

Grimoire.js ハンズオン

Grimoire.js はじめに



Grimoire.js

ハンズオンの注意

円滑な準備のためご協力をお願いします

- ・ 本日の資料・必要なファイル類は以下に上がっています。(常にかけるようにしておいてください)

<https://github.com/kyasbal-1994/grimoirejs-handson>

- ・ FirefoxかChromeを入れてください
- ・ 何らかのローカルサーバーを建ててください
 - ・ WindowsならXAMPP、MacならNPMにあるhttp-serverが楽に構築できます。

わからない場合は遠慮なくご質問ください

Wifi

- SSID: mozilla
- Security: WPA-PSK/WPA2-PSK
- PSWD: roppongi756

ごあんない

- ・ 懇親会の参加者は1000円をお支払いください。
(参加と答えてなくても参加可能ですが、17時に締め切ります)
- ・ ご意見や疑問はハッシュタグ`#grimoire_handson`でツイートしていただけるとウォッチしているので気がつきます
- ・ 写真・動画等を適宜撮りますが、写りたくない方はご申告ください。
- ・ 本日のスタッフは赤い紐の名札をしています。

タイムスケジュール

(時間は適宜調整することがあります)

受付時間	13:30 - 14:00
Mozilla Japan様より挨拶	14:00 - 14:10
Grimoire.js概要・さわってみる	14:10-14:55(45分)
休憩	10分
Grimoire.jsことはじめ	15:05 - 16:10(65分)
休憩	10分
モデルビューワーを作ろう	16:20 - 17:05(45分)
休憩	10分
マテリアルをいじってみよう	17:15 - 18:00(45分)
閉会(のち懇親会、希望者のみ)	18:00 ~

ToC

- WebGLの活用例/Grimoire.jsでできること
- Grimoire.jsの概要/特徴
- ライブラリを使ってみよう(ハンズオン)

WebGLの活用例

- ・ 今まで: ゲーム目的
- ・ これから: Webサービス/ Webデザインの一環も
- ・ 例: AppleのWebサイト、石油のやつ、グリルのやつ

**WebGLはもうWebエンジニアに
手の届かない技術じゃない**

一つのWebデザインのテクニックへ



Grimoire.js

Web 開発のための WebGL フレームワーク

Grimoire.jsで何ができるの？

- ・ 画像リストと連携したWebGL
- ・ 寿司のGUI
- ・ GGJで作ったゲーム

Grimoire.js概要

- Chrome/Firefox/Safari/Edgeで動くライブラリ
(ただし、Edgeはバグにより動作せず(v0.14.0現在))
- Typescriptにより構築され、Javascriptからでも利用可能
- npm及びscriptタグを用いて使用を開始
- まだまだベータ版

Grimoire.jsを使ってみる

ここからハンズオン

必要なファイルを作る(前準備)

- 1. 適当な作業用フォルダを作成します
(ローカルサーバーから参照できる場所が良い)
- 2. 空のindex.html , index.js , index.gomlを作成します。
- 3. 適当にHTMLを揃えます。
html内にbodyとheadがあれば良い。
(ローカルサーバーでアクセス可能か見てください)

ライブラリをインポート

- CDNを利用してライブラリを読み込ませます
- scriptタグでsrc属性でアドレスを
`grimoirejs-preset-basic`のものにします。

GOMLを書く

- index.gomlに一番基本的なgomlを記述します
 - Session1/01.gomlを参照
- html内の埋め込みたい位置にscriptタグを入れてtype=“text/goml”としてsrc=“index.goml”を指定します。

カメラを動かそうにしてみる

- cameraタグの子要素を以下のようにする。

```
<camera>  
  <camera.components>  
    <MouseCameraControl/>  
  </camera.components>  
</camera/>
```

この記法の詳細な意味については、2つ目のセッションで詳しく解説する。

テクスチャを張ってみよう

- meshにtexture属性を指定する。
- Session1/02.gomlを参照

ライトを使ってみる

- ・ ライティング関係はまだ開発途上
- ・ CDNで`grimoirejs-forward-shading`プラグインを読み込む(preset-basicの下で読み込む)

ライトを使ってみる

- lightノードを追加する
- type属性は`point`,`spot`,`directional`が指定可能、colorで色、intensityで強さ

モデルを読み込んでみる

- モデルも開発途上
- CDNで`grimoirejs-gltf`プラグインを読み込む

モデルを読み込んでみる

- modelタグを追加して、srcにモデルファイルへのファイルパスを追加する
- Session1/04.gomlを参照