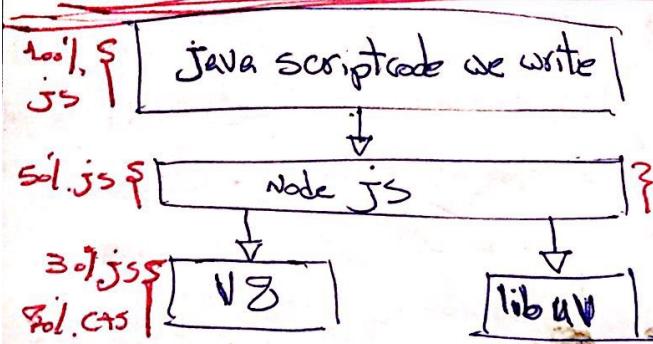


Node.js - advanced topics

(160 Videos)



مودودی اگر کوئی حکایتی حاصل از کچھ اسے
node اپنے دستی میں مالی سستھی کے لئے دیکھو۔
C++ راجی نہ کرے بلکہ اسی library از C++ دوست کرو۔

* خود داری برپا کردن این ساخته های فلزی ساده با هدف دست و ... است که براحتی بسیار امکان استفاده از آنها را بسیار ممکن کند. داشتن این قابلیت می تواند در مواردی که می خواهیم کار را راحت نماییم، می تواند بسیار مفید باشد.

← در پروژه خود فولدر **طبا** حلوی قرار گیرد. حاوی اسلایری است که داده در درون خود اسپرینچ می‌شوند. فولدر **src** نیز حلوی می‌باشد. محتوازی **src** فانلسن هی مارکوبی اسح دلسا از آنرا استفاده می‌شوند. تا بجانب دهاتی مانند **app** گزینه **src** به دفعه سه‌گانه برویم. بخشی از فعل گذاشتگارها همچوی دختر فولدر طبا است و بخشی از لایادسکریپت بر استفاده کرده اند. در فولدر **src** صورت **cat** است. تحویل **includes** in **index.html** کوئن توایع **cat** صورت زیرا است:

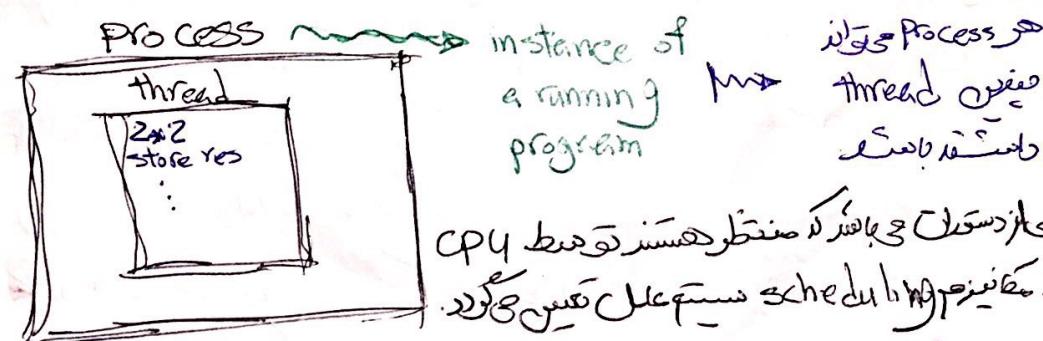
```
const { PBKDF2 } = process.binding('crypto');
```

~~connects JS & C++~~

functions

متوافق ۸ لایه‌ی مکانیکی اسکرین کاری باشد و مفهوم را توجه کنید که آنها با استفاده از اسکرین کاری \leftrightarrow استفاده کنند و آنها هم امکان دسترسی به لایه‌های زیرین دارند رام مکانیکی \leftrightarrow دسترسی

Event loop : در فرود بروک هنگام کردن کارهای Lasync استفاده می شود.



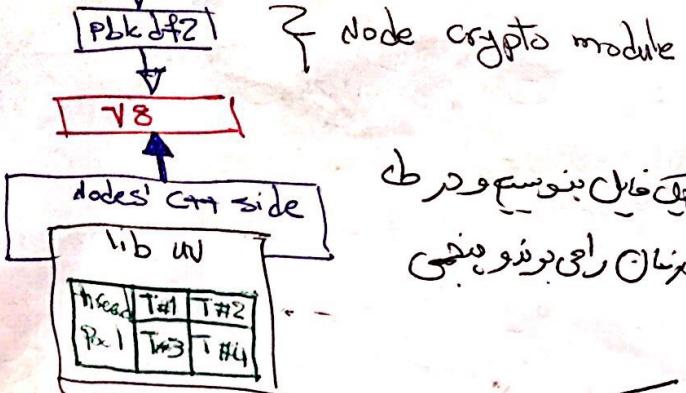
بردهای thread یا unit های از دستگاه حسنه را متنظر هستند تا همین طبقه CP4 اجرا شوند. اگری آنها قسم特 مکانیزم scheduling سیستم عمل تحسین چیزی کرد.

لہجہ Node دینے والے thread اس کا وہ نام طبیعی صورت میں ہے۔

نهی از خالقین چیزی که در library استفاده node مکار دارد در واقع Single thread نیست!
نهی بعنی (خالقین کسی node کار در زنده ندارند) از آنرا استفاده کنیم خارج (Event loop) برای اینها

جیسے حنوناً خارج کرنے والے سینگل ٹرینڈ اور ایک

با این حالت میتوانید استاندارد C++ mode را در library های مورد نظر خود ایجاد کنید.



