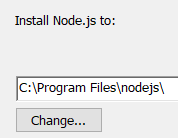
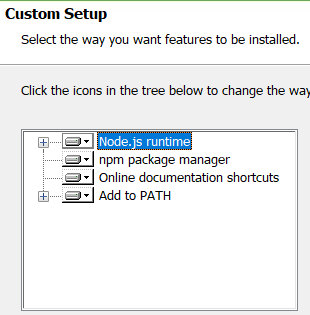
# 1. Install Ionic

## 1. Cài đặt nodejs:

site: nodejs.org

Hiện tại thì mình xài nodejs 8, có 10 nhưng chưa stable nên xài 8.



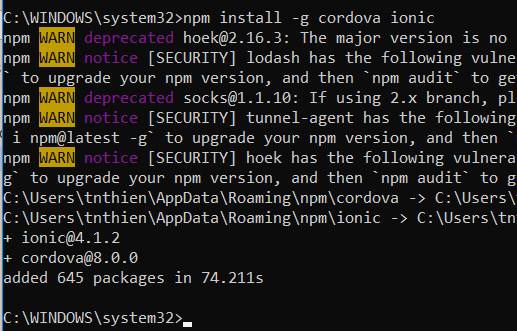


## 2. pull ionic

Sau khi cài xong, bạn mở cửa sổ console lên, nhập vào:

*npm install -g cordova ionic*

Chờ khoảng 10p, việc cài đặt đã xong rồi đấy



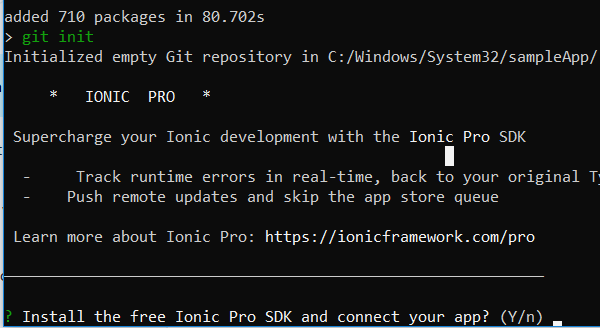
## 3. Tạo ứng dụng đầu tiên

Tham khảo ở <http://ionicframework.com/getting-started/>

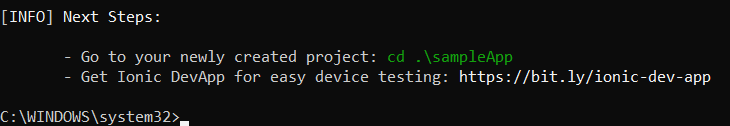
Chạy trên cmd:

*ionic start sampleApp tabs*

đầu tiên chọn y thử: (muốn hiểu tham khảo link <https://ionicframework.com/docs/intro/deploying/>)

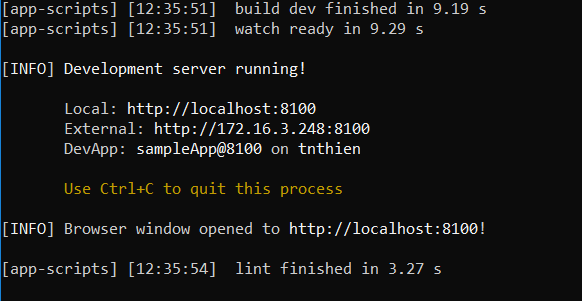


Chọn N hết thì:

