

**In order to understand the true power of web development, a look at the past, present, and future of this broad spectrum of services is necessary.**

**The Past of Web Development**

The inventor of the World Wide Web, Tim Berners-Lee, published the first website on August 6th, 1991 from a lab in the Swiss Alps. At the time, he used HyperText Markup Language (HTML) to write the browser and server software. When the HTML Internet draft expired, CHSS rose in popularity. Similar to today’s CSS, it ruled 1994 before giving way to Bert Bos’ Stream-based Style Sheet Proposal (SSP) in 1995. However, HTML made a comeback and its second version, which was completed in 1995, became the basic standard on which future implementations were based. HTML continued strongly after 1997’s HTML 3.2 release and 4.01 release in 1999. However, other languages came in between, including CSS, ECMA-262, and WCAG.

When it came to using these, presentational data was used for what is now the job of style sheets. Meanwhile, scripts manipulated document contents as well as eyes, causing structure and presentation to overlap. This went against the ideal world theory where maintainable web development is achieved by separating structure, presentation, and behavior while ensuring minor overlap for the sake of integration and interfaces. However, this is understandable considering the technology limitations and support limitations arising from incomplete, inconsistent, or non-interoperability. **Low output quality was another issue that arose due to inaccessible, unmaintainable, or invalid code.** Finally, experience was very bad because of the lack of awareness of usability and the slow performance of bloated sites and slow connections.

After the threat of the Y2K bug was well over, web developers put their heart and soul to separate behavior, structure, and presentation to achieve the ideal world. XHTML versions 1.0, 1.1 and 2.0 followed by WCAG 2.0 gave developers the chance to let behavior move over to manipulate more documents in an obtrusive way. Structure and presentation didn’t overlap as much since developers were using less presentational markup. Still the same issues were there. There weren’t sufficient features for web applications and support was still limited since implementations were incomplete. However, while output quality remained low, experiences became suboptimal as Internet speeds increased. Slow, bloated sites along with the lack of awareness of usability still continued plaguing developers.

**The Present of Web Development**

**From 2010 through today, CSS and HTML 5 became an integral part of every web developer’s toolkit.**These and other technologies that came in between allowed the separation of structure, presentation, and behavior, leading developers to slowly achieve the ideal world and setting them on the path to succeed in the coming years. Though many of the issues continued over the years, developers grew more committed to standardizing procedures and creating non-volatile best practices, especially for accessibility, performance, and maintainability.

The present also witness **the rise of mobile web,** directing developers to learn how to create websites and web apps for the users. Though Nokia commercially offered the technology for the first time in 1996, it took almost a decade and a half to become popular. One of the factors that made this evolution possible is the arrival of XHTML at the time. However, new mobile devices and their browsers can now support a range of web formats, including HTML variants that were used by desktops. Still, there are a few limitations developers have yet to overcome, including small screen size, lack of cookies, slow service, and cost of mobile Internet in some parts of the world where a flat fee per month isn’t offered.

**The Future of Web Development**

In the future, web development is in for a rapid change. Experts believe that the industry will expand to new platforms and reach out to new audiences. With the increase of mobile products, web **developers will be challenged to build sites that run EFFECTIVELY on both desktops and smartphones.**To reduce the burden on their shoulders, languages are expected to grow easier. Take for example Python; the programming language is very easy to understand since it reads like a language rather than a program. Combined with hardware’s increasing power, future developers won’t need to worry about tech specs as much.

Further helping developers gain new skills is the growth of open dynamic communities that expertly tackle all the aspects of web development. There are numerous groups and meet-ups designed to bring together developers and allow them to discuss different tools. New developers are encouraged to attend these events to get their feet wet and/or secure projects, which can deepen their knowledge and expertise. And yes, [**jobs are plenty**](http://greeneggmedia.com/blog/entry/web-development-opportunities-slated-to-increase-in-2016) in this industry thanks to buyers and sellers’ increased reliance on the Internet.

## Who are the innovators of web technologies? past, present, and future[[edit](https://en.wikiversity.org/w/index.php?title=Emerging_Web_Technologies&action=edit&section=2)]

[Bill Gates](https://en.wikipedia.org/wiki/Bill_Gates) There wouldn't be much of an interest in the web if we didn't have a PC in every home and office with a GUI interfaced operating system on it. We can thank Bill Gates and his team for that.

[Steve Jobs](https://en.wikipedia.org/wiki/Steve_Jobs) It can't go without being said that Steve Jobs was the one who pushed the envelope in web development by his innovative ideas and team at Apple and Pixar.

[Vannevar\_Bush](https://en.wikipedia.org/wiki/Vannevar_Bush) The man who first conceived HTML. (1945)

## Emerging Web Technology Glossary[[edit](https://en.wikiversity.org/w/index.php?title=Emerging_Web_Technologies&action=edit&section=6)]

### AJAX [[[Wikipedia:AJAX|AJAX]][[edit](https://en.wikiversity.org/w/index.php?title=Emerging_Web_Technologies&action=edit&section=7)]

Asynchronous JavaScript and XML is a group of technologies used to create asynchronous web applications

### API [API](https://en.wikipedia.org/wiki/API)[[edit](https://en.wikiversity.org/w/index.php?title=Emerging_Web_Technologies&action=edit&section=8)]

Application programming interface is a specification used to allow software components to communicate with each other.

