
```

Ezután telepíthetjük először az SQLite, majd a MySQL adapterének Ruby API-ját. Ez két lépésből áll, először a operációs rendszertől függő fájlok fordításához szükséges forrásokat (C és C++ headereket) telepítjük.

```
root@debian:~# apt-get install libsqlite3-dev
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
  sqlite3-doc
The following NEW packages will be installed:
  libsqlite3-dev
0 upgraded, 1 newly installed, 0 to remove and 20 not upgraded.
Need to get 0 B/704 kB of archives.
After this operation, 2,063 kB of additional disk space will be used.
Selecting previously unselected package libsqlite3-dev:amd64.
(Reading database ... 237403 files and directories currently installed.)
Preparing to unpack .../libsqlite3-dev_3.16.2-5+deb9u1_amd64.deb ...
Unpacking libsqlite3-dev:amd64 (3.16.2-5+deb9u1) ...
Setting up libsqlite3-dev:amd64 (3.16.2-5+deb9u1) ...
```

```
root@debian:~# apt-get install libmysqlclient-dev
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  libmysqlclient-dev
0 upgraded, 1 newly installed, 0 to remove and 20 not upgraded.
Need to get 946 kB of archives.
After this operation, 5,660 kB of additional disk space will be used.
Get:1 http://ftp.hu.debian.org/debian oldstable/main amd64 libmysqlclient-dev amd64 5.5.60-0+deb8u1 [946 kB]
Fetched 946 kB in 0s (2,146 kB/s)
Selecting previously unselected package libmysqlclient-dev.
(Reading database ... 237414 files and directories currently installed.)
Preparing to unpack .../libmysqlclient-dev_5.5.60-0+deb8u1_amd64.deb ...
Unpacking libmysqlclient-dev (5.5.60-0+deb8u1) ...
Setting up libmysqlclient-dev (5.5.60-0+deb8u1) ...
Processing triggers for man-db (2.7.6.1-2) ...
```

Ezután telepíthetjük magukat az adatbáziskezelő-adaptereket, a `sqlite3`-t és a `mysql2`-t. A PostgreSQL adapterét `pg`-nek hívják, a gyakorlaton azt

nem fogjuk használni.

```
root@debian:~# gem install sqlite3
Fetching: sqlite3-1.3.13.gem (100%)
Building native extensions. This could take a while...
Successfully installed sqlite3-1.3.13
Parsing documentation for sqlite3-1.3.13
Installing ri documentation for sqlite3-1.3.13
Done installing documentation for sqlite3 after 0 seconds
1 gem installed
```

```
root@debian:~# gem install mysql2
Fetching: mysql2-0.5.2.gem (100%)
Building native extensions. This could take a while...
Successfully installed mysql2-0.5.2
Parsing documentation for mysql2-0.5.2
Installing ri documentation for mysql2-0.5.2
Done installing documentation for mysql2 after 0 seconds
1 gem installed
```

Az alap Rails rendszerünk összeállt, menetközben szükségünk lesz további gemek telepítésére. Ilyen a `therubyracer`, ami egy szerver oldali JavaScript interpreter, vagy a `turbolinks` API nélkül a Rails már nem hajlandó elindulni, noha az opcionális.

```
root@debian:~# gem install execjs therubyracer
Fetching: execjs-2.7.0.gem (100%)
Successfully installed execjs-2.7.0
Parsing documentation for execjs-2.7.0
Installing ri documentation for execjs-2.7.0
Done installing documentation for execjs after 0 seconds
Fetching: ref-2.0.0.gem (100%)
Successfully installed ref-2.0.0
Fetching: libv8-3.16.14.19-x86_64-linux.gem (100%)
Successfully installed libv8-3.16.14.19-x86_64-linux
Fetching: therubyracer-0.12.3.gem (100%)
Building native extensions. This could take a while...
Successfully installed therubyracer-0.12.3
Parsing documentation for ref-2.0.0
Installing ri documentation for ref-2.0.0
Parsing documentation for libv8-3.16.14.19-x86_64-linux
Installing ri documentation for libv8-3.16.14.19-x86_64-linux
Parsing documentation for therubyracer-0.12.3
Installing ri documentation for therubyracer-0.12.3
Done installing documentation for ref, libv8, therubyracer after 0 seconds
4 gems installed
```

Az `action` és az `active` kezdetű függvénykönyvtárak adják a Rails keretrendszer magját.

2. Rails alkalmazás létrehozása

A Rails használatához a `rails` szkriptet használjuk immáron nem rendszer-gazdaként, hanem egyszerű felhasználóként. A parancs Debian/Ubuntu Linux és 2.3-as ruby esetén a `/usr/local/bin/` könyvtárba került.

A `-d` kapcsoló különös jelentőséggel bír számunkra, ezzel adhatjuk meg a használni kívánt adatbáziskezelő típusát. A gyakorlatok keretében ezek közül a `sqlite`-ot mint alapértelmezett adatbáziskezelőt és a `mysql`-t fogjuk használni. A `-B` kapcsoló mellőzi a Rails alkalmazásunk Ruby függőségeinek feloldását, amelyet azonban később pótolnunk kell. Javascript API-nak a `jquery` tökéletesen meg fog felelni számunkra, így azt nem módosítjuk.

```

kovacs@debian:~# rails
Usage:
  rails new APP_PATH [options]

Options:
  [--skip-namespace], [--no-skip-namespace] # Skip namespace
  (affects only isolated applications)
  -r, [--ruby=PATH] # Path to the
  Ruby binary of your choice # Default: /usr/
  bin/ruby
  -m, [--template=TEMPLATE] # Path to some
  application template (can be a filesystem path or URL)
  -d, [--database=DATABASE] # Preconfigure
  for selected database (options: mysql/postgresql/sqlite3/oracle/
  frontbase/ibm_db/sqlserver/jdbcmysql/jdbcsqlite3/jdbcpostgresql/jdbc)
  # Default:
  sqlite3
  [--skip-yarn], [--no-skip-yarn] # Don't use Yarn
  for managing JavaScript dependencies
  [--skip-gemfile], [--no-skip-gemfile] # Don't create a
  Gemfile
  -G, [--skip-git], [--no-skip-git] # Skip .gitignore
  file
  [--skip-keeps], [--no-skip-keeps] # Skip source
  control .keep files
  -M, [--skip-action-mailer], [--no-skip-action-mailer] # Skip Action
  Mailer files
  -O, [--skip-active-record], [--no-skip-active-record] # Skip Active
  Record files
  [--skip-active-storage], [--no-skip-active-storage] # Skip Active
  Storage files
  -P, [--skip-puma], [--no-skip-puma] # Skip Puma
  related files
  -C, [--skip-action-cable], [--no-skip-action-cable] # Skip Action
  Cable files
  -S, [--skip-sprockets], [--no-skip-sprockets] # Skip Sprockets
  files
  [--skip-spring], [--no-skip-spring] # Don't install
  Spring application preloader
  [--skip-listen], [--no-skip-listen] # Don't generate
  configuration that depends on the listen gem
  [--skip-coffee], [--no-skip-coffee] # Don't use
  CoffeeScript
  -J, [--skip-javascript], [--no-skip-javascript] # Skip JavaScript
  files
  [--skip-turbolinks], [--no-skip-turbolinks] # Skip turbolinks
  gem
  -T, [--skip-test], [--no-skip-test] # Skip test files
  [--skip-system-test], [--no-skip-system-test] # Skip system
  test files
  [--skip-bootsnap], [--no-skip-bootsnap] # Skip bootsnap

```

