VOLUME 17 | ISSUE 11 | NOVEMBER 2022

digit FastTrack

YOUR HANDY GUIDE TO EVERYDAY TECHNOLOGY

TO CODING IN 2022

< JAVASCRIPT > | < PYTHON > | < HTML >
< CSS > | < JAVA > | < SQL > | < NOSQL > | < C# > | < RUST >
< PERL > | < GO > | < WHAT'S MORE >

FREE WITH DIGIT NOVEMBER 2022

Fed up of this $#!7:

#EPICFAIL U MAD? ROTFL GET A LIFE
ROFL CAPS LOCK NO SH!T LAME
STFU PWNED LOLZ
#BIATCH :-O *FACEPALM* D'OH!
#LOSER IDC 4RLZZZZZ?
HA HA HA HA ROFLMAOL ID10T
:-D #DORK WTF LOL
DAFUQ O RLY? LMAO W/E
JELLY? #R-TARD S.M.F.W.A.S.
YSOSRS? SHADDAP (>.<) #D-BAG

Come home to digit geek

Technology | Science | Geek Culture | Gaming | All things Geek

geek.digit.in

e-mag | Community Forum | e-DVDs | Geek articles and more...

FAST TRACK to

CODING IN 2022

powered by

digit

CHAPTERS

CODING IN 2022
NOVEMBER 2022

CREDITS
The people behind this book

EDITORIAL
Editor-in-Chief
Robert Sovereign-Smith

Managing Editor
Mithun Mohandas

Dy. Featues Edtior & Copy-editing
Satvik Pandey

DESIGN
Sr. Art Director
Anil VK

Associate Art Director
Baiju NV

digit.in

November 2022
Free with Digit. If you have paid to buy this Fast Track from any source other than 9.9 Group Pvt. Ltd. (Formerly known as Nine Dot Nine Mediaworx Private Limited), please write to editor@digit.in with details

Custom publishing
If you want us to create a customised Fast Track for you in order to demystify technology for your community, employees or students contact editor@digit.in

e-mail

VOLUME 17 | ISSUE 11 | NOVEMBER 2022
digit
FastTrack
YOUR HANDY GUIDE TO EVERYDAY TECHNOLOGY
TO
CODING
IN 2022
< JAVASCRIPT > | < PYTHON > | < HTML >
< CSS > | < JAVA > | < SQL > | < NOSQL > | < C# > | < RUST >
< PERL > | < GO > | < WHAT'S MORE >
FREE WITH DIGIT NOVEMBER 2022

COVER DESIGN: BAIJU NV

Welcome to the world of coding!

We will be taking you on a voyage through the seven seas of code...

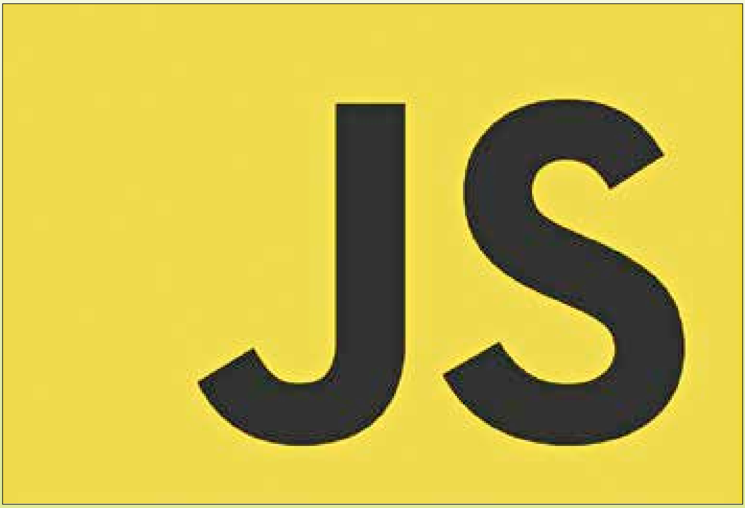
As you're aware, you're about to embark on an exciting journey that will ultimately arm you with the skills needed to change the world as we know it. Programming languages today are used for a variety of applications, from facilitating machine learning, transforming industries, analysing and collecting Big Data, creating immersive video games, and developing software and applications that change the way consumers behave.