### Biometric Authentication [Biometric](https://en.wikipedia.org/wiki/Biometric)[[edit](https://en.wikiversity.org/w/index.php?title=Emerging_Web_Technologies&action=edit&section=9)]

is the process of using a unique physical or behavioral traits as a method to confirm the identity and determine the access profile of a person. For example, face recognition through a webcam can be used in place of a password to unlock a computer.

### HTML5 Canvas[[edit](https://en.wikiversity.org/w/index.php?title=Emerging_Web_Technologies&action=edit&section=10)]

HTML 5 native video and audio support5 for all major codecs, set to replace flash...forever.

### Ruby On Rails[[edit](https://en.wikiversity.org/w/index.php?title=Emerging_Web_Technologies&action=edit&section=13)]

A Rails Framework place on top of the Ruby language [[1]](http://www.rubyonrails.org/) and is leading the web in portable web application this year.Ruby on rails is server-side scripting language that every Web Developer needs to know.

<http://fittextjs.com/>

<http://thegreatdiscontent.com/>

<https://www.filamentgroup.com/examples/responsive-images/>

## **1989**

**Tim Berners-Lee proposes the web**

Berners-Lee writes a paper entitled Information management: a proposal that envisages a system of interlinked documents that would be stored in a variety of locations, and contain non-hierarchical links to one another. These documents could be looked at using a 'browser' application, which would open the internet to potential mass use.

## **1991**

**The web is born**

The first web pages begin to appear. At first they are of limited general appeal, but the system has become a reality.

## **1993-94**

**Browsers develop**



A computer screen logged on to the Netscape Navigator internet search engine. Photograph: Garry Weaser

The web's transition to the mainstream is helped by the appearance of Mosaic, an intuitive, user-friendly browser, in 1993, and then a year later by Netscape Navigator, which attains an 80% share of web browser usage by 1996.

## **1996**

**Internet Explorer 3.0**

[Internet](https://www.theguardian.com/technology/internet) Explorer 3.0 is provided free of charge with Windows 95, a practice known as 'bundling' that later brings Microsoft to the attention of anti-monopoly bodies in the US and EU. With its massive market dominance and tendency to favour Microsoft's own applications such as Media Player, there comes a de facto influence on emerging web technologies; if it doesn't work on IE, it doesn't work for most users. The web is at risk of becoming a proprietary technology.

## **1997**

**Google**

Before Google, sites such as Yahoo! created searchable directories of websites, but Google's focus on technology and indexing gives the impression that it is a gateway to the whole web.

## **1998**

**Blog comments**

Blogging emerged in the mid-1990s but the launch of Open Diary in 1998 is a watershed because it is the first significant platform to encourage reader responses or comments on individual blogposts.

**Monica Lewinsky**



A picture of Monica Lewinsky and Bill Clinton taken on 17 November 1995. Photograph: AP

When the Drudge Report, a gossipy news blog, breaks the story on 17 January of then US president Bill Clinton's relationship with Monica Lewinsky, its headline is "Newsweek kills story on White House intern", reflecting the impotence of offline media to control the news agenda in an online world.

## **2001**

**Wikipedia**

Wikipedia is founded as a "multilingual, web-based, free-content encyclopedia", to which anyone can contribute, and which anyone can access. By 2014 the English-language version has about 4.5million entries.

## **2003**

**The great firewall**

[The Chinese government begins to pursue several initiatives](https://www.theguardian.com/media/2003/apr/24/chinathemedia.digitalmedia) to counter the potentially subversive influence of the open web, including this system of web filters that allows the authorities to block individual web pages, whole websites, or any page referencing a particular term.

## **2004**

**Facebook**



Facebook CEO Mark Zuckerberg. Photograph: Jonathan Ernst /Reuters

The site, which only becomes generally accessible in 2006, helps millions of people become more active online but is not on the open web as many of its pages, for good reasons, are only accessible to signed-in users.

## **2006**

**Google's Chinese censorship**

The search engine [compromises with the government](https://www.theguardian.com/technology/2006/jun/07/news.searchengines) in Beijing on the issue of censoring search results, in order to get access to the billion-strong Chinese market.

## **2007**

**iPhone**

The iPhone, like Facebook, is another innovation that makes many people more active online, but at the same time draws them away from the open web, in this case into Apple's app ecosystem. Apps are small applications that may use the internet, or even web protocols, but are not usually web browsers.

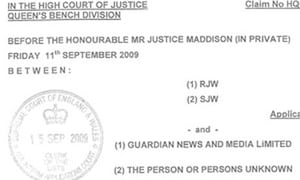
## **2008**

**Google Chrome browser and Chrome OS**

Google releases a suite of products including a freeware browser called Chrome, and an operating system that consists of the Chrome browser plus a set of web-based applications to replace traditional, locally installed software.