```

gem
  [--dev], [--no-dev] # Setup the
    application with Gemfile pointing to your Rails checkout
  [--edge], [--no-edge] # Setup the
    application with Gemfile pointing to Rails repository
  [--rc=RC] # Path to file
    containing extra configuration options for rails command
  [--no-rc], [--no-no-rc] # Skip loading of
    extra configuration options from .railsrc file
  [--api], [--no-api] # Preconfigure
    smaller stack for API only apps
-B, [--skip-bundle], [--no-skip-bundle] # Don't run
  bundle install
  [--webpack=WEBPACK] # Preconfigure
    for app-like JavaScript with Webpack (options: react/vue/angular/
    elm/stimulus)

Runtime options:
  -f, [--force] # Overwrite files that already exist
  -p, [--pretend], [--no-pretend] # Run but do not make any changes
  -q, [--quiet], [--no-quiet] # Suppress status output
  -s, [--skip], [--no-skip] # Skip files that already exist

Rails options:
  -h, [--help], [--no-help] # Show this help message and quit
  -v, [--version], [--no-version] # Show Rails version number and quit

Description:
  The 'rails new' command creates a new Rails application with a default
  directory structure and configuration at the path you specify.

  You can specify extra command-line arguments to be used every time
  'rails new' runs in the .railsrc configuration file in your home
  directory.

  Note that the arguments specified in the .railsrc file don't affect the
  defaults values shown above in this help message.

Example:
  rails new ~/Code/Ruby/weblog

  This generates a skeletal Rails installation in ~/Code/Ruby/weblog.

```

A Rails keretrendszerben a rails parancs az az univerzális eszköz mellyel többek között új komponenseket hozhatunk létre és elindíthatjuk a beépített webszerveret. Egy új Rails alkalmazást a rails parancsnak new opciót megadva hozhatunk létre, a második argumentum az alkalmazás neve. A parancsot tetszőleges felhasználóként kiadva a konzolon láthatjuk az automatikusan generált fájlokat. Az egyes könyvtárak értelmezéséről előadáson esett szó, e gyakorlat keretében megnézzük az egyes fájlok szerepét és tartalmát.

A parancs végén automatikusan lefutna a Rails keretrendszer egy másik parancsa, a bundle, amely az install opció hatására összeszedi a gemek közül azokat, amelyekre az alkalmazásunknak szüksége lesz, azonban ezt a -B kapcsolóval letiltjuk, mert testre akarjuk szabni a keretrendszerünket.

```
kovacs@debian:~ # rails new gyakorlat -B
```

```

create
create README.md
create Rakefile
create .ruby-version
create config.ru
create .gitignore
create Gemfile
  run git init from "."
Initialized empty Git repository in /home/kovacs/gyakorlat/.git/
create package.json
create app
create app/assets/config/manifest.js
create app/assets/javascripts/application.js
create app/assets/javascripts/cable.js
create app/assets/stylesheets/application.css
create app/channels/application_cable/channel.rb
create app/channels/application_cable/connection.rb
create app/controllers/application_controller.rb
create app/helpers/application_helper.rb
create app/jobs/application_job.rb
create app/mailers/application_mailer.rb
create app/models/application_record.rb
create app/views/layouts/application.html.erb
create app/views/layouts/mailer.html.erb
create app/views/layouts/mailer.text.erb
create app/assets/images/.keep
create app/assets/javascripts/channels
create app/assets/javascripts/channels/.keep
create app/controllers/concerns/.keep
create app/models/concerns/.keep
create bin
create bin/bundle
create bin/rails
create bin/rake
create bin/setup
create bin/update
create bin/yarn
create config
create config/routes.rb
create config/application.rb
create config/environment.rb
create config/cable.yml
create config/puma.rb
create config/spring.rb
create config/storage.yml
create config/environments
create config/environments/development.rb
create config/environments/production.rb
create config/environments/test.rb
create config/initializers
create config/initializers/application_controller_renderer.rb
create config/initializers/assets.rb
create config/initializers/backtrace_silencers.rb
create config/initializers/content_security_policy.rb
create config/initializers/cookies_serializer.rb
create config/initializers/cors.rb
create config/initializers/filter_parameter_logging.rb
create config/initializers/inflections.rb
create config/initializers/mime_types.rb
create config/initializers/new_framework_defaults_5_2.rb
create config/initializers/wrap_parameters.rb
create config/locales

```

```

create  config/locales/en.yml
create  config/master.key
append .gitignore
create  config/boot.rb
create  config/database.yml
create  db
create  db/seeds.rb
create  lib
create  lib/tasks
create  lib/tasks/.keep
create  lib/assets
create  lib/assets/.keep
create  log
create  log/.keep
create  public
create  public/404.html
create  public/422.html
create  public/500.html
create  public/apple-touch-icon-precomposed.png
create  public/apple-touch-icon.png
create  public/favicon.ico
create  public/robots.txt
create  tmp
create  tmp/.keep
create  tmp/cache
create  tmp/cache/assets
create  vendor
create  vendor/.keep
create  test/fixtures
create  test/fixtures/.keep
create  test/fixtures/files
create  test/fixtures/files/.keep
create  test/controllers
create  test/controllers/.keep
create  test/mailers
create  test/mailers/.keep
create  test/models
create  test/models/.keep
create  test/helpers
create  test/helpers/.keep
create  test/integration
create  test/integration/.keep
create  test/test_helper.rb
create  test/system
create  test/system/.keep
create  test/application_system_test_case.rb
create  storage
create  storage/.keep
create  tmp/storage
create  tmp/storage/.keep
remove  config/initializers/cors.rb
remove  config/initializers/new_framework_defaults_5_2.rb

```

Az `app` könyvtár fogja tartalmazni az általunk készített Ruby és beágyazott Ruby kódot tartalmazó HTML forrásokat, amelyeket a MVC minta alapján struktúrál a Rails, amelyeket a `test` könyvtárban elhelyezett teszt osztályokkal ellenőrzünk. A `bin` könyvtár elérhetővé teszi számunkra a `rails`, a `rake` és a `bundle` parancsokat, amelyeket a Rails alkalmazásunk menedzsmentjét fogjuk megvalósítani. A `config` könyvtár a Rails alkalmazásunk

konfigurációs beállításait tartalmazza. A db könyvtár az aktuális adatbázis sémát, az összes eddig adatbázis séma migrációt és sqlite adatbáziskezelő esetén szerializált formában magát az adatbázis tartalmazza. A doc a dokumentációk gyűjtőhelye, a `lib` és `vendor` könyvtárak mások által készített Ruby, illetve Rails függvénykönyvtárakat tartalmazhatnak. A `public` könyvtár a beépített webservert területe, az összes ott bekövetkezett esemény a `log` könyvtárban található az aktuális Rails környezetnek megfelelő állományban kerülnek naplózásra, a webservert `tmp` könyvtárban helyezheti el az átmeneti fájljait, mint például `session` azonosítókat, sütiket.

A `Gemfile` és `Rakefile` a Rails alkalmazásunk által használt Ruby függvénykönyvtárakat specifikálja, amelyek vagy a telepített Ruby és Rails környezeten vagy a már említett `lib` és `vendor` könyvtárakban érhetők el.

Az első dolgunk a `Gemfile` módosítása. Először kikommentezzük az első sort, hogy a Rails a Ruby függvénykönyvtárakat a helyi, frissen telepített helyen keresse, és ne a megadott távoli szerveren. Ez azért szükséges, hogy a rendszer egy rögzített verziójú keretrendszer mellett tudhassuk fejleszteni, és az ne frissüljön a `Gemfile` minden módosításakor. Egy esetleges frissítés fejlesztés közben elronthatja a teljes addigi munkánkat! Praktikusan a `rails`, az adatbáziskezelő adaptere és a `jquery-rails`-en kívül a többi, a fájlban szereplő függőségre nincs szükségünk. A webservert futtatásához szükségünk lesz szerver oldali JavaScript értelmezőre, ezért a fájlba felvesszük az alábbi gem függőséget, amit az elé írt kommentjel eltávolításával tehetünk meg. A `Gemfile` minden további módosítása után futtatnunk kell majd a `bundle install` vagy `bundle update` parancsot.