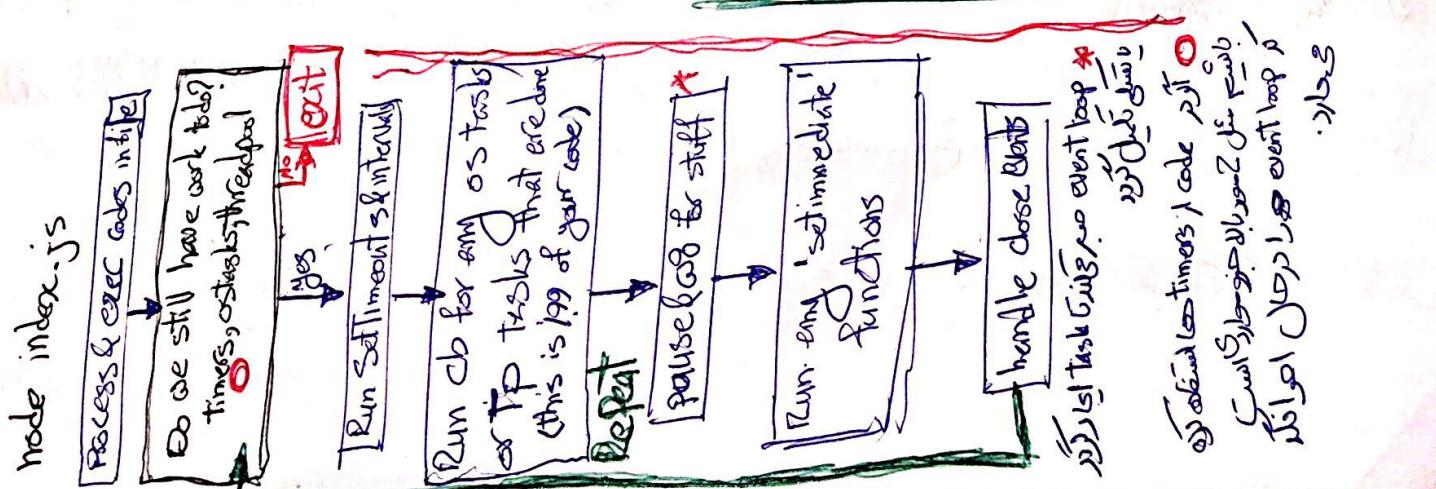
مکانیزم Thread pool می باشد که درین مکانیزم چندین Thread برای یک Task از یک Thread pool استفاده می شوند و هر کدام این Thread برای یک Task خاص خود را دریافت می کنند.

التي تفتح اربع خوارim بالبلوك threadpool باين دسكتور حسنه دعم:
Process.env.UV_THREADPOOL_SIZE = 2

لایج ورزل Cryptocurrency میں بستگی حاصل ہے وہی حاصل ہے کوئی سوکھ نہیں اگر برادر Boose CPUs باشے۔ جو دنیا میں زبان بیرونی کے طبق اور کوئی
م طور پر سمجھی جائے تو اس کیمپ میں مانگوں کیمپ ایک دفعہ ایک سوکھ جاتے تھے اس کے علاوہ جو
ارکیو Cryptos کا جیسا میں استعمال اڑاکنے کی اس طرح ایک ایسا جو اینٹرینیم SJ لکھا ہے تو سیم دہ
بیولدرز IP استھا ہے لہر، کامیابی پر یہی IP Pending Operation داخل ہوئے (لہر کا باعث مارے
پاسدہ event loop کی جملہ درحال اور باقی جیسا۔

علوه بر `TP libuv` فا نلشون هیچ وجود دارد لایع فکاره و بینیم Underlying `libuv` هایلایت استفاده کند. معلماتی این وعی که درخواست `http://را همیان` می شود و در طبق زمان این را لایع بیند همیان این را این را در عاقع خود `OS Asynchron helpers` `libuv` بازیابی کردار دو از لایر که `OS` استفاده کند، سئل همیان درخواست `http://` که در عاقع قسط که این را در عاقع بیند `OS` مسئول همیان کرد این درخواست های این را در `event loop` برپا نمایم این را پنجه

• مطروح كلّي عارض مرتبط بـ networking يستعمل في قطع
OS's sync feature
• إنّما يكدرّ كرنبي لـ event loop وعود ممّا يهدّد Pending Tasks



جتنی بروزی مجموعه ای دستورات چند دستور در یک خالی برینم = Multi tasking

ابن ایم یک درست http است، ایم یک hash با استفاده از f2 که f1 و f2 خالی خواش باشد. حالا جلسه

function doRequest() {
 log(1)

function doHash() {
 log(2)

DoRequest(),

fs.readFile(' - ', 'utf8', () =>
 log(3)

) ?

doHash();

doHash();

doHash();

doHash();

۱۰۰٪ رود و حساب آن ارکی بروزد. آن ۱۰۰٪ خودی حذف کی دارد که دارای دو pause جای باشد.

اعلی درخواست خواش از f1 دسترسی و سایر stats، f2 که f1 دارد و دسترسی باز خواش خالی است.

ابن ایم، ابنا ۳ hash و fs چار، TP، السیگل کردند. وقتی که باز اولی فوران TP آن

۱۰۰٪ جای انتظار hash چهار خود را درخواست کردند. در این حین یکی از hashها ابراز نمودند

خرنگی بود. همچنان که ابراهیم TP از این طبقه کلید و وقایع جواب بار اول fs از f1 بروزی کرد،

قلدری نیز و با سخاخ خروجی بود. میں سایر hashها در TP بر اینرا و بخوبی بروزند.