## **2009**

**Trafigura**



Trafigura's super-injunction against Guardian News and Media

The open web comes to the defence of old media in October, when [blogs and tweeters reveal that the oil company behind a gagging order against the Guardian is Trafigura](https://www.theguardian.com/world/2009/oct/16/trafigura-carter-ruck-the-guardian).

## **2010**

**WikiLeaks**

In April, WikiLeaks makes the first release from the cache of documents it has received from Bradley Manning [which includes video footage of a US army helicopter firing on civilians](https://www.theguardian.com/world/2010/apr/05/wikileaks-us-army-iraq-attack). Later that year, [the US embassy cables](https://www.theguardian.com/world/the-us-embassy-cables) make public a vast trove of diplomatic memos.

## **2011**

**Anti-WikiLeaks subpoenas and response**

The US government responds to the WikiLeaks revelations by issuing subpoenas to Twitter and Google to reveal what they know about users suspected of involvement. Twitter reacts by informing the users. Elsewhere, [a number of companies withdraw their services from WikiLeaks](https://www.theguardian.com/media/2010/dec/07/wikileaks-under-attack-definitive-timeline), including Visa, Paypal and Apple, which removes a WikiLeaks app from its store.

**The Arab spring**



A general view of Tahrir Square in Cairo, taken in November 2012. Photograph: Gianluigi Guercia/AFP/Getty Images

Particularly in Egypt and Tunisia, Twitter and Facebook play a key role in the spread of revolutions that shake the Arab world over the year.

## **2012**

Advertisement

**Sopa, Pipa, Acta**

[The Stop Online Piracy act](https://www.theguardian.com/technology/sopa) (Sopa) and the Protect Intellectual Property Act (Pipa) are two pieces of proposed US legislation that their proponents claim will protect copyright owners. Others say the effect will be web censorship. Wikipedia takes the threat so seriously that it goes on strike for a day. Both bills falter.

**Two Twitter trials**

The law around free expression on the web has been tested several times in the UK in recent years, with differing outcomes. In one case a trainee accountant is convicted of sending a menacing tweet about his local airport. The high court eventually overturns the sentence. The court also finds in favour of Lord McAlpine in a case against Sally Bercow, who had sent a tweet wrongly linking him to paedophile allegations.

## **2013**

**The NSA files**



A child holds a cut out of Edward Snowden. Photograph: Philippe Lopez/AFP

Classified documents leaked by Edward Snowden, a former National Security Agency contractor, reveals a number of mass-surveillance programmes undertaken by the NSA and Britain's surveillance agency, GCHQ.

## **2014**

**Cameron's 'opt-in' censorship**

[David Cameron pushes for default content filters operated by internet service providers](https://www.theguardian.com/technology/2014/jan/08/david-cameron-great-firewall), with users able to request that they be lifted, saying it will protect children from pornography. Campaigner Cory Doctorow says it will fail in this, but succeed in establishing "a regime of unaccountable censorship". The opt-in version goes live.

# Don’t Stop at ‘Good’; Become ‘Great’

<http://mashable.com/2017/10/05/google-event-2017-lineup/#KkF2fYvY0Pqb>

# Why doesn't Facebook and Twitter use a responsive web design?

Responsive web design is not the only way to deliver a mobile-friendly website, but it's the one that Google recommend us to implement[[1]](https://www.quora.com/Why-doesnt-Facebook-and-Twitter-use-a-responsive-web-design" \l "XKOzO) . And yet, it's not quite perfect. One of RWD limitation, back then, is how to make image responsive. Though there's new picture element[[2]](https://www.quora.com/Why-doesnt-Facebook-and-Twitter-use-a-responsive-web-design" \l "mVBcx) , there are browsers that don't support it yet [[3]](https://www.quora.com/Why-doesnt-Facebook-and-Twitter-use-a-responsive-web-design#LgFuv) .

Facebook and Twitter use different approach to handle their mobile browser visitor. They use separate URL[[4]](https://www.quora.com/Why-doesnt-Facebook-and-Twitter-use-a-responsive-web-design" \l "fLyzN) .

<https://m.facebook.com>

<https://mobile.twitter.com>

The benefits of this method are:

* Different code for mobile and desktop browser
* Serve different image for mobile browser (smaller one, minimize the bandwith for mobile phone)
* Isolate the development environment, so developer can optimize the code, the features to certain audience (mobile or desktop browser)

User Agent Plays key role in identifying from where the target is launching.

Will be redirected based on the request and browser user agent.