```
gem 'therubyracer', platforms: :ruby
gem 'execjs'
```

A konzolon kiadott `bundle install` paranccsal telepíthetjük az összes az esetlegesen hiányzó vagy nem megfelelő verziójú Ruby függvénykönyvtárat, a `bundle update` paranccsal pedig az éppen használni kívánt verziót tudunk betölteni az egyes gemekből. A `bundle` parancshoz a `--path` kapcsolót és egy fájlrendszeri útvonalat hozzáfűzve elérhetjük, hogy garantáltan mindig a saját Ruby környezetünkkel dolgozhassunk, és ne ütközzünk az időközben bekövetkező rendszerfrissítések okozta inkompatibilitás áldozatául. Ha a Ruby csomagokat minden felhasználó számára elérhetővé akarjuk tenni, meg kell adnunk az adminisztrátor jelszavát.⁴

```
kovacs@debian:~/gyakorlat# bundle install
```

⁴A Rails telepítése után nem minden csomag áll rendelkezésre a `bundle` számára, az további csomagok telepítésére kérni fogja a rendszergazdai jelszót. A gyakorlaton csalást követtünk el, az összes további csomag már előtelepített állapotban volt.

The dependency tzinfo-data (≥ 0) will be unused by any of the platforms Bundler is installing for. Bundler is installing for ruby but the dependency is only for x86-mingw32, x86-mswin32, x64-mingw32, java. To add those platforms to the bundle, run 'bundle lock --add-platform x86-mingw32 x86-mswin32 x64-mingw32 java'.

```
Resolving dependencies...
Using rake 12.3.1
Using concurrent-ruby 1.0.5
Using i18n 1.1.0
Using minitest 5.11.3
Using thread_safe 0.3.6
Using tzinfo 1.2.5
Using activesupport 5.2.1
Using builder 3.2.3
Using erubi 1.7.1
Using mini_portile2 2.3.0
Using nokogiri 1.8.4
Using rails-dom-testing 2.0.3
Using crass 1.0.4
Using loofah 2.2.2
Using rails-html-sanitizer 1.0.4
Using actionview 5.2.1
Using rack 2.0.5
Using rack-test 1.1.0
Using actionpack 5.2.1
Using nio4r 2.3.1
Using websocket-extensions 0.1.3
Using websocket-driver 0.7.0
Using actioncable 5.2.1
Using globalid 0.4.1
Using activejob 5.2.1
Using mini_mime 1.0.1
Using mail 2.7.0
Using actionmailer 5.2.1
Using activemodel 5.2.1
Using arel 9.0.0
Using activerecord 5.2.1
Using mimemagic 0.3.2
Using marcel 0.3.3
Using activestorage 5.2.1
Using public_suffix 3.0.3
Using addressable 2.5.2
Using io-like 0.3.0
Using archive-zip 0.11.0
Using bindix 0.5.0
Using msgpack 1.2.4
Using bootsnap 1.3.2
Using bundler 1.16.5
Using byebug 10.0.2
Using xpath 3.1.0
Using capybara 3.8.1
Using ffi 1.9.25
Using childprocess 0.9.0
Using chromedriver-helper 2.1.0
Using coffee-script-source 1.12.2
Using execjs 2.7.0
Using coffee-script 2.4.1
Using method_source 0.9.0
Using thor 0.20.0
Using raities 5.2.1
Using coffee-rails 4.2.2
Using multi_json 1.13.1
```

```

Using jbuilder 2.7.0
Using libv8 3.16.14.19 (x86_64-linux)
Using rb-fsevent 0.10.3
Using rb-inotify 0.9.10
Using ruby_dep 1.5.0
Using listen 3.1.5
Using puma 3.12.0
Using sprockets 3.7.2
Using sprockets-rails 3.2.1
Using rails 5.2.1
Using ref 2.0.0
Using rubyzip 1.2.2
Using sass-listen 4.0.0
Using sass 3.6.0
Using tilt 2.0.8
Using sass-rails 5.0.7
Using selenium-webdriver 3.14.0
Using spring 2.0.2
Using spring-watcher-listen 2.0.1
Using sqlite3 1.3.13
Using therubyracer 0.12.3
Using turbolinks-source 5.2.0
Using turbolinks 5.2.0
Using uglifier 4.1.19
Using web-console 3.7.0
Bundle complete! 20 Gemfile dependencies, 81 gems now installed.
Use 'bundle info [gemname]' to see where a bundled gem is installed.

```

A Rails rendszerünk használatához minden függőséget feloldottunk, ezután az alábbi Ruby függvénykönyvtárak kerültek telepítésre:

```

kovacs@debian:~# gem list

*** LOCAL GEMS ***

actioncable (5.2.1)
actionmailer (5.2.1)
actionpack (5.2.1)
actionview (5.2.1)
activejob (5.2.1)
activemodel (5.2.1)
activerecord (5.2.1)
activestorage (5.2.1)
activesupport (5.2.1)
addressable (2.5.2)
archive-zip (0.11.0)
arel (9.0.0)
bigdecimal (1.2.8)
bindex (0.5.0)
bootsnap (1.3.2)
builder (3.2.3)
bundler (1.16.5)
byebug (10.0.2)
capybara (3.8.1)
childprocess (0.9.0)
chromedriver-helper (2.1.0)
coffee-rails (4.2.2)
coffee-script (2.4.1)
coffee-script-source (1.12.2)
concurrent-ruby (1.0.5)
crass (1.0.4)

```

```
did_you_mean (1.0.0)
erubi (1.7.1)
execjs (2.7.0)
ffi (1.9.25)
globalid (0.4.1)
i18n (1.1.0)
io-console (0.4.5)
io-like (0.3.0)
jbuilder (2.7.0)
json (1.8.3)
libv8 (3.16.14.19 x86_64-linux)
listen (3.1.5)
loofah (2.2.2)
mail (2.7.0)
marcel (0.3.3)
method_source (0.9.0)
mimemagic (0.3.2)
mini_mime (1.0.1)
mini_portile2 (2.3.0)
minitest (5.11.3, 5.9.0)
msgpack (1.2.4)
multi_json (1.13.1)
mysql2 (0.5.2)
net-telnet (0.1.1)
nio4r (2.3.1)
nokogiri (1.8.4)
passenger (5.3.4)
power_assert (0.2.7)
psych (2.1.0)
public_suffix (3.0.3)
puma (3.12.0)
rack (2.0.5)
rack-test (1.1.0)
rails (5.2.1)
rails-dom-testing (2.0.3)
rails-html-sanitizer (1.0.4)
railties (5.2.1)
rake (12.3.1, 10.5.0)
rb-fsevent (0.10.3)
rb-inotify (0.9.10)
rdoc (4.2.1)
ref (2.0.0)
ruby_dep (1.5.0)
rubyzip (1.2.2)
sass (3.6.0)
sass-listen (4.0.0)
sass-rails (5.0.7)
selenium-webdriver (3.14.0)
spring (2.0.2)
spring-watcher-listen (2.0.1)
sprockets (3.7.2)
sprockets-rails (3.2.1)
sqlite3 (1.3.13)
test-unit (3.1.7)
therubyracer (0.12.3)
thor (0.20.0)
thread_safe (0.3.6)
tilt (2.0.8)
turbolinks (5.2.0)
turbolinks-source (5.2.0)
tzinfo (1.2.5)
uglifier (4.1.19)
```



```
web-console (3.7.0)
websocket-driver (0.7.0)
websocket-extensions (0.1.3)
xpath (3.1.0)
```

A `rails` parancsot az újonnan létrehozott Rails alkalmazásunk bármely alkönyvtárából kiadva más üzenetet látunk. Az új alkalmazást létrehozó `new` mellett jelen van a kódgenerálásra alkalmas `generate` opció, a beágyazott webservert indító `server` opció, a Ruby, illetve az adatbáziskonzolt indító `console`, illetve `dbconsole` opciók.