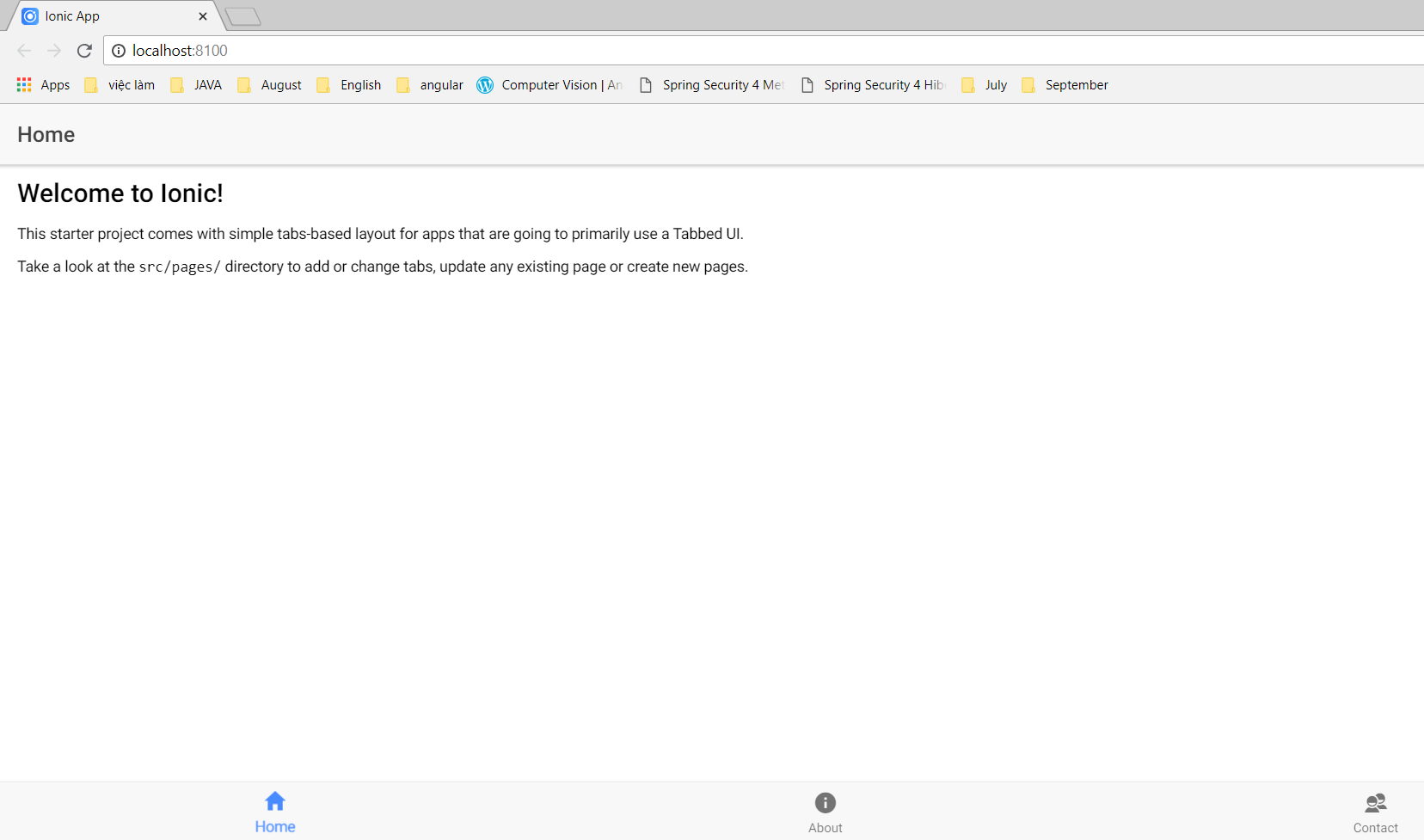


## 4. Start ionic

Chạy trên cmd lệnh này: ionic serve



Sau đó sẽ view được trên chrome:



## 4.4 Ionic 4

Nó hỏi chơi Ionic 4 không ok chơi luôn, choose n cái vụ SDK Ionic gì thôi, kết quả cuối cùng:

[INFO] Next Steps:

- Go to your newly created project: cd .\sampleApp

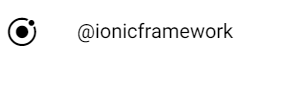
- Get Ionic DevApp for easy device testing: <https://bit.ly/ionic-dev-app>