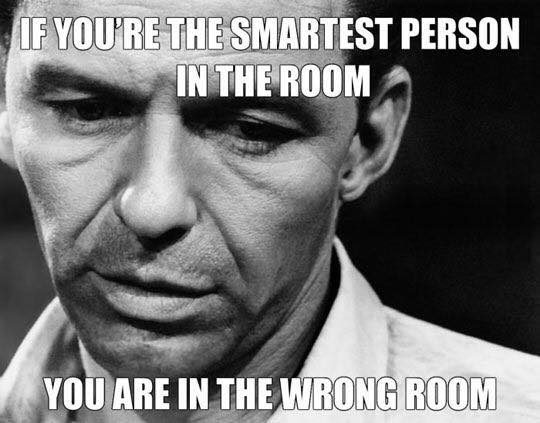
<https://www.digitalplatforms.co.za/interesting-facts-web-technology/>

**Emails and Spam Facts:**

• 60 billion emails are sent daily, 97% of which are spam.  
• Spam generates 33bn KWt-hours of energy every year, enough to power 2.4 million homes, producing 17 million tons of CO2.  
• 9 out of every 1,000 computers are infected with spam.  
• Spammer get 1 response to every 12 million emails they send (yet it still makes them a small profit).  
Social Media:  
• A “twillionaire” is a twitterer with a million or more followers.  
• There are some 1 billion computers in use.  
• There are some 2 billion TV sets in use.  
• Facebook has 500 million registered users… and still about 100 million less than QQ – a Chinese social media portal.  
• About 20% of the videos on YouTube are music related.  
• 24 hours of video viewing is uploaded every minute on YouTube.  
• People view 15 billion videos online every month.  
• On average, US onliners view 100 videos per month each.  
• Flickr hosts some 5 billion photographs, Facebook hosts more than 15 billion.

**Gadgets & Technology:**

• The first public cell phone call was made on April 3, 1973 by Martin Cooper.  
• The Motorola DynaTAC 8000X was the first cell phone sold in the US; launched on April 11, 1984, it was designed by Rudy Krolopp and weighed 2 pounds.  
• There are more than 4 billion cell phones in use. About 3 million cell phones are sold every day.  
• The first known cell phone virus, Cabir.A, appeared in 2004.  
• Since 2008, video games have outsold movie DVDs.  
Search Engines, Internet and Energy:  
• Amazon now sells more e-books than printed books.  
• About 1.8 billion people connect to the Internet, only 450 million of them speak English.  
• Google indexed it’s 1 trillionth unique URL on July 25, 2008. That is thought to be about 20% of all the pages on the Internet but a high percentage of the World Wide Web (the public Internet).  
• One Google search produces about 0.2g of CO2. But since you hardly get an answer from one search, a typical search session produces about the same amount of CO2 as does boiling a tea kettle.  
• Google handles about 1 billion search queries per day, releasing some 200 tons of CO2 per day.  
• The average US household uses 10.6 megawatt-hours (MWh) electricity per year.  
• Google uses an estimated 15 billion kWh of electricity per year, more than most countries. However, google generates a lot of their own power with their solar panels.  
Information Technology:  
• IBM celebrated 100 years in business in 2011 in honor of the formation of the core predecessor companies that would become International Business Machines under one combined umbrella in 1911.  
• n 1981, IBM started the PC revolution with the introduction of the IBM 5150, a compact personal computer that smoked mainframe processing and came at a price tag of under $1,600.  
• IBM invests $6 billion a year on research.  
• Microsoft was originally named Micro-Soft. They dropped the dash in 1976.  
• Microsoft total revenue in its 1st year: $16,000