```
kovacs@debian:~/gyakorlat# rails
The most common rails commands are:
generate      Generate new code (short-cut alias: "g")
console       Start the Rails console (short-cut alias: "c")
server        Start the Rails server (short-cut alias: "s")
test          Run tests except system tests (short-cut alias: "t")
test:system   Run system tests
dbconsole     Start a console for the database specified in config/database.
              yml
              (short-cut alias: "db")

new           Create a new Rails application. "rails_new_my_app" creates a
              new application called MyApp in "./my_app"
```

All commands can be run with `-h` (or `--help`) for more information. In addition to those commands, there are:

Rails:

```
console
credentials:edit
credentials:show
dbconsole
destroy
encrypted:edit
encrypted:show
generate
new
runner
secrets:edit
secrets:setup
secrets:show
server
test
version
```

Rake:

```
about
active_storage:install
app:template
app:update
assets:clean [keep]
assets:clobber
assets:environment
assets:precompile
cache_digests:dependencies
cache_digests:nested_dependencies
db:create
db:drop
```

```
db:environment:set
db:fixtures:load
db:migrate
db:migrate:status
db:rollback
db:schema:cache:clear
db:schema:cache:dump
db:schema:dump
db:schema:load
db:seed
db:setup
db:structure:dump
db:structure:load
db:version
dev:cache
initializers
log:clear
middleware
notes
notes:custom
restart
routes
secret
stats
test
test:db
test:system
time:zones[country_or_offset]
tmp:clear
tmp:create
yarn:install
```

A rails parancs gyakran használt argumentuma a `console` vagy röviden `c`, amivel az első gyakorlatról megismert interaktív Ruby értelmezőt indíthatunk, melyben a Rails alkalmazásunk környezeti beállításai inicializálásra kerültek.

```
kovacs@debian:~/gyakorlat/bin# rails c
Loading development environment (Rails 5.2.1)
irb(main):001:0 >
```

3. Web- és adatbázisszerver kapcsolat beállítása

A Rails keretrendszer webserverek számára állít elő dinamikus tartalmat adatbázisbeli adatok alapján. A gyakorlat során kétféle webservert és adatbáziskezelőt vizsgáltunk meg.

Az alapértelmezett webservert, a Puma-t a `rails server` vagy röviden `s` opciójával indíthatjuk el a Rails alkalmazásunk tetszőleges könyvtárából. Ha a szerveret háttérszolgáltatásként kívánjuk futtatni, akkor a `-d` kapcsolót is hozzá kell fűznünk a parancssorhoz. Ezután a szerver elérhető a `http://localhost:3000` webcímen. Ha megnyitjuk az oldalt egy böngészőben, akkor annak eseményei megjelennek a konzolon, és bekerülnek a naplófájlba

is. Itt azt látjuk, hogy HTTP GET művelet fért hozzá a "/" erőforráshoz, és, hogy a Rails ennek hatására milyen műveleteket végzett el. A kimeneten láthatjuk, hogy a webszerver a fejlesztői környezet konfigurációját használja.

```
kovacs@debian:~/gyakorlat# rails s
=> Booting Puma
=> Rails 5.2.1 application starting in development
=> Run 'rails server -h' for more startup options
Puma starting in single mode...
* Version 3.12.0 (ruby 2.3.3-p222), codename: Llamas in Pajamas
* Min threads: 5, max threads: 5
* Environment: development
* Listening on tcp://0.0.0.0:3000
```

A Rails a már említett Passenger plugin segítségével illeszthető Apache2 webszerverhez. Ehhez az alábbi kódrészletben mutatott parancsot kell kiadnunk. A parancs lefordítja, majd telepíti az Apache webszerver Rails (és egyéb további) modulját, a konzolra kiírja az Apache2 konfigurációs állományába beírandó Rails specifikus három sort, és a Rails alkalmazásunkhoz való hozzáféréshez szükséges VirtualHost beállításokat. A Passenger alapértelmezés szerint a Rails éles környezetével működik együtt, ha ezt módosítani szeretnénk, akkor a `RailsEnv development` sort kell elhelyeznünk az Apache konfigurációs állományába a fejlesztői környezet eléréséhez. Az alábbi naplórészlet a Passenger plugin telepítését mutatja be, a C források fordítása nélkül.

```
root@debian:~# passenger-install-apache2-module
Welcome to the Phusion Passenger Apache 2 module installer, v5.3.4.

This installer will guide you through the entire installation process. It
shouldn't take more than 3 minutes in total.

Here's what you can expect from the installation process:

  1. The Apache 2 module will be installed for you.
  2. You'll learn how to configure Apache.
  3. You'll learn how to deploy a Ruby on Rails application.

Don't worry if anything goes wrong. This installer will advise you on how to
solve any problems.

Press Enter to continue, or Ctrl-C to abort.

-----

Which languages are you interested in?

Use <space> to select.
If the menu doesn't display correctly, press '!'.

  x  Ruby
  >  Python
  _  Node.js
  _  Meteor
```

Checking for required software...

- * Checking for C compiler...
Found: yes
Location: /usr/bin/cc
- * Checking for C++ compiler...
Found: yes
Location: /usr/bin/c++
- * Checking for Curl development headers with SSL support...
Found: yes
curl-config location: /usr/bin/curl-config
Header location: /usr/include/x86_64-linux-gnu/curl/curl.h
Version: libcurl 7.52.1
Usable: yes
Supports SSL: yes
- * Checking for Zlib development headers...
Found: yes
Location: /usr/include/zlib.h
- * Checking for Apache 2...
Found: yes
Location of httpd: /usr/sbin/apache2
Apache version: 2.4.25
- * Checking for Rake (associated with /usr/bin/ruby2.3)...
Found: yes
Location: /usr/bin/ruby2.3 /usr/bin/rake
- * Checking for OpenSSL support for Ruby...
Found: yes
- * Checking for RubyGems...
Found: yes
- * Checking for Ruby development headers...
Found: yes
Location: /usr/include/ruby-2.3.0/ruby.h
- * Checking for rack...
Found: yes
- * Checking for OpenSSL development headers...
Found: yes
Location: /usr/include/openssl/ssl.h
- * Checking for Apache 2 development headers...
Found: yes
Location of apxs2: /usr/bin/apxs2
- * Checking for Apache Portable Runtime (APR) development headers...
Found: yes
Location: /usr/bin/apr-1-config
Version: 1.5.2
- * Checking for Apache Portable Runtime Utility (APU) development headers...
Found: yes
Location: /usr/bin/apu-1-config
Version: 1.5.4

Checking whether there are multiple Apache installations...
Only a single installation detected. This is good.

Compiling and installing Apache 2 module...

Almost there!

Please edit your Apache configuration file, **and** add these lines:

```
LoadModule passenger_module /var/lib/gems/2.3.0/gems/passenger-5.3.4/
  buildout/apache2/mod_passenger.so
<IfModule mod_passenger.c>
  PassengerRoot /var/lib/gems/2.3.0/gems/passenger-5.3.4
  PassengerDefaultRuby /usr/bin/ruby2.3
</IfModule>
```

After you restart Apache, you are ready to deploy any number of web applications on Apache, with a minimum amount of configuration!

Press ENTER **when** you are done editing.

Validating installation...