## 5. Item placement

| **Attribute** | **Description** |
| --- | --- |
| item-start | Placed to the left of all other elements, outside of the inner item. |
| item-end | Placed to the right of all other elements, inside of the inner item, outside of the input wrapper. |
| item-content | Placed to the right of any ion-label, inside of the input wrapper. |

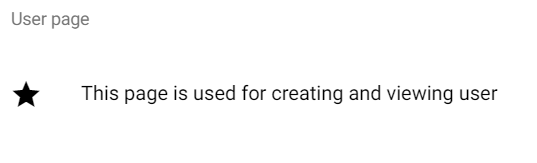
## 6. name attribute in icon-icon

<ion-icon name="ionic" item-start></ion-icon>



<ion-icon name="star" item-start></ion-icon>

This page is used for creating and viewing user



<ion-item>

<ion-icon name="star" item-start></ion-icon>

This page is used for creating and viewing user

</ion-item>

<ion-item>

<ion-icon name="home" item-start></ion-icon>

This is second item in icon-list

</ion-item>



## 7. tabIcon:

<https://ionicframework.com/docs/ionicons/>

## 8. HttpClient:

<https://www.djamware.com/post/59924f9080aca768e4d2b12e/ionic-3-consuming-rest-api-using-new-angular-43-httpclient>

Chạy command: ionic g provider Rest

Hình như không cần chạy lệnh trên vẫn xài được

Nó sẽ tạo ra 1 file:

Rest.ts

Và chỉnh sửa app.module.ts để thêm thằng rest này vào

## 9. Promise

Promise là một type đặc biệt của 1 Object có thể sử dụng hay để cấu trúc việc xử lý bất đồng bộ.

Một Promise có 3 trạng thái **pending**, **resolved** và **rejected.**

Cách phổ biến sử dụng:

new Promise(/\* executor\*/ function (resolve, reject) { ... } );

## 10. Router

Can not find angular/router:

npm install @angular/router

<https://viblo.asia/p/co-ban-ve-router-trong-angular-2-63vKjn7yK2R>

Sync dựa theo modal:

<https://www.youtube.com/watch?v=T5iGAAypGBA>

<https://stackoverflow.com/questions/47000913/how-to-get-entire-object-using-navparams-ionic-3-angular-4>

## 11. click event on ionic

<https://stackoverflow.com/questions/39847181/ionic2-ion-list-with-a-button-both-having-click-event>

## 12. Storage Ionic

<https://www.youtube.com/watch?v=lLhH5GM1qk0>

<https://www.youtube.com/watch?v=2_xpwdoMLAg>

Đầu tiên phải chạy command line này:

ionic cordova plugin add cordova-sqlite-storage

npm install --save @ionic-native/sqlite

## 13. Test ios

Hình như không chạy sqLite trên web được, nên phải chạy trên ios hoặc android:

Ios:

<https://ionicframework.com/docs/v1/guide/testing.html>

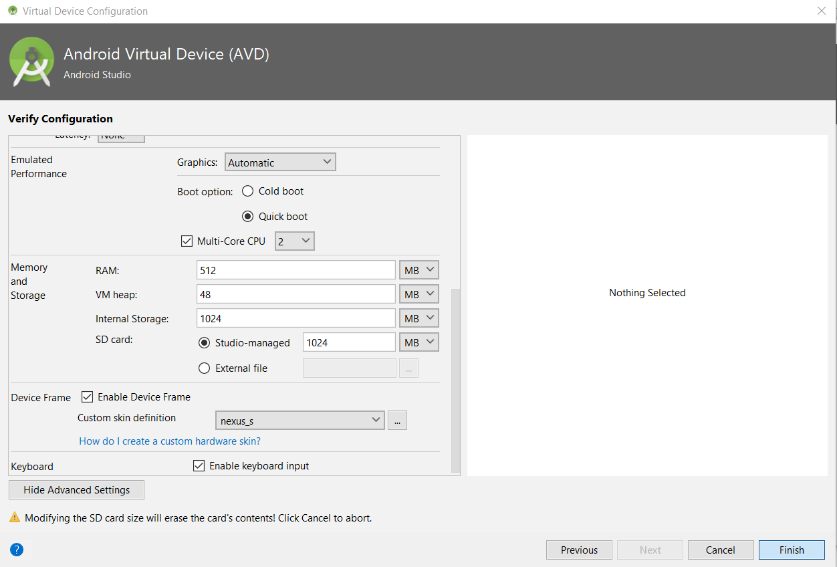
ionic cordova build ios

ionic cordova emulate ios

## 14. Test android

1. Install android studio

<https://www.youtube.com/watch?v=ZWrKakQOCYE>



down vote

Bị lỗi hardware

As per [this response](https://stackoverflow.com/a/27997670/1515058), the complete steps are:

1) Open SDK Manager (In Android Studio, go to Tools > Android > SDK Manager) and Download Intel x86 Emulator Accelerator (HAXM installer) if you haven't.

2) Now go to your SDK directory C:\users\%USERNAME%\AppData\Local\Android\sdk\extras\intel\Hardware\_Accelerated\_Execution\_Manager\ and run the file named intelhaxm-android.exe.

In case you get an error like "Intel virtualization technology (vt,vt-x) is not enabled". Go to your BIOS settings and enable Hardware Virtualization.

3) Restart Android Studio and then try to start the AVD again.

Search all:

Ctrl+shift+a

Run ionic:

Ionic cordova run android –prod

Require JDK 1.8:

Config 3 places of environment:

JAVA\_HOME

JRE\_HOME

Path

Cordova not available:

<https://stackoverflow.com/questions/40815183/ionic-2-cordova-is-not-available-make-sure-to-include-cordova-js-or-run-in-a-d>

Old file:

"use strict";

Object.defineProperty(exports, "\_\_esModule", { value: true });

var path = require("path");

exports.LOGGER\_DIR = '\_\_ion-dev-server';

exports.IONIC\_LAB\_URL = '/ionic-lab';

exports.IOS\_PLATFORM\_PATHS = [path.join('platforms', 'ios', 'www')];

exports.ANDROID\_PLATFORM\_PATHS = [

path.join('platforms', 'android', 'assets', 'www'),

path.join('platforms', 'android', 'app', 'src', 'main', 'assets', 'www')

];

## 15. Date picker in Ionic

ionic cordova plugin add cordova-plugin-datepicker

npm install --save @ionic-native/date-picker

## 16. tag ionic

<https://stackblitz.com/edit/ionic-tags-input-wn2jdi?file=pages%2Fhome%2Fhome.html>

npm install ionic-tags-input –save

Sửa **import** { Platform } **from '@ionic/angular'** trong ion-tags-input.d.ts và tabs-tabs-module.js.map

## 17. rest trên ionic 4

ionic g service rest-api

# 3. GoWithFriends Group

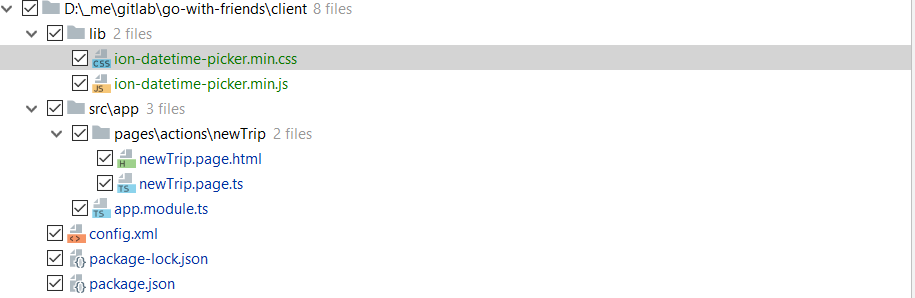
Mockup:

<https://balsamiq.cloud/s8raxo2/padoy87/r1676>

Management task:

<https://app.asana.com/0/850110033751800/850110033751782>

# 4. Không commit tạm thời:

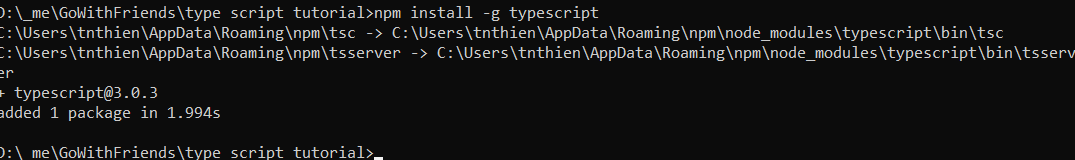


# 2. Tìm hiểu chơi TypeScript

## 1. Sơ sơ

Setup tyepscript:

npm install -g typescript



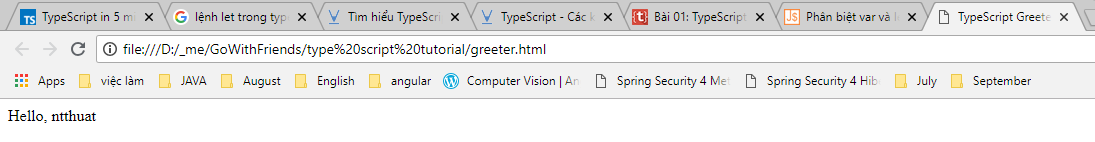
Viết greeter.ts xong chạy lệnh:

tsc greeter.ts



Nó sẽ generate ra greeter.js

Sau đó dựa vào file html greeter.html sẽ view được kết quả:



### Vì sao cần sử dụng typescript

* Dễ dàng tổ chức code cho các ứng dụng cực lớn và phức tạp nhờ modules, namespaces và hỗ trợ OOP mạnh mẽ.
* TypeScript có một bước biên dịch thành JavaScript, sẽ bắt tất cả các loại lỗi trước khi chúng chạy và làm hỏng một vài thứ.

…

## 2. Var và let

* **Var** và **let** đều là từ khoá – keyword.
* **Var** và **let** cùng được dùng để khai báo biến trong [JavaScript](https://completejavascript.com/category/javascript/javascript-co-ban/)

Let được sử dụng từ ES6, let là sử dụng cho biến cục bộ, var sử dụng cho biến toàn cục, vậy thôi.

## 3. Generics

Generics là function cho phép 1 method nhưng chấp nhận nhiều kiểu param truyền vào khác nhau

Ví dụ Generics với param truyền vào là array.

**function** *genericFunc*<T>(argument: T): T[] {  
 **let** arrayOfT: T[] = [];  
 arrayOfT.push(argument);  
 **return** arrayOfT;  
}  
  
**let *arrayFromString*** = *genericFunc*<**string**>(**"beep"**);  
***console***.log(***arrayFromString***[0]); *// "beep"****console***.log(**typeof *arrayFromString***[0]); *// String***let *arrayFromNumber*** = *genericFunc*(42);  
***console***.log(***arrayFromNumber***[0]); *// 42****console***.log(**typeof *arrayFromNumber***[0]); *// number*

**Kết quả:**

beep

string

10 42

Number

Ví dụ Generics với param truyền vào không phải array

**function** *identity*<T>(arg: T): T {  
 **return** arg;  
}  
  
**let *output*** = *identity*<**string**>(**"myString"**); *// type of output will be 'string'****console***.log(***output***);  
***output*** = *identity*(**"myString2"**);  
***console***.log(***output***);  
  
**let *outputNumber*** = *identity*<**number**>(12345);  
***console***.log(***outputNumber***);

Kết quả:



## 4. Array trong Typescript

Khai báo và sử dụng như sau:

**let *alphas***:**string**[];  
***alphas*** = [**"1"**,**"2"**,**"3"**,**"4"**]  
***console***.log(***alphas***[0]);  
***console***.log(***alphas***[1]);

### Push()

Adds one or more elements to the end of an array and returns the new length of the array.