Terminal

log(1) حسنه

log(2) سریع اجرایی کرد زیرا خواص خالی

از داخل سیم جی بهتر دلالت دارد.

log(3) این مقدار کم است

log(2) این مقدار کم است

log(3) این مقدار کم است

log(2) این مقدار کم است

log(3) این مقدار کم است

log(2) این مقدار کم است

log(3) این مقدار کم است

log(2) این مقدار کم است

log(3) این مقدار کم است

log(2) این مقدار کم است

log(3) این مقدار کم است

log(2) این مقدار کم است

log(3) این مقدار کم است

log(2) این مقدار کم است

log(3) این مقدار کم است

log(2) این مقدار کم است

log(3) این مقدار کم است

log(2) این مقدار کم است

log(3) این مقدار کم است

log(2) این مقدار کم است

log(3) این مقدار کم است

log(2) این مقدار کم است

log(3) این مقدار کم است

log(2) این مقدار کم است

log(3) این مقدار کم است

log(2) این مقدار کم است

log(3) این مقدار کم است

برای این FS:

بعنوان معمول همان پایه ها معمول در FS هست.
و همچنان که call یعنی مسافت کم که f1 بتواند
پایه ای خود را باز کرد، این را می تواند

f1 را درstats باز کرد و بعد از آن f2 را باز کند. می تواند

node.stats را در حالتی باز کرد که f1 باز باشد.

برای بردازدن

(Video #20)

افزایش کارایی Node

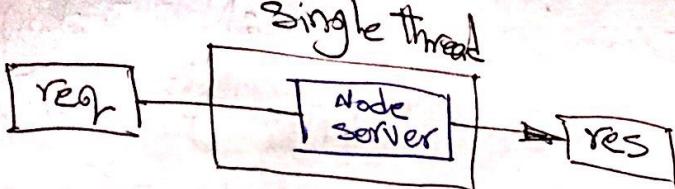
use node in "cluster" mode

? Recommended

use worker threads

? experimental

2

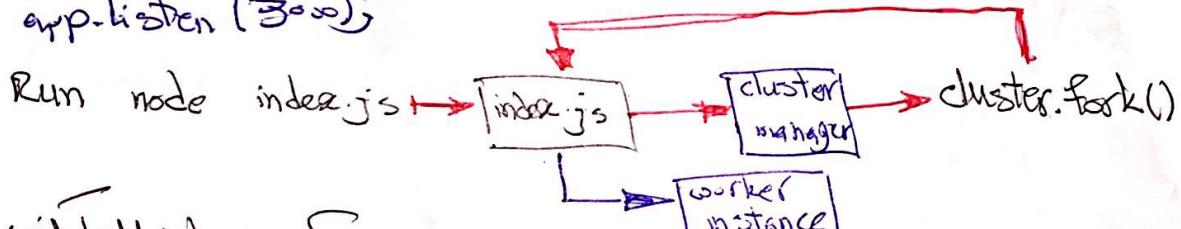


مُحدّث خواستی ملایم سرعت دادن و تحریر کرد درین لاین Threaded event loop - جی بس سریعتر کار کرد
که هر آن ارسال می گردید. سئل از آنجی بی شروع و پنهان

شل: وقی مایک حلفر هائین هاند زیر دارم، هنین حلقدر، تو یک express سرمه است، هالی لادھاٹان که تقدیز نایی در ہو است لارام کر ڈرام پع آگرا بسی رزا نہ است.

دوبارہ بعد در `ent loop` سا عمل دارو تاکہ نتیجہ ہمیں `req` میں پاسخ دادہ گئے۔
`const express = require('express');`
`" app = express(),`
`function doWork(duration) {`
`const start = Date.now(),`
`while (Date.now() - start < duration) {`
`app.get('/1', (req, res) => {`

بواي بیوود performance سے حق تر انہیں اسٹریٹیجی cluster بنوائیں جو مختصر مدت کے لئے بھروسہ رہتے۔



* cluster manager این دلایلی است که در حوزه Cloud می تواند یکی با استفاده از یک خالص index.js این را اجرا کند. بعلاوه با اول اجرای وسیله فن cluster manager (یادی کرد و هر یک سفار از worker instance) این را اجرا کند. بعلاوه با این دلایل این دلایلی است که در حوزه Cloud می تواند یکی با استفاده از یک خالص index.js این را اجرا کند.

```
const cluster = require('cluster');
if (cluster.isMaster) {
    // cause index.js to be executed again in child mode.
    cluster.fork();
} else {
    // I'm a child mode, I'm going to act like Server.
    // all of the above code for creating server
}
```

درکن اگر باشند این برای میرا $fork$ است. با تغییر کنال همراه سطح مطالعه ماتریس M و \mathbf{y} کنونه عمل نموده
خواهد شد که در اینجا M از $n \times n$ است. n کل کارهای n است و \mathbf{y} هم مطالعه ماتریس M است. حال با همین دلیل
که هر کارهای n داریم و آن را که در خواست f تو دیگر منویسیم که $f = 5$ به این دلیل دلسته باشد و
تقریباً هر کار اگر کنیم در خواست f هم باشد $f = 5$ به این دلیل دلسته باشد $f = 5$ به این دلیل دلسته باشد و
جی بریدتا اگر f استور اما اگر f چنین $fork$ دلسته باشد و همین کارای آردهم در خواست f دوں دلیله دلایا مثلاً جوابی f هم

بالا بدلن تعداد cluster های ابر مریودیکلر این همیشه حوب هست! با این نتایج نتایجی (رسور جو درست آوریم و برآن معنای داشتیم، از Appache bench mark

(ست ریسیت با ۲ هسته) ! appache bench mark

تا آن Thread pool را برابر این داشتم و 2 fork داشتم با همیم و 2 bench mark همی (رسور جو درست آوریم) چی بود داست باشیم. مثلاً ۲ کلک ۱ لیسیم و ۱ افریان و ۱ هما فتن hash را در یوت که بدل باشیم هر لایی داشتم و res درخواست get را در طبقه هما فتن hash آغاز کارم.