```
* Checking whether this Passenger install is in PATH... x
* Checking whether there are no other Passenger installations... x
* Checking whether Apache is installed... x
* Checking whether the Passenger module is correctly configured in
  ApacheDocumentRoot [/home/kovacs/gyakorlat/public] does not exist
x
```

Everything looks good. :-)

A Rails kényelmesebb hozzáférése végett módosíthatjuk a helyi gépen doménnévtáblát (Windowson C:\windows\system32\drivers\etc\hosts, Linuxon /etc/hosts) a következő bejegyzés hozzáadva, amely minden, a `gyakorlat.com`, illetve `www.gyakorlat.com` címre küldött kérést a helyi gépen kezeltet le.

```
127.0.0.1      gyakorlat.com www.gyakorlat.com
```

Az Apache2 modul elérhetővé tételére az Apache2 konfigurációs könyvtárban létre kell hoznunk a modul beállításait. Szerencsére a telepítő parancs a helyes beállításokat kiírja a konzolra. A fájlt `/etc/apache2/mods-available/rails.load` néven hozzuk létre, a tartalma az alábbi konfigurációrészlethez hasonló. A modult a `a2enmod rails` paranccsal, majd a webszerver újraindításával tehetjük aktívvá.

```
LoadModule passenger_module /var/lib/gems/2.3.0/gems/passenger-5.3.4/
  buildout/apache2/mod_passenger.so
<IfModule mod_passenger.c>
  PassengerRoot /var/lib/gems/2.3.0/gems/passenger-5.3.4
  PassengerDefaultRuby /usr/bin/ruby2.3
</IfModule>
```

Az Apache2 virtuális hoszt beállításait szintén konfigurációs állományok között kell elhelyeznünk például `/etc/apache2/sites-available/gyakorlat.conf` néven, majd az `a2ensite gyakorlat.conf` paranccsal kon-

zolon engedélyezzük a gyakorlat oldalt⁵. Ennek tartalma a telepítő parancs alapján a következő lehet:

```
RackEnv development

#<VirtualHost *:80>
<VirtualHost www.gyakorlat.com:80>

    ServerName www.gyakorlat.com
    ServerAdmin admin@gyakorlat.com
    DocumentRoot /home/kovacs/gyakorlat/public
    ServerSignature On

    CustomLog /var/log/apache2/gyakorlat_access.log combined
    ErrorLog /var/log/apache2/gyakorlat_error.log
    LogLevel info

    <Directory />
        Options FollowSymLinks
        AllowOverride None
    </Directory>
    <Directory /home/kovacs/gyakorlat/public/>
        Require all granted
        Options Indexes FollowSymLinks
        AllowOverride None
        Order allow,deny
        Allow from all
        Options -MultiViews
    </Directory>

</VirtualHost>
```

Mivel módosítottuk az Apache2 beállításait, újra kell töltenünk annak konfigurációs állományait.

```
root@debian:~# /etc/init.d/apache2 reload
[ ok ] Reloading web server config: apache2.
```

Ezután egy böngészőbe beírva a `www.gyakorlat.com` címet a Rails alkalmazásunk fog megjelenni! A Rails alkalmazás módosítása után az Apache-beli frissítése a `tmp/restart.txt` fájl időpecsétjének módosításával lehetséges (mentés, létrehozás, `touch` stb.)

Mivel a Rails alkalmazás létrehozásakor nem adtunk meg, hogy melyik adatbáziskezelőt kívánjuk használni, ezért az alapértelmezett SQLite konfigurációs paraméterei jelennek meg a `database.yml`-ben mindhárom definiált környezethez, vagyis a fejlesztési (`development`), tesztelési (`test`) és éles (`production`). Az egyes környezetek adapter opciója határozza meg az adatbáziskezelő típusát és a használható paraméterkészletet. Az SQLite az adatait fájlokban tárolja, amelyek a Rails alkalmazás db könyvtárában találhatóak, és a környezet nevével azonosítottak e fájl szerint.

⁵Alternatív megoldás: `ln -s /etc/apache2/sites-available/gyakorlat.conf /etc/apache2/sites-enabled`

```
# SQLite version 3.x
# gem install sqlite3
#
# Ensure the SQLite 3 gem is defined in your Gemfile
# gem 'sqlite3'
#
default: &default
  adapter: sqlite3
  pool: 5
  timeout: 5000

development:
  <<: *default
  database: db/development.sqlite3

# Warning: The database defined as "test" will be erased and
# re-generated from your development database when you run "rake".
# Do not set this db to the same as development or production.
test:
  <<: *default
  database: db/test.sqlite3

production:
  <<: *default
  database: db/production.sqlite3
```

```
kovacs@debian:~/gyakorlat/bin# rails db
SQLite version 3.16.2 2017-01-06 16:32:41
Enter ".help" for usage hints.
sqlite>
```

MySQL esetén (`rails new gyakorlat -d mysql`) a `mysql2` adaptert használjuk. A különbség a `Gemfile`-ban és a `database.yml`-ben jelentkezik. Fontos opció az alapértelmezett karakterkódolás megadása, ami, ha lehet hagyjunk változatlanul `utf8` értéken. A `database` opció a séma nevét tartalmazza értéként. A felhasználónév és jelszó megadása kötelező. A kapcsolat lehet `socket` vagy `host` és `port` alapon megadott.

Távoli gépen található MySQL esetén szükséges a Rails felhasználó hozzáférési jogosultságainak beállítása, amelyet a MySQL `mysql` adatbázisának `user` táblájában kell megtennünk. A Rails szerverének IP címére engedélyeznünk kell a Rails adatbázis-konfigurációs állományában megadott felhasználónévvel és jelszóval azonosított felhasználó számára a `select`, `update`, `insert`, `delete`, `create`, `alter`, `drop` és `index` jogosultságokat.

Konzolos adatbáziskapcsolatot a `rails db` paranccsal tudunk előhozni.

```
kovacs@debian:~/gyakorlat/app/models# rails db
>Welcome to the MariaDB monitor.  Commands end with ; or \g.
>Your MariaDB connection id is 15
>Server version: 10.1.26-MariaDB-0+deb9u1 Debian 9.1

Copyright (c) 2000, 2017, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input
statement.
```

```

MariaDB [gyakorlat_development]> Bye
kovacsg@debian:~/gyakorlat# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 3
Server version: 10.1.26-MariaDB-0+deb9u1 Debian 9.1

Copyright (c) 2000, 2017, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input
statement.