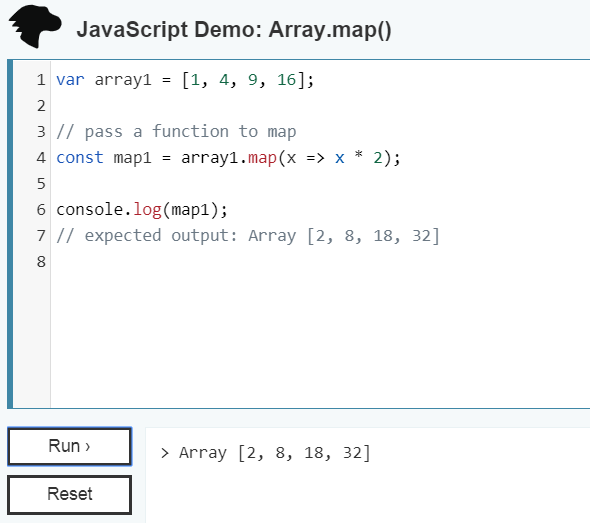
Push 1 phần tử mới vào cuối array và tăng kích thước array lên.

***alphas***.push(**"5"**);  
***console***.log(***alphas***[4]);

Kết quả ra console ra 5

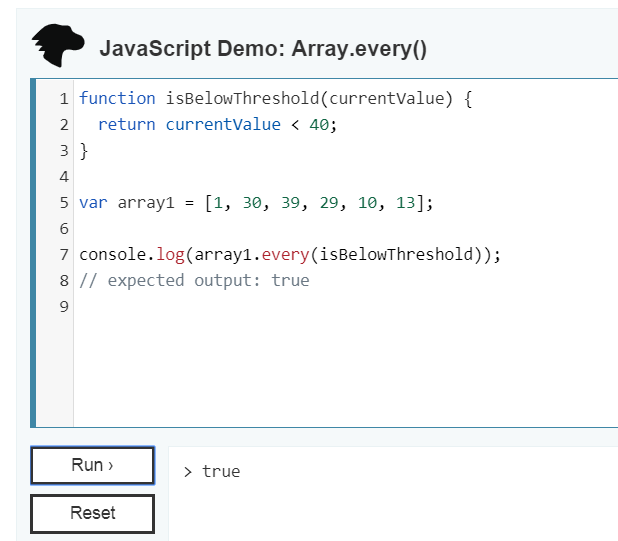
### Map()

Hàm này khá giống map trong java8:

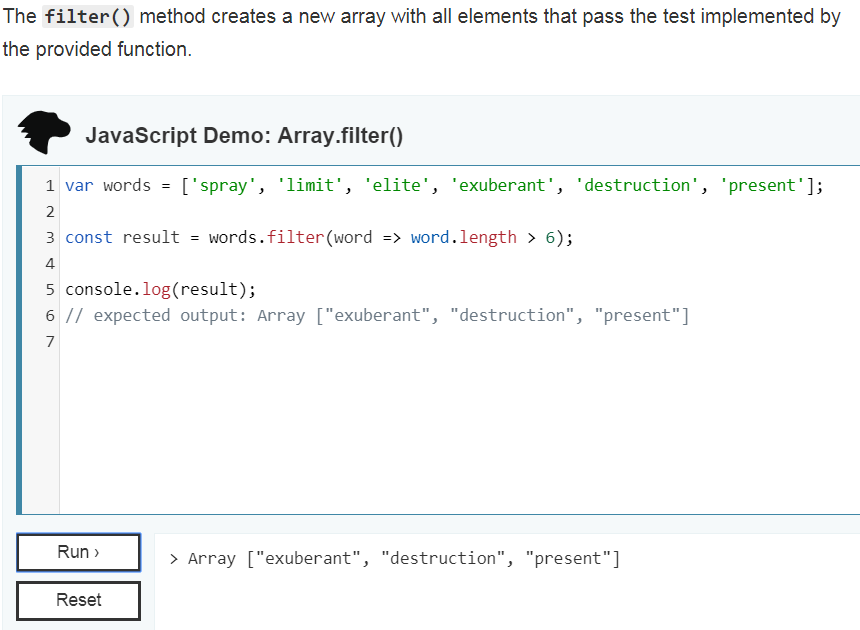


### Every()

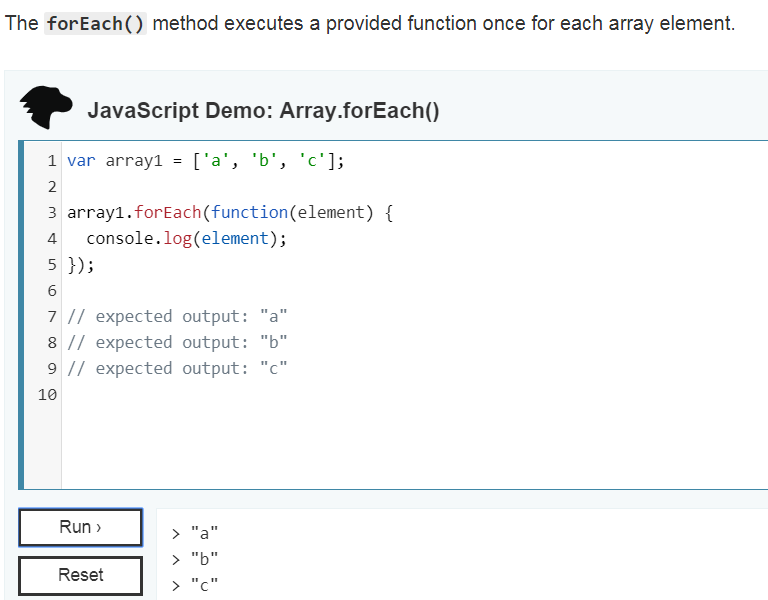
<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/every>



### Filter()



### ForEach()



## 5. Tuple

Tuple giống như Array nhưng cho nhiều type khác nhau trong element.

Ví dụ:

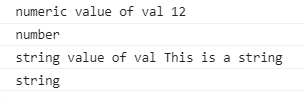
**var *mytuple*** = [10, **"Hello"**]; *//create a tuple  
logTypeAndValue*(***mytuple***[0]);  
*logTypeAndValue*(***mytuple***[1]);  
  
**function** *logTypeAndValue*(input) {  
 ***console***.log(**typeof** input + **': '** + input);  
}



## 6. Union

Cho phép 1 biến có thể có nhiều type

**var *val***:**string**|**number**;  
***val*** = 12;  
***console***.log(**"numeric value of val "**+***val***);  
***console***.log(**typeof *val***);  
***val*** = **"This is a string"**;  
***console***.log(**"string value of val "**+***val***);  
***console***.log(**typeof *val***);



## 7. Interface

Về cái này thì chắc hẳn nó giống trong java

## 8. Import và Export

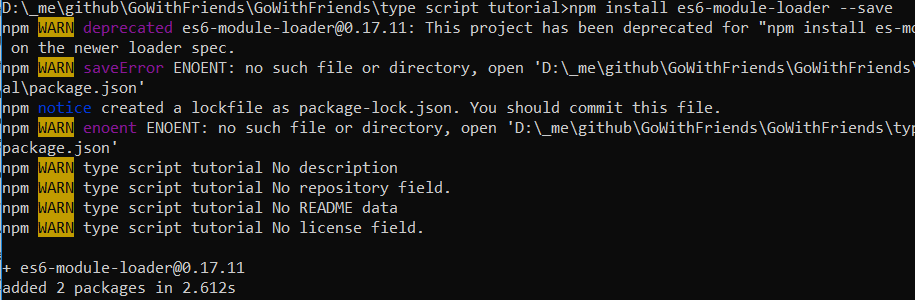
Cái này dùng để thể hiện tính chất encapsulation.

Trong khi export nghĩa là gói gọn lại (đóng gói lại), import là giống như java

Chạy báo lỗi **Uncaught ReferenceError: exports is not defined** nghĩa là setup cái es6-module-loader

Cần setup:

npm install es6-module-loader –save



npm install traceur –save

npm install systemjs

npm install lite-server -g

TODO: không chạy được - fuck