حال با اسال ۲ درخواست همیزی را دو fork خواهیم داشت = ۱ process.env.UV_THREADPOOL_SIZE

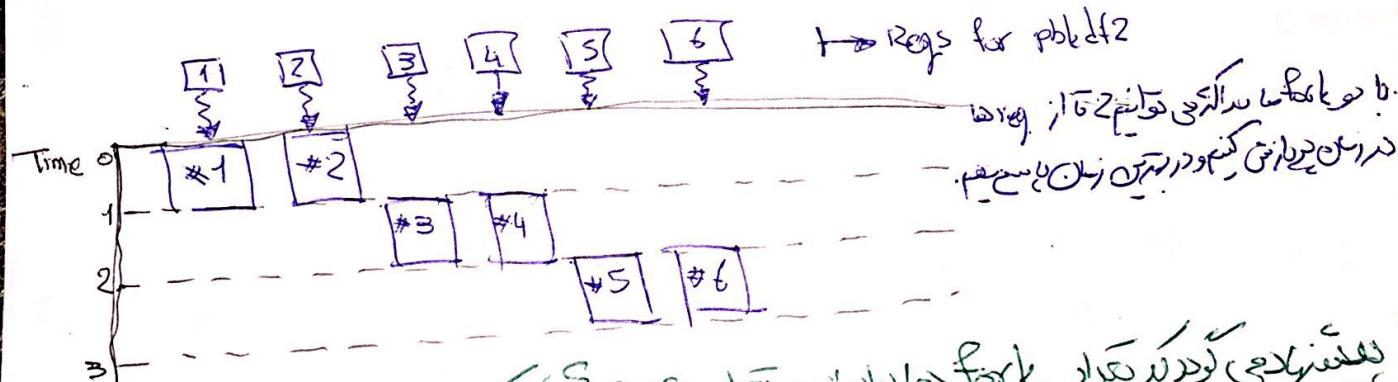
۳ مجموع concurrent بگیری پاسخ دهیم که با ازای هر درخواست ۲ نیاز است.

cluster.fork()
ز (۱۰)

نکه: آن ترتیب fork داشتم هر درخواست ۲ نیاز داشت اهر آنکه

حال آن با زارکن ما fork های را سلیمانی هفت کسر تواند از اینجا در یعنی ۶ عدد و کار درخواست همیزی ایجاد نمی شوند. فتح بعیت است درخواست ها در حدود ۳.۵ تا ۴ این افرادی همیشده که با این مسلمه زیادی طرد ایزا CPU نلایی خواهد کرد با بالاترین کار خوبی همیزی را درخواست همیزی داشتند این over allocated cluster می شوند.

حال آنکه تعداد کاربر ۲ بروزگاه و ۲ همیزی ۶ بخواهند را که این ۳ خواهد بود از ۲۲۵۰ زنگ از



لطفاً لایی تعداد fork های را این تعداد کارهای CPU نهایت نمودم

در بحث production می بینیم که استفاده از ۲ thread یا ۲ worker threads یا clustering یا web workers threads یا event loop یکی هستند اینها همیشه ممکن است استفاده نشوند! ممکن است ممکن است ۲ thread را از ۳ هسته کنم. لیکن ممکن است خود ۲۴۷ از ۲۴۸ thread را ایجاد کنم و کار خواهیم داشت. توجه کنید که نه ممکن است اینها استفاده (۲) بپرسید ایجاد نمی کنند.

(Video #35)

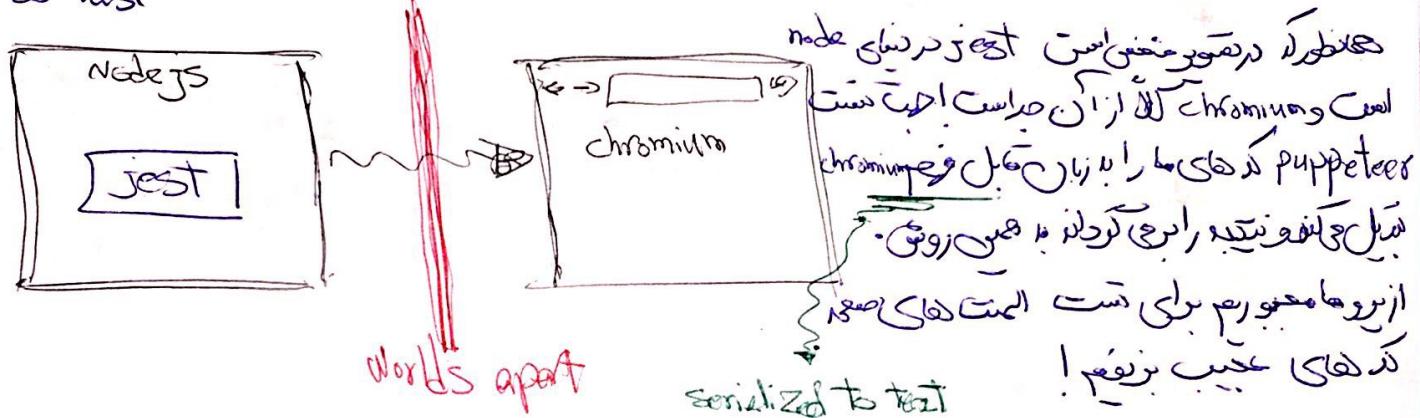
3

video #65



برکتِ سنت ایلیاس از همان نعمت آنکه زین کار بر می رود و می بینیم و این خود را حاجات chromium در دارفل بینند لور
می بینیم و اینکه کاربر در حال لیبل کار می باشد است سانس ایار بلکه جدیده لاین و ... از درون جنادیک اینکه
می دعسی تا اطمینان حاصل کنیم که درستی کار رخشد.