MariaDB [(none)]> use gyakorlat_development;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MariaDB [gyakorlat_development]> Bye

```

Az adatbáziskapcsolatnak a szerver elindítása előtt léteznie kell. Ez SQLite esetén nem probléma, hiszen a fejlesztői adatbázisfájl automatikusan létrejön, MySQL esetén viszont a sémákat explicite létre kell hoznunk. Ebben a Rails korábbi verziói esetén a rake parancs volt segítségünkre, amelyet a Rails keretrendszerhez kapcsolódó olyan műveletekre használtunk, mint az adatbázis menedzsmentje vagy a tesztvégrehajtás vezérlése. A Rails 5-től ezek a műveletek a rails paranccsal is elvégezhetők, a rake műveletek mindazonáltal továbbra is használhatók. A műveletek listája alább látható.

```

rake about # List versions of all Rails
            frameworks and the environment
rake app:template # Applies the template supplied by
                  LOCATION=(/path/to/template) or URL
rake app:update # Update configs and some other
                initially generated files (or use just update:configs or update:bin)
rake assets:clean[keep] # Remove old compiled assets
rake assets:clobber # Remove compiled assets
rake assets:environment # Load asset compile environment
rake assets:precompile # Compile all the assets named in
                       config.assets.precompile
rake cache_digests:dependencies # Lookup first-level dependencies
                                for TEMPLATE (like messages/show or comments/_comment.html)
rake cache_digests:nested_dependencies # Lookup nested dependencies for
                                        TEMPLATE (like messages/show or comments/_comment.html)
rake db:create # Creates the database from
               DATABASE_URL or config/database.yml for the current RAILS_ENV (use db:
               create:all to create all databases in the config)
rake db:drop # Drops the database from
             DATABASE_URL or config/database.yml for the current RAILS_ENV (use db:
             drop:all to drop all databases in the config)
rake db:environment:set # Set the environment value for the
                        database
rake db:fixtures:load # Loads fixtures into the current
                     environment's database
rake db:migrate # Migrate the database (options:
                VERSION=x, VERBOSE=false, SCOPE=blog)
rake db:migrate:status # Display status of migrations
rake db:rollback # Rolls the schema back to the

```



```

previous version (specify steps w/ STEP=n)
rake db:schema:cache:clear      # Clears a db/schema_cache.yml file
rake db:schema:cache:dump      # Creates a db/schema_cache.yml file
rake db:schema:dump             # Creates a db/schema.rb file that
                               # is portable against any DB supported by Active Record
rake db:schema:load             # Loads a schema.rb file into the
                               # database
rake db:seed                    # Loads the seed data from db/seeds.
                               # rb
rake db:setup                   # Creates the database, loads the
                               # schema, and initializes with the seed data (use db:reset to also drop
                               # the database first)
rake db:structure:dump          # Dumps the database structure to db
                               # /structure.sql
rake db:structure:load          # Recreates the databases from the
                               # structure.sql file
rake db:version                 # Retrieves the current schema
                               # version number
rake dev:cache                  # Toggle development mode caching on
                               # /off
rake initializers               # Print out all defined initializers
                               # in the order they are invoked by Rails
rake log:clear                  # Truncates all/specified *.log
                               # files in log/ to zero bytes (specify which logs with LOGS=test,
                               # development)
rake middleware                  # Prints out your Rack middleware
                               # stack
rake notes                       # Enumerate all annotations (use
                               # notes:optimize, :fixme, :todo for focus)
rake notes:custom               # Enumerate a custom annotation,
                               # specify with ANNOTATION=CUSTOM
rake restart                     # Restart app by touching tmp/
                               # restart.txt
rake routes                     # Print out all defined routes in
                               # match order, with names
rake secret                      # Generate a cryptographically
                               # secure secret key (this is typically used to generate a secret for
                               # cookie sessions)
rake stats                       # Report code statistics (KLOCs, etc
                               # ) from the application or engine
rake test                        # Runs all tests in test folder
                               # except system ones
rake test:db                     # Run tests quickly, but also reset
                               # db
rake test:system                 # Run system tests only
rake time:zones[country_or_offset] # List all time zones, list by two-
                               # letter country code ('rails time:zones[US]'), or list by UTC offset ('
                               # rails time:zones[-8]')
rake tmp:clear                   # Clear cache and socket files from
                               # tmp/ (narrow w/ tmp:cache:clear, tmp:sockets:clear)
rake tmp:create                  # Creates tmp directories for cache,
                               # sockets, and pids
rake yarn:install                # Install all JavaScript
                               # dependencies as specified via Yarn

```

Töröljük a korábbi Rails alkalmazásunkat, amely az SQLite3-at használta, és hozzuk létre ugyanazt MySQL adatbáziskezelővel a `-d mysql` opció felhasználásával.

A MySQL adatbázis nem jön automatikusan létre az SQLite3-mal el-

lentétben. A létrehozása következő parancsokkal történhet meg. Először eltávolítjuk az összes azonos nevű adatbázist a `db:drop` opcióval, majd a `db:create` opciót használjuk, amely a development és a test környezetekhez hozza létre egy üres sémát. Az eredmény a Rails adatbázis konzolán ellenőrizzük.

```
kovacsg@debian:~/gyakorlat# rails db:drop
Dropped database 'gyakorlat_development'
Dropped database 'gyakorlat_test'
kovacs@debian:~/gyakorlat# rails db
ERROR 1049 (42000): Unknown database 'gyakorlat_development'
kovacs@debian:~/gyakorlat/app/views/say> rails db:create
Created database 'gyakorlat_development'
Created database 'gyakorlat_test'
kovacs@debian:~/gyakorlat# rails db
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 8
Server version: 10.1.23-MariaDB-9+deb9u1 Debian 9.0

Copyright (c) 2000, 2017, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input
statement.

MariaDB [gyakorlat_development]> show tables;
Empty set (0.00 sec)

MariaDB [gyakorlat_development]> Bye
```

4. Bevezetés a Rails használatába

Az adatbáziskapcsolat-leíró mellett a másik fontos konfigurációs fájlunk a `routes.rb`. Ez azt adja meg, hogy milyen struktúrájú legyen az URL, amivel elérjük a Rails alkalmazásunk egyes funkcióit. A Rails kontrollerek létrehozásakor automatikusan írja ezt a fájl, de megadhatunk egy általános mintát is a HTTP kérések URI-ainak útvonal részére, az alábbi kódrészlet a legáltalánosabb beállítást tartalmazza. A webservert IP címe után a Controller osztály neve (`:controller`), majd a Controller osztály egy metódusa (`:action`), majd egy adatbázis azonosító (`:id`), és végül formázási útmutató következik, például `.html` vagy `.xml`. Az utolsó három megadása opcionális.

```
Gyakorlat::Application.routes.draw do
  match ':controller(/:action(/:id(.:format)))'
end
```

Nézzük meg, hogy miként tudunk dinamikus tartalmat létrehozni Rails-szel. A példák a [1] könyvből valók.

Hozzunk létre egy új kontrollert a `rails` parancs `generate`, röviden `g` opciójával. A második argumentum (`controller`) azt mondja meg, hogy egy új kontrollert hozunk létre, a harmadik a controller nevét. A negyedik

és minden további paraméter a controllerben definiál akciókat. E parancs négy Ruby forrásfájlt és egy könyvárat hoz létre az akcióknak megfelelő web-oldalak, view-k számára. A controller nevének megfelelő controller osztályt (`say_controller.rb`), helper osztályt, illetve ezek funkcionális és egységtesztjéhez használható osztályokat.

```
kovacs@debian:~/gyakorlat# rails generate
Usage: rails generate GENERATOR [args] [options]

General options:
  -h, [--help]           # Print generator's options and usage
  -p, [--pretend]       # Run but do not make any changes
  -f, [--force]         # Overwrite files that already exist
  -s, [--skip]          # Skip files that already exist
  -q, [--quiet]         # Suppress status output

Please choose a generator below.