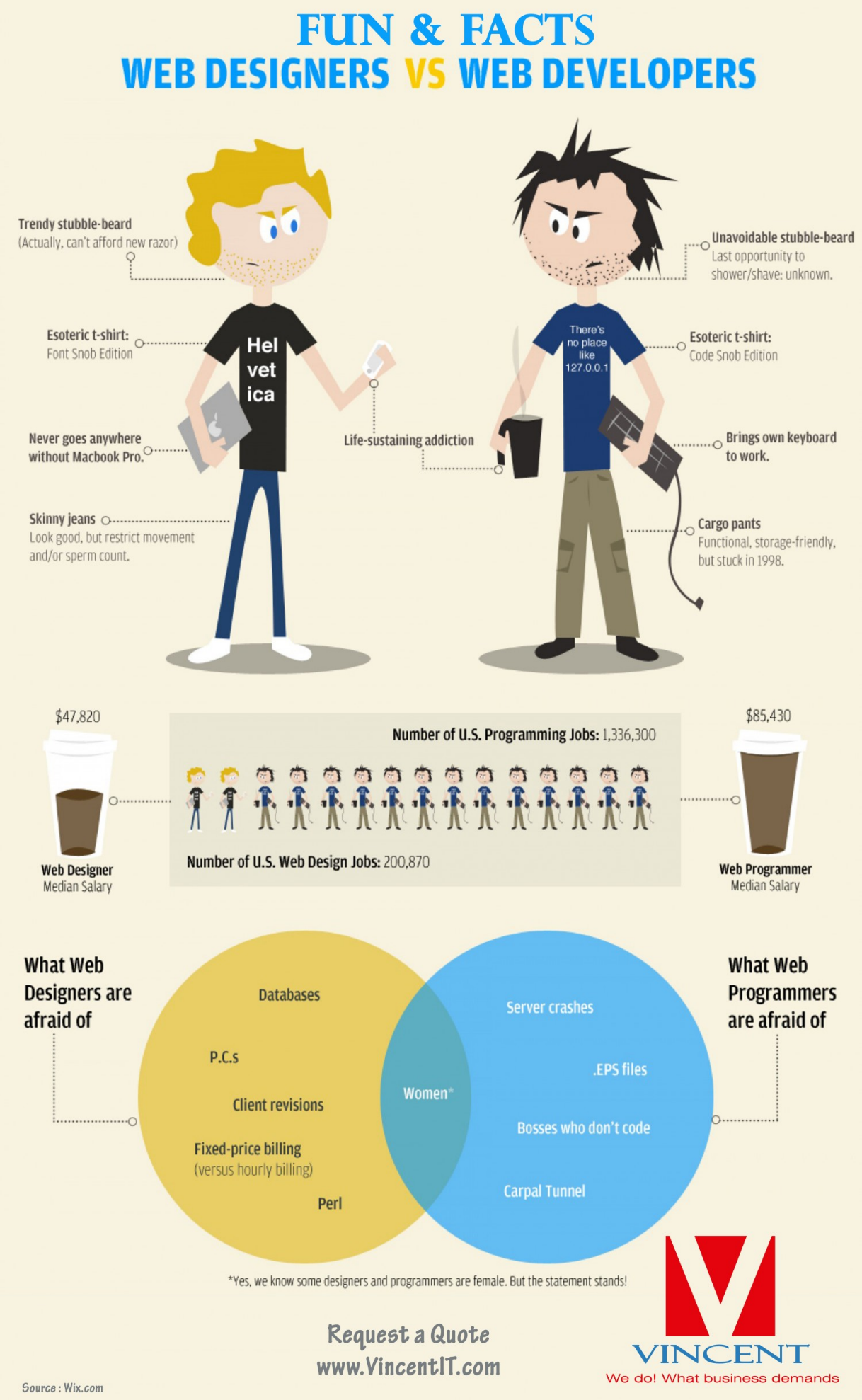


1. A red panda is an animal native to the Himalayas and southwestern China. The English word for red panda is ‘Firefox’ which is where the browser gets its name from – so the Firefox logo is actually a red panda, not a fox!
2. Dragon Skin Body Armour is a newer invention that can withstand a grenade explosion at point-blank range.
3. The very first Apple logo featured Sir Isaac Newton sitting underneath a tree, with an apple about to hit his head.
4. Google rents out goats from a company called California Grazing to help cut down the amount of weeds and brush at Google HQ!
5. The founder of Microsoft, Bill Gates, was a college drop out.
6. Bill Gates house was designed using a Macintosh Computer.
7. The word robot comes from the Czech “*robota*“. This translates into forced labour, or work.
8. CD’s, or Compact discs read from the inside to the outside edge, which is the reverse of how records work.
9. You cannot reverse a [Bitcoin](https://www.thefactsite.com/2015/06/bitcoin-facts.html) transaction, or be forced to pay.
10. The average computer user blinks seven times a minute, the normal rate is twenty times per minute.
11. The first ever VCR, which was made in 1956, was the size of a piano!
12. On an average work day, a typist’s fingers travel about 12.6 miles.
13. The first alarm clock could only ring at 4am.
14. The 30th of November is known as “[Computer Security Day](https://www.thefactsite.com/2011/11/computer-security-day-30th-november.html)“.
15. On [1st April](https://www.thefactsite.com/2012/03/april-fools-day.html) 2005, NASA pulled a prank telling the world that they had found water on Mars.
16. The radio took 38 years to reach a market audience of 50 million.
17. The television took 13 years to reach a market audience of 50 million.
18. The iPod took 3 years to reach a market audience of 50 million.
19. If you was to have your picture taken by the very first camera, you’d need to sit still for 8 hours.
20. A dentist named Alfred Southwick invented the electric chair.
21. The creators of the PNG file format wanted it to be pronounced as ‘ping’.
22. Skype is banned from the public in China.

### Computer Facts

1. Alaska is the only state in America that can be typed on one row of a traditional English QWERTY keyboard.
2. The Apple II had a hard drive of only 5 megabytes when it was launched.
3. In general, people tend to read as much as 10% slower from a screen than from paper.
4. Ubuntu is one of the more popular distributions of Linux. The word Ubuntu comes from an African word meaning “I am because of you”.
5. The Name “Macintosh” Was Inspired by an Apple with the name “mcintosh”.
6. In 1932 Professor August Dvorak created the [Dvorak keyboard](https://www.thefactsite.com/2009/01/things-you-didnt-know-about-keyboards.html), which was made to be superior to the standard QWERTY keyboard.
7. Doug Engelbart created the very first computer mouse from wood in 1964.
8. Christopher Sholes invented the QWERTY keyboard in 1868.
9. The [Apple Lisa](https://www.thefactsite.com/2013/01/amazing-fact-about-apple-lisa.html) was the first commercial computer with a graphical user interface (GUI) and a mouse.
10. The Apple Lisa was released in June 1983.
11. In 1822, Charles Babbage created the first computer.
12. 1024 Gigabytes is equal to 1 Terabyte.
13. 1024 Terabytes is equal to 1 [Petabyte](https://www.thefactsite.com/2010/05/visualizing-petabyte.html).
14. 1 Petabyte can hold 13.3 years of HD-TV video.
15. 86% of people try to plug their USB devices upside down.

