



НИ Е ВЯРВАМЕ ВЪВ ВАШЕТО БЪДЕЩЕ

Page Load Ime



Зареждане и обработка

- За да покаже една страница в крайният ѝ вид, браузъра минава през няколко основни етапа:
 - теглене на HTML документа (download)
 - прочитане на **head** частта и изтегляне на всички линкове в head-a (пак download)
 - парсване (обработка, идва от думата **parse**) на всичко в head-a: CSS файлове, фонтове, генериране на CSSOM-a, изпълняване на скриптовете, генериране на BOM (**window** обекта)
 - прочитане на **body**
 - парсване на html-a и генериране на DOM-a (**document** обекта), теглене на допълнителни скриптове и картинки, генериране на CSSOM-a
 - визуализация (чак след като всичко по-горе е изпълнено)

Good to know..

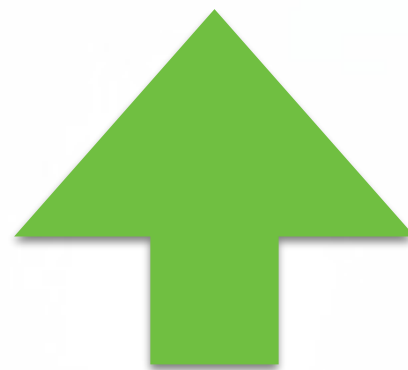
- Скоростта на зареждане е критична за потребителя
 - ако страницата се забави над 10секунди, много от потребителите се отказват да я чакат)
- Броят и видът на ресурсите, които страницата зарежда допълнително (фонтове, css, скриптове, картинки) са критични за времето за зареждане на страницата
- Начинът и последователността на включване на ресурсите може много да подобри performance-a на страницата
 - no `<script>` in the head!
 - or: **async/defer** the scripts (not fully supported so avoid)
 - generate dynamically all extra content after DOM ready

First things - first!

- Когато искаме страницата ни да се зареди бързо, трябва много внимателно да си организираме линковете и скриптовете
- Прието е всичко, което е необходимо за генерирането на визията на страницата, да се зарежда най-напред
- Това са CSS, fonts, background картинки и евентуално някои скриптове, от които зависи показването на съдържание
- Зареждането на всичко останало се отлага, за да може потребителя да не усети забавяне в зареждането на страницата

Златното правило:
style - горе ; script - долу

CSS



JS

Заклучение

- Запомнете, че браузърът винаги зарежда линковете в head-а преди всичко останало
- Запомнете също, че скриптовете, които са там, ги тегли и ги изпълнява още преди да е парснат html кода !!
(т.е. преди да е заредил DOM-а в document обекта)
- Запомнете, че визуализацията на страницата се случва едва след като е зареден DOM-а
- И не забравяйте, че скоростта на зареждане и показване (визуализация) на страницата е критична!

За четене

- Performance by Google: <https://developers.google.com/web/fundamentals/performance/?hl=en>
- Optimizing the Critical Rendering Path: <http://www.sitepoint.com/optimizing-critical-rendering-path/>
- Малко по-авансeд: <http://devnet.kentico.com/articles/optimizing-page-load-time> (статията е от 2011!)
- Fast enough: <http://www.vinta.com.br/blog/2015/web-page-optimization.html>
- async и defer: <http://peter.sh/experiments/asynchronous-and-deferred-javascript-execution-explained/>

Въпроси?

MAKE ALL EXAMPLES



1. Напишете функция **greeting**, която приема параметър `name` и като резултат връща стринг, който поздравява.

Например:

```
greeting("Jeni"); // "Hello, Jeni!"
```

2. Напишете клас **Person**, който да задава собствено и фамилно име на конструираните обекти.

Пример:

```
new Person("Бойко", "Борисов"); // Person { firstName: "Бойко",  
lastName: "Борисов" }
```

3. Направете втора функция, която да поздравява, но я кръстете **personGreeting** и вместо име ѝ подавайте обект от тип **Person**.

Пример:

```
personGreeting(new Person("Бойко", "Борисов")); // "Hello, Бойко  
Борисов!"
```


4. Направете така, че последната функция да приема още един параметър `beFormal`, който може да има стойност `true` или `false`. Ако е `true`, нека поздрав да е официален (т.е. по двете имена на човека), а ако е `false` - да е само по първо име.

Пример:

```
var person = new Person("Бойко", "Борисов");  
personGreeting(person, true); // "Hello, Бойко Борисов!"  
personGreeting(person, false); // "Hi, Бойко!"
```

5. Направете функция (метод) `introduce` в класа `Person`, която да връща стринг, който представя текущия обект (т.е. човека, който е инстанция на класа)

Пример:

```
var person = new Person("Бойко", "Борисов");  
person.introduce(); // "Здравейте, казвам се Бойко Борисов!"
```

6. Добавете още едно поле (property) в класа Person, което се казва isPolite

Нека при създаване на обект от този клас, това поле да се задава автоматично на true

Пример:

```
new Person("Бойко", "Борисов"); // Person { firstName: "Бойко",  
lastName: "Борисов", isPolite: true }
```

7. Направете функция (метод) bePolite() в класа Person. Тя трябва да може да променя isPolite пропъртито по подаден параметър, който може да е true или false

Пример:

```
var person = new Person("Бойко", "Борисов");  
person.isPolite; // true  
person.bePolite(false);  
person.isPolite; // false
```


8. Променете метода `introduce`, така че да връща учтиво представяне, когато `isPolite` полето е `true` и неформално представяне, когато `isPolite` е `false`

Пример:

```
var person = new Person("Бойко", "Борисов");  
person.introduce(); // "Казвам се Бойко Борисов!"  
person.bePolite(false);  
person.introduce(); // "Аз съм Бойко!"
```

9. Направете функция (метод) `greet()` в класа `Person`, която получава като параметър друг човек (`greet(otherPerson)`) и която поздравява другия човек:

Пример:

```
var boiko = new Person("Бойко", "Борисов");  
var lili = new Person("Лили", "Иванова");  
boiko.introduce(); // "Казвам се Бойко Борисов!"  
lili.bePolite(false);  
lili.greet(boiko); // "Здрасти, Бойко!"  
lili.introduce(); // "Аз съм Лили."
```


10. Променете метода `introduce`, така че да може да получава параметър от тип `Person` (`introduce(otherPerson)`). Комбинирайте `greet` и `introduce` като използвате резултата от `greet(otherPerson)` в `introduce`.

Променете `greet`, така че да проверява дали параметъра `otherPerson` е подаден (`if (otherPerson !== undefined) {..}`). Ако да - тогава поздравете по име (използвайте съответно учтива и неучтива форма, както досега).

Ако не е подаден параметър `otherPerson`, тогава просто върнете само поздрав без име ("`Здрасти!`" / "`Здравейте!`")

Пример:

```
var boiko = new Person("Бойко", "Борисов");
var lili = new Person("Лили", "Иванова");
boiko.introduce(); // "Здравейте! Казвам се Бойко Борисов!"
lili.bePolite(false);
lili.introduce(boiko); // "Здрасти, Бойко! Аз съм Лили."
```

Въпроси?

Полезни връзки



**KEEP
CALM
AND
LEARN
JAVASCRIPT**

Примери

<http://zenlabs.pro/courses/lessons/lesson16/examples.zip>

Домашно

<https://github.com/zzeni/swift-academy-homeworks/blob/fe-03/tasks/L16>