chromium a headless browser. This means it has no display for interaction, so instead we should interact with through our code. We can run it from command line and it is so fast



```
const test = await page.$eval('a.brand-log', el => el.innerHTML);
```

برگشت بهزار لاین سایه ایچ در میانه این کلمه . شکل این حساس است لا بزرگ لاین از طبقه دستیAuth provider ها سند google , chromistim , و دیگر عوامل مختلف عیسی اس و captchas و حفظ ای راه حل chromium با fake session هایی صورت است که بار و سی سایه ای همراه با

نحوين Already لـ **پیشنهاد** (مراجعه) **پیشنهاد** ! (header.test.js و index.js و cookiesession services/Passport.js)

سے تکونہ از ای خرچ سکوئے دھوافت لائیں یا بعد و لائیں کر۔ جو inspect جسے network ایجینٹ کر۔ درسی آن درنام ای callback وجوددارد جو سمت راس بھی Response تو اپنے دو عمار set cookie میں کارڈ اکرا ایکسیشن کو لے جاؤ۔ میرار دھی ہے اور این اس browsed session کے دلائل کا ما لائیں ہے جس کے browses ای دسکریپٹن request کی جیسا نہ تھا باہر کو ما لائیں ہے۔ حالا میرار Session، copy کرنے۔ تاصل از و مطرود کر ریورابنوسنر کا داخل آکر راسکارہ خانہ!
 * In our server session, it's used to ensure session does not touch.

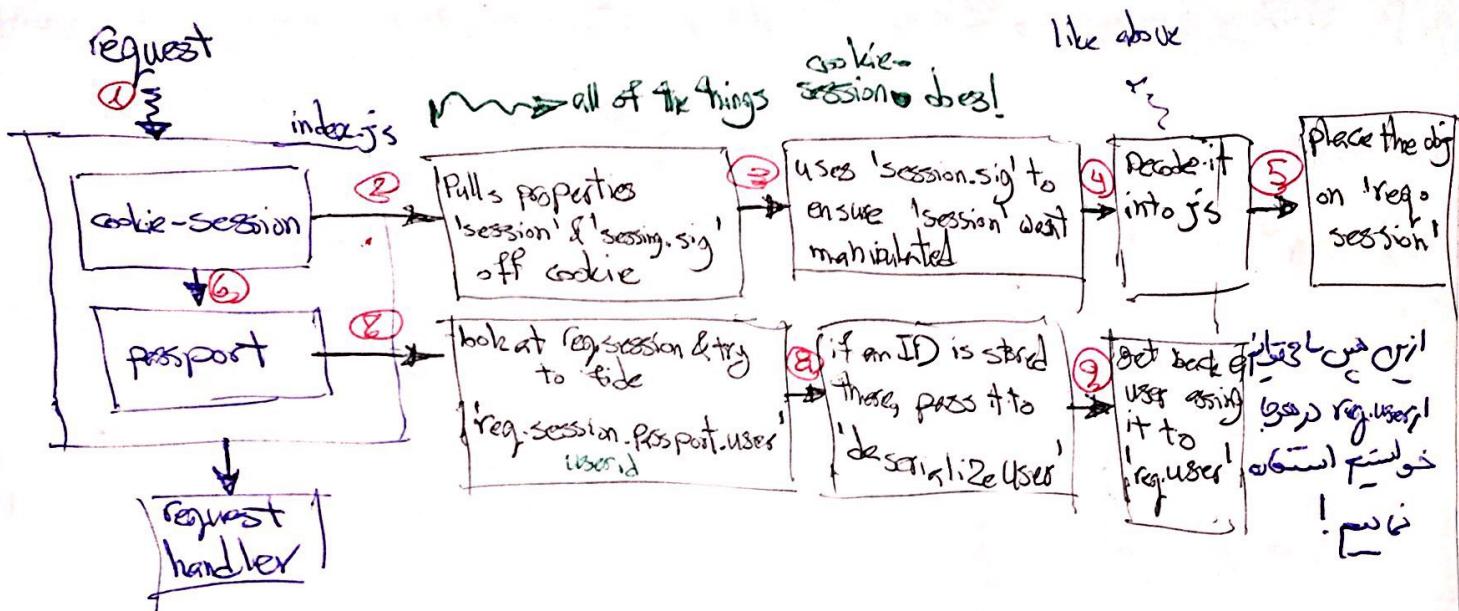
Terminel: nl : node

> const session = 'xxx what we copy from browser';
> ~~session~~ → ~~Session~~

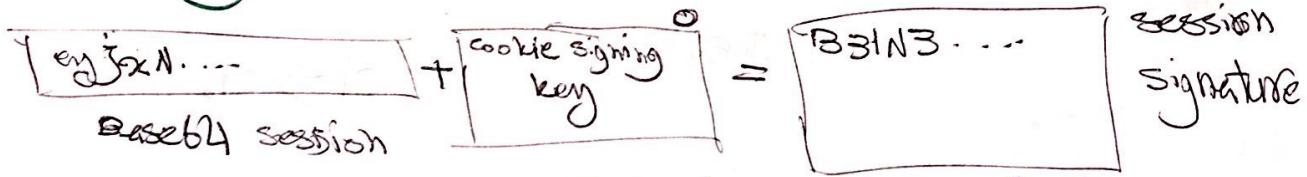
```
> Buffer.const Buffer = require('safe-buffer').Buffer;  
> Buffer.from()
```