<https://www.thefactsite.com/2013/02/top-100-technology-facts.html>



<https://github.com/showcases/front-end-javascript-frameworks>

<https://www.w3schools.com/angular/angular_intro.asp>

## AngularJS History

AngularJS version 1.0 was released in 2012.

Miško Hevery, a Google employee, started to work with AngularJS in 2009.

The idea turned out very well, and the project is now officially supported by Google.

<https://angularjs.org/>

MVC

### Components[[edit](https://en.wikipedia.org/w/index.php?title=Model%E2%80%93view%E2%80%93controller&action=edit&section=2)]

* The *model* is the central component of the pattern. It expresses the application's behavior in terms of the [problem domain](https://en.wikipedia.org/wiki/Problem_domain), independent of the user interface.[[6]](https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller#cite_note-6) It directly manages the data, logic and rules of the application.
* A *view* can be any output representation of information, such as a chart or a diagram. Multiple views of the same information are possible, such as a bar chart for management and a tabular view for accountants.
* The third part, the *controller*, accepts input and converts it to commands for the model or view.[[7]](https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller#cite_note-7)

### Interactions[[edit](https://en.wikipedia.org/w/index.php?title=Model%E2%80%93view%E2%80%93controller&action=edit&section=3)]

In addition to dividing the application into three kinds of components, the model–view–controller design defines the interactions between them.[[8]](https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller#cite_note-posa-8)

* A *model* stores data that is retrieved according to commands from the controller and displayed in the view.
* A *view* generates new output to the user based on changes in the model.
* A *controller* can send commands to the model to update the model's state (e.g., editing a document). It can also send commands to its associated view to change the view's presentation of the model (e.g., scrolling through a document, movement of document)

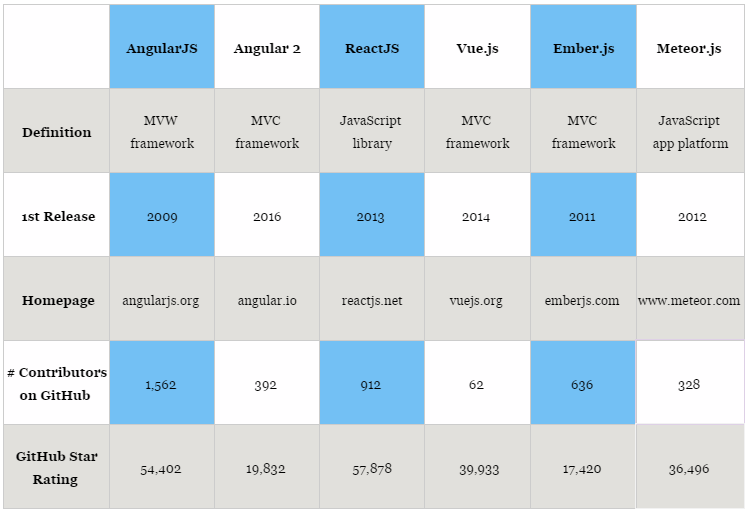
### Advantages[[edit](https://en.wikipedia.org/w/index.php?title=Model%E2%80%93view%E2%80%93controller&action=edit&section=10)]

* *Simultaneous development* – Multiple developers can work simultaneously on the model, controller and views.
* [*High cohesion*](https://en.wikipedia.org/wiki/Cohesion_(computer_science)) – MVC enables logical grouping of related actions on a controller together. The views for a specific model are also grouped together.
* [*Low coupling*](https://en.wikipedia.org/wiki/Loose_coupling) – The very nature of the MVC framework is such that there is low coupling among models, views or controllers
* *Ease of modification* – Because of the separation of responsibilities, future development or modification is easier
* *Multiple views for a model* – Models can have multiple views

### Disadvantages[[edit](https://en.wikipedia.org/w/index.php?title=Model%E2%80%93view%E2%80%93controller&action=edit&section=11)]

* *Code navigability* – The framework navigation can be complex because it introduces new layers of abstraction and requires users to adapt to the decomposition criteria of MVC.
* *Multi-artifact consistency* – Decomposing a feature into three artifacts causes scattering. Thus, requiring developers to maintain the consistency of multiple representations at once.
* *Pronounced learning curve* – Knowledge on multiple technologies becomes the norm. Developers using MVC need to be skilled in multiple technologies.

MVVM



NODE JS

Node.js is a JavaScript runtime built on Chrome's V8 JavaScript engine. Node.js uses an event-driven, non-blocking I/O model that makes it lightweight and efficient. The Node.js package ecosystem, [npm](https://www.npmjs.com/), is the largest ecosystem of open source libraries in the world.

The Node.js project is supported by the [Node.js Foundation](https://nodejs.org/en/foundation/). Contributions, policies, and releases are managed under an [open governance model](https://github.com/nodejs/node/blob/master/GOVERNANCE.md).