According to the StackOverflow Survey 2022, there has been a 60-70% year-on-year increase in the number of students who want to learn coding online. While developers over the age of 45 mostly relied on books and traditional education to learn about this field, younger developers are turning to digital assets such as online courses, open source projects, websites that host challenges, and so on to improve their skill. So, where does that leave you? And where does this book fit into the picture?

This FastTrack to Coding could possibly be your introduction to the world of coding or be a book after reading which, you are able to take your coding game one step ahead. We'll be covering the top 11 programming languages that most developers use today, how you can use them, how you can get started, and what the latest industry trends corresponding to each language are. By the end of this book, you'll be able to succinctly differentiate between these top programming languages to the point where you can identify which language to start your journey with.

Good luck!

CHAPTER #01



JavaScript

Scripting away!

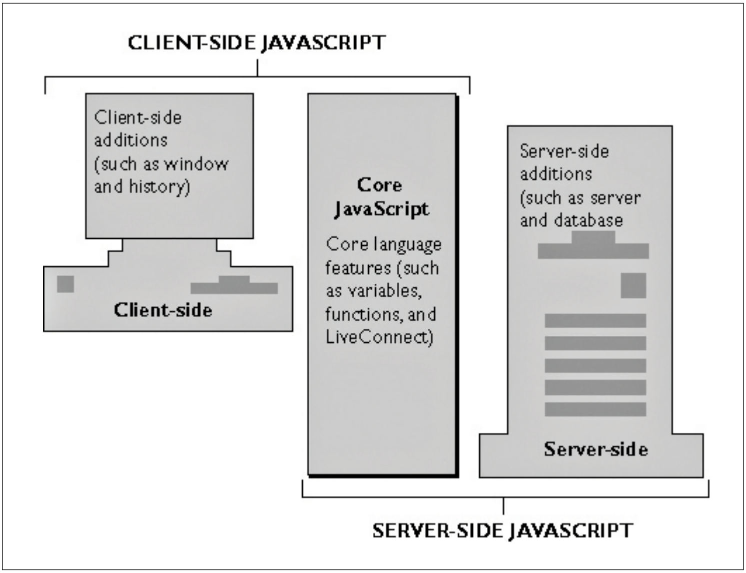
As one of the most popular programming languages in the world, JavaScript is an important weapon in every programmer's arsenal. It is classified as a dynamic computer programming language that is quite lightweight. JavaScript is typically used as a part of web pages where the implementations help with client-side scripts. These allow the website to interact with the user, thereby offering dynamic pages. It is also important to note that this is an interpreted programming language that has object-oriented capabilities.

Did you know that JavaScript was originally called LiveScript? Netscape changed its name to JavaScript, and it is believed that this was because of the hype of Java within the industry. JavaScript was launched as a part of Netscape 2.0 in 1995. The language was embedded in many different web browsers such as Netscape, Internet Explorer, and more.

What is Client-Side Javascript?

The most common version of JavaScript is known as Client-Side Javascript. It offers a broad range of advantages over traditional CGI server-side scripts, such as validating email addresses in online forms before submitting the data. JavaScript can also be used to trap different user-initiated events such as link navigation, button clicks, and other implicit or explicit actions.

When developing a website using Client-Side JavaScript, it is important to note that the script itself should be included in or referenced by an HTML document. This will enable the code to be interpreted by the browser. What this means is that the page does not have to be a static HTML, and it can include many interactive programs and dynamically create HTML content.


CLIENT-SIDE JAVASCRIPT
Client-side additions (such as window and history)
Client-side
Core JavaScript
Core language features (such as variables, functions, and LiveConnect)
Server-side additions (such as server and database
Server-side
SERVER-SIDE JAVASCRIPT

What is Javascript used for?

JavaScript is mainly used to build websites and web applications.The following are some practical applications of this language:

1. Web Development

JavaScript is a highly popular scripting language that is used to create web pages. This language lets developers create interactive and dynamic web page that can carry out complex actions. It also lets users simply load content in a document without having to reload the page. Many websites use this language for validation. It also lets them support external applications, such as PDF documents, and widgets.

2. Web Applications

Different types of JavaScript frameworks can be used to develop and build robust web applications. For instance, Google Maps is powered by JavaScript. The language interacts with the browser without communicating with the servers. React Native, React and Angular are some JavaScript front-end frameworks that help build web apps.