~~'s "Passport" : { "user": "userId" } }~~

شہریک روپہ اسکالا جائیک لئد Passport دارا ہی ریجیو ڈی جی مائن



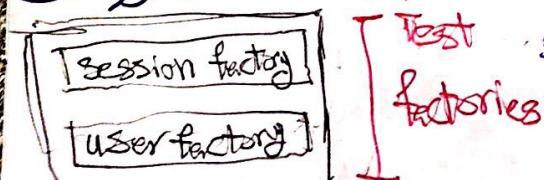
Session. Signature :



مکالمہ و ادراجه میں کسی مستقلات اپنی (ام) یعنی کاریخ و قانون ایجاد (Establishment Session) میں ایک خود بریشن (یعنی کوئی) میں لے را تعمیر دھدرو یا ہمارے دل کے لیے ایک نوٹ ٹکٹ کو حاصل کرنے والے دستیں کر دیں۔

ما در کردن آن را تغییر نمی‌کنیم و باید این باسکت public هستند. با این روش می‌توانیم از طریق صفحه مکالمه یک‌نمونه داشتن از key (key session) را در session با قابلیت خود را داشته باشیم. در این لیست config / delljs mcastkey نمایش داده شده است.

از تکار لرها برای نیت استفاده خواهند شد. سلله تولید آزادی دارد و بسیاری از هسته های این را دارند.



نحوه خالی دسترسی دارد لذا از `test` است. این روشهای `server` و `test` ممکن است هم براست. آنرا بخواهیم از طبیعت های در کلاس `Page` تعریف نموده است استناداً به `mongoose` باید آنرا در حیطه نسبت تعریف کنیم. سازه استناده اسلوکی `setup` است برای `Tests` که در `Chromium` خود را بخواهد. بعای نیز `test` منویم شود لاثر از آن است داشتم `setup` که این را اجرا نمود در `Package.json` بالای `scripts` باید آن `test` باشد این فایل را بخوبی می‌کنیم.

الفون برای همان هزاری فرآیند لاین جیت بلکبری از فرادران لدهای بازگشتن `Page` در کامپیو و اینم لاین های فواید بخواهیم که رویرو لاین انجام شود. این که در صورتی که `test`. استناده شود `await Page.login()` بجای `new session` که ممکن است `chromium` و بازگشتن `session` و سمتگذرنده و...

بلکه این که سیب این `cache` اینم دارم و با آنها ذکر در سطح `Page` نداریم و `mongoose` راه حل هایی می‌توانیم بازیابی `session` کرد. بازگشتن `session` و بازگشتن `chromium` و سمتگذرنده.

۱) اضافه کردن متالاین با کلاس `Page` در `puppeteer` نمودن `CustomPage` اینم داشم و من خطا چشم گذاشت
۲) توانیم از کلاس `Page` معمود در `puppeteer` نیز استفاده کنیم و لاین را در آن اینم داشم.
اقطع این نوعی این استند نداشتم
۳) راهی نهی توانیم `chromium` را بخوبی کنیم بجا کلاس `CustomPage` که ممکن است این استناده کند!

۴) استناده از کلاس `Page` در `CustomPage` که بتوانیم راهی نیز ممکن است داشتم دسترسی پیدا کنیم.
class Page {
 assume that Page class in Puppeteer
 is something like below!
 goto () { console.log('new page') }
 setCookie() { console.log('set cookie') }
}

class CustomPage {
 constructor (Page) { this.page = Page }
 login () {
 this.page.goto('localhost:3001');
 this.page.setCookie();
 }
}

اعتفاده از کهای بخوبی نیز ممکن است
// const page = browser.launch(); in real world.
const page = new Page();
const customPage = new CustomPage (page)
customPage.login();
customPage.page.goto();
customPage.page.goto();

ستکل هیچ ڈاں اسے کر دیں دستیں برمدھی خود کلنس Page مدام باز Page FunctionName

لکوار کئیم! (لکھی صفحہ پر رادر codepen.io چوناں نہ سچ کئیم)

in customPage class under login method:

get() {

this.page.get()

setCookie() {

this.page.setCookie()

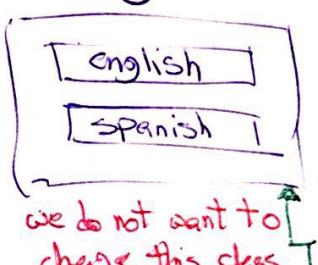
کوئی نہ تھا جس کی وجہ سے class را ڈاں روپی در کلنس

customPage کویت کئیم کر داں روپی ہم بیمار

رنج برائے!

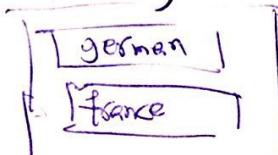
لہا) فرق لئے کہ اسی داریم کے اسکے تغیرات نہیں بلکہ اسی احتیاج داریم کے علاوہ برلاں میں ملی کلنس جنہیں ہم نیاز داریم۔ بہ ہم جب نیاز بہ ابڑی داریم کے تغیرات استاد از این کلنس ہمارا سعیرت لدے

greeting class

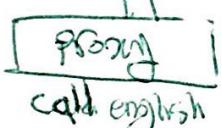


we do not want to change this class

more Greeting Class



a class we write



بعد گردانو، حال یوں تغیرات کی کلنس سے نیاز داریم زبان کی
بیسکی ہوں اضافہ کئیم۔ سعیرت ایسا بوساسی زبان ارکان
کلنس استاد کے تغیرات نیاز بہ ابڑا داریم کے طور پر
ES 2015 بوجود آکدو بہ احراز ہو (کوئی بیکنے یا حیند زمط
دسترسی نہیں کیا ہے)۔