**Node.js** is an [open-source](https://en.wikipedia.org/wiki/Open-source_software), [cross-platform](https://en.wikipedia.org/wiki/Cross-platform) [JavaScript](https://en.wikipedia.org/wiki/JavaScript) [run-time environment](https://en.wikipedia.org/wiki/Runtime_system) for executing JavaScript code [server-side](https://en.wikipedia.org/wiki/Server-side). Historically, JavaScript was used primarily for [client-side scripting](https://en.wikipedia.org/wiki/Client-side_scripting), in which scripts written in JavaScript are embedded in a webpage's HTML, to be run client-side by a JavaScript engine in the user's web browser. Node.js enables JavaScript to be used for [server-side scripting](https://en.wikipedia.org/wiki/Server-side_scripting), and runs scripts server-side to produce [dynamic web page](https://en.wikipedia.org/wiki/Dynamic_web_page) content *before* the page is sent to the user's web browser. Consequently, Node.js has become one of the foundational elements of the "JavaScript everywhere" paradigm,[[5]](https://en.wikipedia.org/wiki/Node.js" \l "cite_note-5) allowing [web application](https://en.wikipedia.org/wiki/Web_application)development to unify around a single programming language, rather than rely on a different language for writing server side scripts.

Though .js is the conventional [filename extension](https://en.wikipedia.org/wiki/Filename_extension) for JavaScript code, the name "Node.js" does not refer to a particular file in this context and is merely the name of the product. Node.js has an [event-driven architecture](https://en.wikipedia.org/wiki/Event-driven_architecture) capable of [asynchronous I/O](https://en.wikipedia.org/wiki/Asynchronous_I/O). These design choices aim to optimize [throughput](https://en.wikipedia.org/wiki/Throughput) and [scalability](https://en.wikipedia.org/wiki/Scalability) in [Web applications](https://en.wikipedia.org/wiki/Web_application) with many input/output operations, as well as for [real-time Web](https://en.wikipedia.org/wiki/Real-time_Web)applications (e.g., [real-time communication](https://en.wikipedia.org/wiki/Real-time_communication) programs and [browser games](https://en.wikipedia.org/wiki/Browser_game)).[[6]](https://en.wikipedia.org/wiki/Node.js#cite_note-readwrite-6)

The Node.js [distributed development](https://en.wikipedia.org/wiki/Distributed_development) project, governed by the Node.js Foundation,[[7]](https://en.wikipedia.org/wiki/Node.js" \l "cite_note-7) is facilitated by the [Linux Foundation](https://en.wikipedia.org/wiki/Linux_Foundation)'s Collaborative Projects program.[[8]](https://en.wikipedia.org/wiki/Node.js#cite_note-8)

Node.js was originally written by Ryan Dahl in 2009,[[23]](https://en.wikipedia.org/wiki/Node.js" \l "cite_note-training.com-23) about thirteen years after the introduction of the first server-side JavaScript environment, Netscape's LiveWire Pro Web.[[24]](https://en.wikipedia.org/wiki/Node.js#cite_note-24) The initial release supported only Linux and Mac OS X. Its development and maintenance was led by Dahl and later sponsored by [Joyent](https://en.wikipedia.org/wiki/Joyent).[[25]](https://en.wikipedia.org/wiki/Node.js#cite_note-Google_Groups_post_by_Ryan_Dahl_about_Joyent-25)

WhatsApp

Part of the trick is that the company builds its service using a programming language called **Erlang**. Though not all that popular across the wider coding community, Erlang is particularly well suited to juggling communications from a huge number of users, and it lets engineers deploy new code on the fly

JavaScript was traditionally the language of the web browser, performing computations directly on a user’s machine. This is referred to as “client-side” processing. With the advent of Node.js, JavaScript has become a compelling “server-side” language as well, which was traditionally the domain of languages like Java, Python and PHP.

[**Isomorphic application**](http://isomorphic.net/javascript)

In web development, an [isomorphic application](http://isomorphic.net/javascript) is one whose code (in this case, JavaScript) can run both in the server and the client.

## **Benefits of going isomorphic**

* There is less code, as it is shared by both the client and the server.

## **Risks of isomorphic applications**

### DEBUGGING IS TRICKIER

### BYE BYE JQUERY (But not so easy because I struggled to write a POST request using XMLHttpRequest instead of $.post().)

