

Untitled Note

前書き

- できるならいじり過ぎないほうがいいので、ここで挙げた該当エラーが出た場合のみ修正することを推奨します
- ところどころ出てくるbeは自分の環境で設定した短縮エイリアスなのでbeではなく `bundle exec` と読み換えること
- 環境によって違う項目もあるかもしれないので注意(パスとか引数とか特に)

capistranoでyarnがないエラーの解決

- `nvm => node => yarn`の順番にec2内でinstall
- yarnは別にいらないかも(nodeまで入れてダメならyarnの入れる感じで)
- 以下のサイトを参考に
- <https://qiita.com/nju33/items/aaba3f995b4bb298a13e>

production環境のdb:dorpとcreateしたい問題

- seedでシードデータを用意してるなら気にしなくて良い
- capistranoのtaskを実行してseed_fuで2回目以降にシードデータ作成したいのなら必要な処理
- 基本的にproduction環境でのdbへの破壊的な操作は禁止されているが、それを突破するやり方

```
1 # EC2
2 RAILS_ENV=production DISABLE_DATABASE_ENVIRONMENT_CHECK=1 bundle exec rake db:drop
3 bundle exec bin/rake db:create RAILS_ENV=production
4 # local(再デプロイ)
5 be cap production deploy
```

```
1 db:seed_fuの実行タスクの作成と実行
```

asset precompile時のerrorの解決

- 背景に画像を指定したい場合は`background: url("path");`と書きたくなるがAWSだと`background: image-url("path");`か`background-image: image-url("path");`でないと反映されない
- しかし、.cssの拡張子のファイルにimage-urlを使うとコンパイル時にエラーがでる(ES6以降の書式は使えない)
- 自分の環境では、style.cssをstyle.scssにリネームして.scssファイルにするとコンパイルが通過できた
- image-url以外の要因でコンパイルエラーが出る場合以下を試してみることをオススメする

```
1 config/environments/production.rb内を以下に修正
2 config.assets.js_compressor = Uglifier.new(harmony: true, compress: { unused: false })
```

何回デプロイしても過去releasesのものはや存在しないviewsファイルをみに行くエラーの解決

- productionのlogをみて原因を特定
- これは2回目以降のデプロイで発生したのだが、原因が以下の方法で解決できるのかそれとも再デプロイ時にunicornのプロセスをkillしていないことで反映がうまく行かなかったのか原因が定かでない
- よって、以下の対処方法の内config/unicorn/production.rbをいじる前にunicornプロセスのkillを実行後に再デプロイをして、変更差分が反映されているかを確認することをオススメする

```
1 # unicornの設定をいじることで解決を試みるパターン
2 # config/unicorn/production.rbに以下を追記
3 working_directory "/var/www/i_know_that/current"
4 # 実行するディレクトリを明示的に指定することで解決?
5
6 # 再びデプロイしても変更を認識しない問題のためunicornをkillしてから再デプロイ
7 #unicorn プロセスをkillしてからbe cap production deploy
8 ps -ef | grep unicorn      # ここで表示されるunicornプロセスのpidを下記コマンドで使用
9 kill -QUIT pid            # このコマンドでunicornプロセスを根絶やしにする
10 be cap production deploy   # これはローカルでプロジェクトディレクトリをカレントにして実行
11
12 # (おまけ)EC2内で直でプリコンパイルしたい場合
13 rails assets:precompile RAILS_ENV=production
```

413 Request Entity Too Large問題(nginxのアップロード受け入れ可能サイズ変更)

- デフォルトでは1Mに設定されている
- confの場所はローカルとec2で違うのでその都度調べる
- 今回はec2内のアプリケーション単位の設定ファイルを書き換える(一般設定ファイルではない)
- 場所はカリキュラムのawsのnginx設定の場所に乗ってた気がする

```
1 # nginx.conf内に以下を追記
2 http {
3     server {
4         client_max_body_size 6M;      # <=この1行だけ追加(容量は好きに設定してよい)
5     }
6 }
```

AWSでjqueryが動かない問題

- railsのlink_toでmodthod: :deleteを指定した場所でcssは当たっているのにdeleteではなくgetが飛んでしまったのでこれはjsかjqueryがうまく動作してないことがわかる
- deleteメソッドはgetをjsでdeleteに偽装しているため、getになっていたので気づく

```
1 gem 'jquery-rails'
2 gem 'bootstrap', '~> 4.3.1'
3 $ bundle
4 # application.js
5 //= require jquery3
```

```

6 //= require rails-ujs
7 //= require popper
8 //= require bootstrap-sprockets
9 //= require activestorage
10 //= require turbolinks
11 //= require_tree .
12 # application.css => application.scss
13 *= require "bootstrap/dist/css/bootstrap.min.css" #<= いないかも
14 *= require_tree .
15 *= require_self
16 */
17 @import "bootstrap";

```

yarnからgemでbootstrapとjqueryを読むようにして、assets:precompileを再度実行したらtestが全部落ちた(compile前は部分的にbootstrapを使う部分だけ落ちた)

- ここらへんは何で解決したか謎だから、違う方法を探したほうがいいかもしれません
- また、yarnでインストールしてたbootstrapとjqueryをgemで動かすように設定したあとにyarnでaddしたbootstrapとjqueryをマニフェストで読み込まないようにすると、なぜかテストが落ちる問題
が起きたので、自分は両方読み込むように設定したままにしています

```

1 # 解決策
2 config/environments/test.rb内で定義されている
3 config.action_dispatch.show_exceptions = false
4 # 上記を以下に書き換え
5 config.action_dispatch.show_exceptions = true

```

おまけ

```

1 # capistranoで任意のタスクのみ実行したい場合
2 #書式
3 be cap production 名前空間:タスク名
4
5 # Usage
6 # unicorn restartしたい場合
7 be cap production deploy:restart
8
9 # unicorn restartと言っても、プロセスキルしないと何故かデプロイ差分が認識されないこともあるので
10 # そのときはEC2内でkill -QUIT後に再デプロイしてください

```

```

1 # プロダクション環境のログを見たい場合(i_know_that部分は自分のアプリ名に書き換える)
2 # log
3 cat /var/www/i_know_that/current/log/production.log
4
5 # 最新30行分だけのログを表示したい場合
6 tail -n 30 /var/www/i_know_that/current/log/production.log

```

```

1 # nginx 再起動
2 sudo service nginx restart

```