جیسا کہ اسی معنی پر کلنس کی استاد تھے۔

ES 2015 کے function constructor (Proxy)

لہوڑت اہ گلو و ہو دار ایجی نیاری بے یا

داریم۔ درویں کے علاوہ برلاں ہایکے استاد کے عطا

اک ڈی اسٹ object کے set of functions اسے درکان اسے۔ یوں

کلنسوں کے ایک نہیں بے جمہ اعلیٰ دسترسی میں ہے (moreGreetings)

ہریاں ایجی سٹھن۔

target[function] & target[property] است اسیو جیز

سے get و ہوئی است سارے ایجی کوئی کم علاوہ بر

اپنے property کے من استاد کے لئے خواجہ کوئی کم اترد (کوئی کوئی

بندوں کے لئے بے دبیل آن چکر)۔

```

console.clear();
class Greetings {
    english() { return 'Hello'; }
    spanish() { return 'Hola'; }
}
class MoreGreetings {
    german() { return 'Hallo'; }
    french() { return 'Bonjour'; }
}

```

const greetings = new Greetings();

const moreGreetings = new MoreGreetings();

const allGreetings = new Proxy({ greetings, moreGreetings },

get: function(target, property) { return target[property] || greetings[property]; }

console.log(allGreetings.english());

moreGreetings کے property

greetings کے property

لَّهُمَّ : در شکل مدل اگر بحاجی (Hello) در کنول لگ ، (1) را نکارید، پس معنی است که این متغیر `inital` نشود لذا در خروجی سفار روبرو بحاجی جو مسدود = ؟؛ زیرا `english()` return 'Hello' است.

ندروایچ `reference` این طایفه انسان است و ممکن است اینها نیستند.

با استفاده از این داشتی `new` که جو اینستین یک ساختن `Page` را در `CustomPage` برای ساختن `Page` را در `CustomPage` در کلاس `CustomPage` قرار دهیم.

ساخته کل کلاس `CustomPage` مابه صورت زیر نگهفته شد:

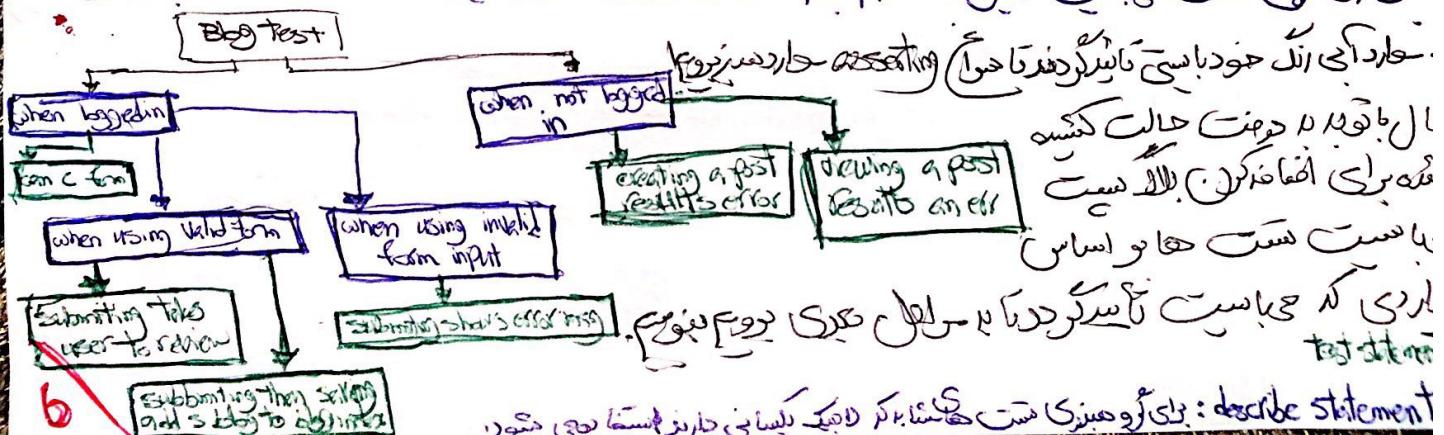
```
class CustomPage {
    static build() {
        const page = new Page();
        // CustomPage=new CustomPage(page);
        const superpage = new Proxy(customPage, {
            get: function(target, property) {
                return target[property] || page[property];
            }
        });
        return superpage;
    }
}
const superPage = CustomPage.build();
superPage.login();
superPage.goto();
;
```

پس از این 4 روشی - که یعنی تابعی ایجاد کننده حیثیت ای داشته باشیم که لاتین همین باده!

Video #11

نحوه `jest` و `mocha` در این تست در چهار قسمتی می خواهد
حمدرا با چشم افراحتی این در صورتی که `mocha` نبوده باشد `it` و `xit` این کار را انجام می دهند.
نحوه هارا کوچکتر نمی در فایلهای سازمانیم تا بصورت مجزا (ندره) بستر ایجاد کنند.

عمل از نسخه سمت جیاست تحلیل کلمه ایم ایم معنود و سواری نمی باشد که دست نماید.



برای روشنیزی سمت `describe` کنید که این متنی دارینه مستعاری دارد.