### AVOID EXPOSING SENSITIVE DATA



### **Comparison of Angular || Sencha**

|  |  |  |
| --- | --- | --- |
| **Type** | **Angular JS** | **Sencha** |
| Architecture | MVW(MVC, MVVW) | MVC & MVVW |
| Testing | Supports Automated Testing,  own test runner called Karma which facilitates unit testing | no in-built support for automated testing but it is possible with the use of external tools |
|  | For end-to-end testing, Protractor can be used. Protractor runs tests against the application running in a real browser, interacting with it as a user would. | It does not come with any testing framework – However, other third-party JavaScript testing frameworks such as Siesta, Jasmine or Mocha can be used for testing |
| Programming | Supports Object-oriented, Functional and Event-driven programming paradigms | Supports Object-oriented and Event-driven programming paradigms |
| Cross Browser Compatibility | relies on jqLite/jQuery that provides cross-browser compatibility  Supports all types of browsers. Directives helps in handling the same | It offers complete cross browser compatibility |
| Mobile Applications | Cross platform native mobile applications or Hybrid apps development is possible with Trigger.io, Cordova or Phonegap integration | Cross platform native mobile applications or Hybrid apps development. is possible with Sencha Touch and Apache Cordova or Phonegap integration |
| Mobile Sites Development | developing mobile sites, one can use AngularJS responsive modules such as angular-responsive or angular-deckgrid, UI Bootstrap, AngularJS responsive directives, angular-gestures or the ngTouch library | For developing mobile sites, Sencha touch is used |
| Coding | features such as directives, filters, and automatic data bindings, the developers don’t need to write hundreds of lines of code for simple features, thus making the code more manageable | Rich Suite and Theme-able components helps in developing code in very few lines. |
| Source | open source web application framework | Licensed |
| Owner | Google and a community of passionate developers | Sencha |
| Suitable for | Highly used for Single Page Applications | Rich UI applications.  Single page applications.  With techniques like AJAX, DHTML and DOM scripting, it helps in building highly interactive web applications |
| Drawbacks | Light Weight / Faster | Heavy / bit slow Compared to Angular |
| Charts | D3 charts with D3 integration | Stand Alone charts without any additional plugins |
| Two way Data binding | Supports with directive | Supports |

<http://www.angularjshub.com/code/examples/basics/02_TwoWayDataBinding_HTML/example-section-container.php?url=index.demo.php>

<http://examples.sencha.com/extjs/5.0.0/examples/kitchensink/#binding-two-way>

#### Sencha Ext JS vs AngularJS

If you’re wondering how Ext JS compares to Angular JS, here is a quick overview of the differences between the two:



<https://codepen.io/donovanh/pen/vEjywy>

<http://alfredservice.com/>

**MVC**  
  
1)MVC consists of three layers Model, View, and Controller.   
2) MVC is a compound pattern   
3) It uses a Front Controller pattern that processes Web application requests through a single controller. This enables you to design an application that supports a rich routing infrastructure.   
4)It does not use view state or server-based forms. This makes the MVC framework ideal for developers who want full control over the behavior of an application.   
5)It provides better support for test-driven development (TDD).   
6)It works well for Web applications that are supported by large teams of developers and for Web designers who need a high degree of control over the application behavior.   
  
**MVC Steps**   
  
Step 1: Incoming request directed to Controller.   
Step 2: Controller processes request and forms a data Model.   
Step 3: Model is passed to View.   
Step 4: View transforms Model into appropriate output format.   
Step 5: Response is rendered.   
Example:-   
Please refer below link   
<http://www.codeproject.com/Articles/79577/A-Simple-Tutorial-on-Developing-ASP-NET-Applicatio>   
  
**MVP**  
  
1)1)MVP consists of three layers Model, View, and Presenter.   
2)In MVP, View and Model are more loosely coupled, providing a clearer separation of concerns.   
3)MVP, View is responsible for delegating the user input to the Presenter.   
4)MVP, Presenter and View should have a 1-1 relation, with each View having a reference to its Presenter through the interface.   
5)MVP, view binds to the Model directly through data binding.   
6)In MVP, unit testing is easier, as View knows Presenter through an interface which can easily be mocked.   
  
Example:-   
Please refer this link   
<http://www.c-sharpcorner.com/uploadfile/john_charles/model-view-presenter-mvp-design-pattern-and-data-binding/>   
  
**MVVM**  
  
1)MVVM pattern is a one of the best solutions to handle such problems for WPF and Silverlight application.   
2)When you use MVVM pattern for WPF, Silverlight the view wouldn't have the typical event handlers that's so common in UI code.   
3)MVVM provides a clear separation between the UI and application logic.   
4))The MVVM pattern includes three key parts:   
  
1)Model (Business rule, data access, model classes)   
2)View (User interface (XAML))   
3)ViewModel (Agent or middle man between view and model)   
Example:-   
Please refer this link   
<http://msdn.microsoft.com/en-us/library/gg405484%28v=pandp.40%29.aspx>

Ruby Sample Code

# Hello World Program in Ruby

puts "Hello World!";

s = "Hi there. How are you?"

print s.length, " [" + s + "]\n"

# Selecting a character in a string gives an integer ascii code.

print s[4], "\n"

printf("%c\n", s[4])

# The [n,l] substring gives the starting position and length. The [n..m]

# form gives a range of positions, inclusive.

print "[" + s[4,4] + "] [" + s[6..15] + "]\n"

print "Wow " \* 3, "\n"

print s.index("there"), " ", s.index("How"), " ", s.index("bogus"), "\n"

print s.reverse, "\n"

<https://www.tutorialspoint.com/execute_ruby_online.php>

WEBRTC

<https://webrtc.github.io/samples/src/content/getusermedia/gum/>