Rails:
  application_record
  assets
  channel
  controller
  generator
  helper
  integration_test
  jbuilder
  job
  mailer
  migration
  model
  resource
  scaffold
  scaffold_controller
  system_test
  task

ActiveRecord:
  active_record:application_record

Coffee:
  coffee:assets

Js:
  js:assets

TestUnit:
  test_unit:generator
  test_unit:plugin
```

```
kovacs@debian:~/gyakorlat# rails g controller say hello
create  app/controllers/say_controller.rb
route   get 'say/hello'
invoke  erb
create  app/views/say
create  app/views/say/hello.html.erb
invoke  test_unit
create  test/controllers/say_controller_test.rb
invoke  helper
```

```
create    app/helpers/say_helper.rb
invoke   test_unit
invoke   assets
invoke   coffee
create   app/assets/javascripts/say.coffee
invoke   scss
create   app/assets/stylesheets/say.scss
```

Bármilyen tartalom megjelenítéséhez a `routes.rb` alapján a view könyvtárban kell elhelyeznünk az akciónak megfelelő néven egy beágyazott Ruby kódot tartalmazó HTML fájlt (`.rhtml` vagy `.html.erb`).

Az alkalmazás keretét az `app/views/layouts/application.html.erb` fájl definiálja, amely a HTML dokumentum törzs helyén egy beágyazott `yield` parancsot tartalmaz, amely átadja a vezérlést az akció HTML-ének, ami jelen esetben `hello` lesz.

Ez alapján nézzük meg a szokásos Hello, world alkalmazást ezúttal Railsben. Az `app/views/say` könyvtárban létrehozunk egy `hello.html.erb` nevű view-t, amely a `say` controller `hello` akciójához kötődik. Az eredményt a `http://localhost:3000/say/hello` linken ellenőrizhetjük.

```
<h1>Hello , world!</h1>
```

Ez dinamikussá tehetjük az aktuális idő kiírásával.

```
<%= Time.now %>
```

Mivel a nézetbe nem illik logikát rakni, csak a megjelenítendő értéket, ezért áttesszük az idő lekérdezését a kontrollerbe, annak is az akciónak megfelelő metódusába, a `hello`-ba

```
class SayController < ApplicationController
  def hello
    @time=Time.now
  end
end
```

A nézetben pedig csak hivatkozunk a Controller példányváltozóira.

```
<%= @time %>
```

A nézet többnyelvűsítését a `/config/locales/` könyvtárban lévő YML fájlokkal érhetjük el. Definiáljuk a `hello` fordítását és egy időformátumot.

```
en:
  hello: "Hello , world!"

  time:
    formats:
      default: "%Y.%m.%d_%H
      :%M:%S"
      datetime: "%Y.%m.%d_%H:%M:%S"
```

A nézetünkben (`app/views/say/hello.html.erb`) pedig használjuk a fordítást végző `t` (vagy hosszabban `translate`) azonosítójú, és a lokalizációt végző `l` (vagy hosszabban `localize`) azonosítójú függvényt.

```
<h1>%= t :hello %</h1>
<p>%= 1 @time, format: :datetime %</p>
```

A Rails MVC filozófiájának harmadik eleme a modell, amelyet szintén a `rails generate` opciójával hozhatunk létre. A harmadik argumentum a modell osztály neve, amely a konvenció alapján egy egyes számban megadó és a szavakat `_` szimbólummal összefűző string. Ennek többes számú változatával jön létre az az adatbázisban egy tábla. A parancs kiadása négy fájlt hoz létre: egy adatbázis migrációs Ruby szkriptet, egy `ActiveRecord::Base` leszármazottat a modell osztályok közé, egy egységteszteket tartalmazó osztályt és egy tesztadatokat tartalmazó YAML fájlt.

```
kovacsg@debian:~/gyakorlat/config/locales# rails g model user name:string
password:string email:string neptun:string usertype:int
  invoke  active_record
  create  db/migrate/20180925111126_create_users.rb
  create  app/models/user.rb
  invoke  test_unit
  create  test/models/user_test.rb
  create  test/fixtures/users.yml
```

Az adatbázis-migrációs szkriptben az adatmodell változtatásait adjuk meg. Az előző modellgeneráló szkript egy olyan táblát hozna létre, amelyben egy azonosító és két időpecsét attribútum mellett egy `username`, egy `password`, és egy `email` azonosítójú string típusú attribútum szerepelne.

```
class CreateUsers < ActiveRecord::Migration[5.2]
  def change
    create_table :users do |t|
      t.string :name
      t.string :password
      t.string :email
      t.string :neptun
      t.integer :usertype

      t.timestamps
    end
  end
end
```

Ezután elvégezhetjük a tábla struktúrájának módosítását, amiben a `rake` parancs nyújt segítséget. A `rake db:create` létrehozza a táblákat, ha még nem tettük volna meg, a `rake db:migrate` módosítja a séma struktúráját.

```
kovacsg@debian:~/gyakorlat/db/migrate# rails db:migrate
== 20180925111126 CreateUsers: migrating
=====
-- create_table(:users)
--> 0.0207s
== 20180925111126 CreateUsers: migrated (0.0210s)
=====
```

Az adatbáziskezelővel a `gyakorlat_development` adatbázist kiválasztva ezután ellenőrizhetjük, hogy a táblánk valóban létrejött. Alább ennek ellenőrzése látható a két adatbáziskezelő konzolján.

```

kovacs@debian:~/gyakorlat/db/migrate# rails db
MariaDB [gyakorlat_development]> show tables;
+-----+
| Tables_in_gyakorlat_development |
+-----+
| ar_internal_metadata |
| schema_migrations |
| users |
+-----+
3 rows in set (0.01 sec)

MariaDB [gyakorlat_development]> desc users;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id | bigint(20) | NO | PRI | NULL | auto_increment |
| name | varchar(255) | YES | | NULL | |
| password | varchar(255) | YES | | NULL | |
| email | varchar(255) | YES | | NULL | |
| neptun | varchar(255) | YES | | NULL | |
| usertype | int(11) | YES | | NULL | |
| created_at | datetime | NO | | NULL | |
| updated_at | datetime | NO | | NULL | |
+-----+-----+-----+-----+-----+-----+
8 rows in set (0.00 sec)

MariaDB [gyakorlat_development]> select * from users;
Empty set (0.00 sec)

MariaDB [gyakorlat_development]> Bye

```

Nézzük meg, miként tudunk az adatbázishoz Rubyből hozzáférni. Nyissunk egy konzolt, és hozzunk létre egy felhasználót (2. sor), állítsuk be a tulajdonságait (3-8. sorok), mentjük el az adatbázisba (9. sor). Nézzük meg ezután, hogy létrejött-e a rekord az adatbázisban, vagyis a példány `id` attribútuma még mindig `nil`-e? Létrejött a rekord, az azonosítója `1`, amit a modell osztály `find` osztálymetódusának átadva az adatbázisból előkereshetjük az objektumot.