دستوراتی که child statement های دیگر را بگیرند و با خود آنها describe statement می‌شوند. برای مثال beforeEach, afterEach و beforeEachDS هستند. بولدر ساخته شده توسط آنها زیر است. (blog.test.js در نمونه)

(Left wing tree) video #111

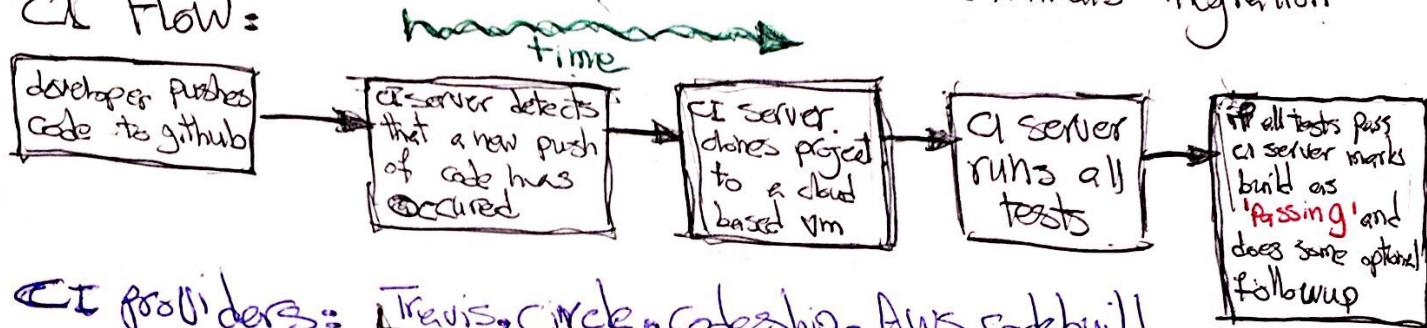
ویدیو #112: پروژه انتقال کدهای سایت سمعان (اعلی) در یک repo گیت ویس را بگذارید.

Process to merge all code changes into a single branch.

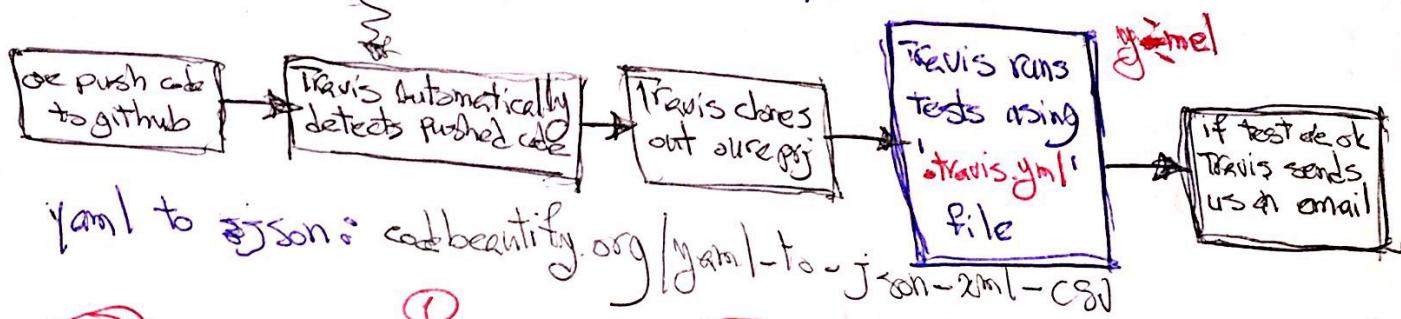
Server that runs automatic checks (tests) on the codebase to ensure the changes have not broken anything

CI = continuous integration

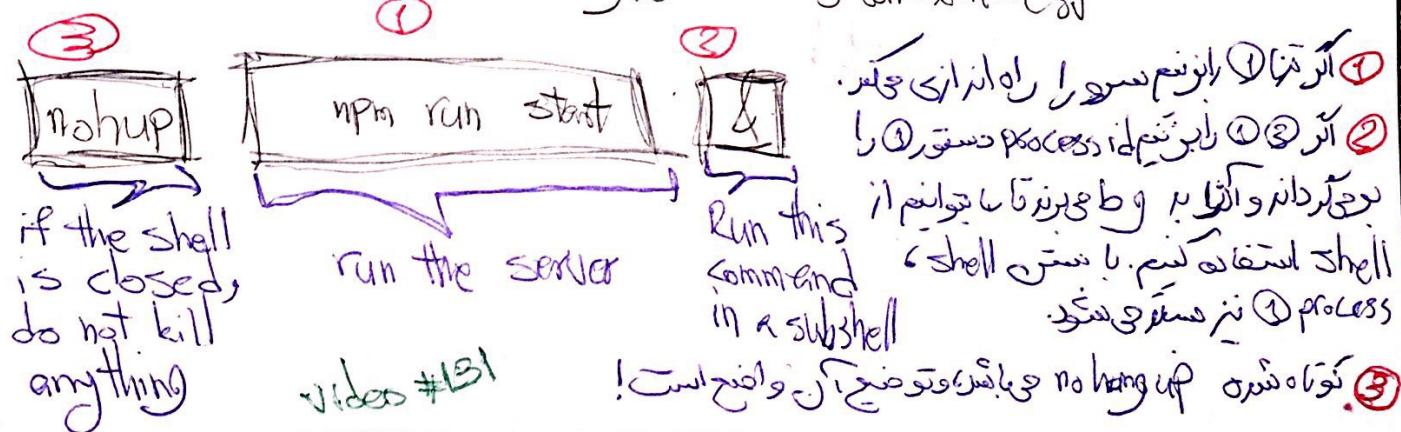
CI Flow:



CI providers: Travis, Circle, Codeship, AWS CodeBuild



yaml to json: codebeautify.org/yaml-to-json-xml-csv



No یا video #140: AWS S3