```

kovacs@debian:~/gyakorlat/db/migrate# rails c
Loading development environment (Rails 5.2.1)
irb(main):001:0> u = User.new
(0.2ms) SET NAMES utf8, @@SESSION.sql_mode = CONCAT(CONCAT(@@sql_mode,
',STRICT_ALL_TABLES'), ',NO_AUTO_VALUE_ON_ZERO'), @@SESSION.
sql_auto_is_null = 0, @@SESSION.wait_timeout = 2147483
=> #<User id: nil, name: nil, password: nil, email: nil, neptun: nil,
usertype: nil, created_at: nil, updated_at: nil>
irb(main):002:0> u.name = 'Valaki'
=> "Valaki"
irb(main):003:0> u.password = 'titok'
=> "titok"
irb(main):004:0> u.email = 'valaki@mail.bme.hu'
=> "valaki@mail.bme.hu"
irb(main):005:0> u.neptun = 'aaaaaa'

```

```

=> "aaaaaa"
irb(main):006:0> u.usertype = 0
=> 0
irb(main):007:0> u
=> #<User id: nil, name: "Valaki", password: "titok", email: "valaki@mail.
    bme.hu", neptun: "aaaaaa", usertype: 0, created_at: nil, updated_at: nil
    >
irb(main):008:0> u.save
(0.6ms) BEGIN
User Create (0.3ms) INSERT INTO 'users' ('name', 'password', 'email', '
    neptun', 'usertype', 'created_at', 'updated_at') VALUES ('Valaki', '
    titok', 'valaki@mail.bme.hu', 'aaaaaa', 0, '2018-09-25_11:18:40', '
    2018-09-25_11:18:40')
(7.6ms) COMMIT
=> true
irb(main):009:0>

```

Ellenőrizzük az adatbáziskonzolon, hogy valóban megjelent-e ez adat az adatbázisban.

```

kovacs@debian:~/gyakorlat/db/migrate# rails db
MariaDB [gyakorlat_development]> select * from users;
+-----+-----+-----+-----+-----+-----+
| id | name | password | email | neptun | usertype |
|-----+-----+-----+-----+-----+-----+
| 1 | Valaki | titok | valaki@mail.bme.hu | aaaaaa | 0 |
|-----+-----+-----+-----+-----+-----+
| 2018-09-25 11:18:40 | 2018-09-25 11:18:40 |
+-----+-----+-----+-----+-----+-----+

1 row in set (0.00 sec)

MariaDB [gyakorlat_development]> Bye

```

Az adatok ezután előkereshetők az adatbázisból.

```

kovacs@debian:~/gyakorlat/db/migrate# rails c
Loading development environment (Rails 5.2.1)
irb(main):001:0> u = User.first
(0.2ms) SET NAMES utf8, @@SESSION.sql_mode = CONCAT(CONCAT(@@sql_mode,
    ',STRICT_ALL_TABLES'), 'NO_AUTO_VALUE_ON_ZERO'), @@SESSION.
    sql_auto_is_null = 0, @@SESSION.wait_timeout = 2147483
User Load (0.2ms) SELECT 'users'.* FROM 'users' ORDER BY 'users'.id
    ASC LIMIT 1
=> #<User id: 1, name: "Valaki", password: "titok", email: "valaki@mail.bme.
    hu", neptun: "aaaaaa", usertype: 0, created_at: "2018-09-25 11:18:40",
    updated_at: "2018-09-25 11:18:40">

```

Hozzuk létre a feladatjavító portálunk feladatsoraira vonatkozó model-
lünkét és a hozzá tartozó kontrollert egy paranccsal. Az Quiz modellünkben
legyen egy dátum típusú, `deadline` nevű, az esemény határidejére hivatkozó
mező, egy string típusú cím mező, és egy egész típusú sorszám mező.

```

kovacs@debian:~/gyakorlat/db/migrate# rails g scaffold quiz deadline:date
number:integer title:string
invoke active_record
create db/migrate/20180925112158_create_quizzes.rb

```

```

create    app/models/quiz.rb
invoke   test_unit
create   test/models/quiz_test.rb
create   test/fixtures/quizzes.yml
invoke   resource_route
  route   resources :quizzes
invoke   scaffold_controller
create   app/controllers/quizzes_controller.rb
invoke   erb
create   app/views/quizzes
create   app/views/quizzes/index.html.erb
create   app/views/quizzes/edit.html.erb
create   app/views/quizzes/show.html.erb
create   app/views/quizzes/new.html.erb
create   app/views/quizzes/_form.html.erb
invoke   test_unit
create   test/controllers/quizzes_controller_test.rb
create   test/system/quizzes_test.rb
invoke   helper
create   app/helpers/quizzes_helper.rb
invoke   test_unit
invoke   jbuilder
create   app/views/quizzes/index.json.jbuilder
create   app/views/quizzes/show.json.jbuilder
create   app/views/quizzes/_quiz.json.jbuilder
invoke   assets
invoke   coffee
create   app/assets/javascripts/quizzes.coffee
invoke   scss
create   app/assets/stylesheets/quizzes.scss
invoke   scss
create   app/assets/stylesheets/scaffolds.scss

```

Hajtsuk végre a scaffold létrehozása során keletkezett migrációt.

```

kovacs@debian:~/gyakorlat/config/locales# rails db:migrate
== 20180925112158 CreateQuizzes: migrating
-----
-- create_table(:quizzes)
--> 0.0195 s
== 20180925112158 CreateQuizzes: migrated (0.0196 s)
-----

```

Ezután a böngészőben nyissuk meg a feladatsorok nézetet (<http://localhost:3000/quizzes>), próbáljuk ki a feladatsor létrehozását, törlését, listázását. Nézzük meg, hogy létrejött-e a rekord az adatbázisban, nyissuk meg az adatbázis konzolt:

Ezzel hasonlóan hozzuk létre a feladatsorok feladatainak modelljét, controllerét és nézetét. Egy feladat egy feladatsorhoz tartozik, amelyre annak típusával megegyező néven, `quiz`-zel hivatkozunk. A feladatnak ezen kívül van egész típusú sorszám, egész típusú pontszáma, szöveg típusú feladatszövege és szöveg típusú javítási útmutatója.

```

kovacs@debian:~/gyakorlat/config/locales# rails g scaffold task quiz:
  references number:integer text:string points:integer solution:text
  invoke active_record
  create db/migrate/20180925112707_create_tasks.rb

```



```

create    app/models/task.rb
invoke   test_unit
create   test/models/task_test.rb
create   test/fixtures/tasks.yml
invoke   resource_route
  route  resources :tasks
invoke   scaffold_controller
create   app/controllers/tasks_controller.rb
invoke   erb
create   app/views/tasks
create   app/views/tasks/index.html.erb
create   app/views/tasks/edit.html.erb
create   app/views/tasks/show.html.erb
create   app/views/tasks/new.html.erb
create   app/views/tasks/_form.html.erb
invoke   test_unit
create   test/controllers/tasks_controller_test.rb
create   test/system/tasks_test.rb
invoke   helper
create   app/helpers/tasks_helper.rb
invoke   test_unit
invoke   jbuilder
create   app/views/tasks/index.json.jbuilder
create   app/views/tasks/show.json.jbuilder
create   app/views/tasks/_task.json.jbuilder
invoke   assets
invoke   coffee
create   app/assets/javascripts/tasks.coffee
invoke   scss
create   app/assets/stylesheets/tasks.scss
invoke   scss
identical app/assets/stylesheets/scaffolds.scss

```

Hajtuk végre a migrációt, majd próbáljuk ki ezt is a webfelületen (<http://localhost:3000/tasks>)!

```

kovacsg@debian:~/gyakorlat/config/locales# rails db:migrate== 20180925112707
CreateTasks: migrating =====
-- create_table(:tasks)
-> 0.0210s

```

Hivatkozások

- [1] Sam Ruby, Dave Thomas, and David Heinemeier Hansson et al. *Agile Web Development with Rails*, volume Third Edition. The Pragmatic Bookshelf, 2009 Mar.