3. Presentations

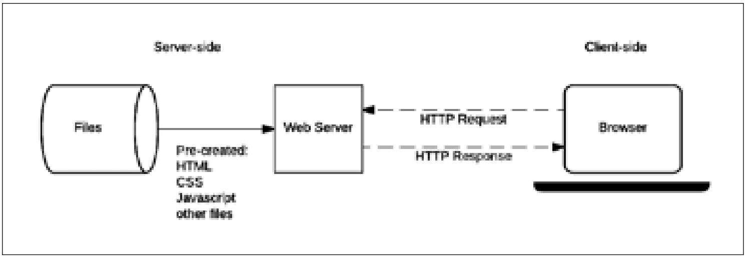
JavaScript can be used to create interactive presentations and publish them as websites. To do so, developers can use RevealJs and BespokeJs libraries. They can help you generate web-based slide decks using HTML.The biggest advantage is that even if a developer is not fully comfortable with a programming language, they can easily use Javascript to build presentations as websites.

4. Server Applications

JavaScript is used to write server-side software by using Node.js. Which is an open-source runtime environment. Using this, you can write, test and debug code in order to enable fast and scalable network applications. JavaScript allows you to generate content. It can also help you manage HTTP requests.

5. Web Servers

By using Node.js, developers can create a web server using Javascript. Node.js is being event-driven. This means that it can move to the next call without waiting for the previous call's response. The servers can then transfer chunks of data without buffering. It is important to note that HTTP module uses the createServer () method in order to create a server.



6. Games

Javascript can also help you create games on the internet. Developers typically use Javascript as well as HTML5. The EaselJS library is filled with rich graphics for games. HTML5, on the other hand, provides total access to the web and it doesn't need additional plugins like Flash. Complex types of browser games like Tower Building, CrossCode, HexGL are based on JavaScript and HTML5.

7. Art

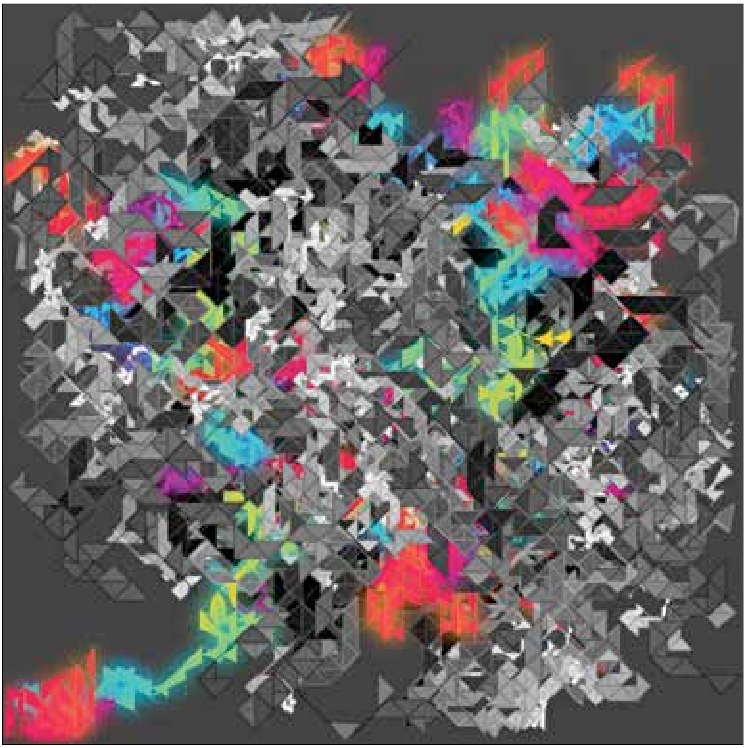
One of the recently integrated features of HTML5 in JavaScript is the canvas element. This allows you to create 2D and 3D graphics easily on a web page. Many people use this for creating digital art projects on the browser.

8. Smartwatch Apps

Pebble.js is a JavaScript framework that has been created by Pebble. It allows developers to make applications for Pebble watches by using JavaScript.

9. Mobile Apps

One of the most commonly explored applications of JavaScript is creating apps for non-web contexts. This means you can create apps for things that are not on the Internet. JavaScript frameworks can be used to facilitate mobile app development irrespective of platforms. You can create apps for iOS, Android, and Windows.



Sample JS art

10. Drones

By using the Node.js ecosystem, you can control numerous small robots, IoT devices, and creative maker projects.

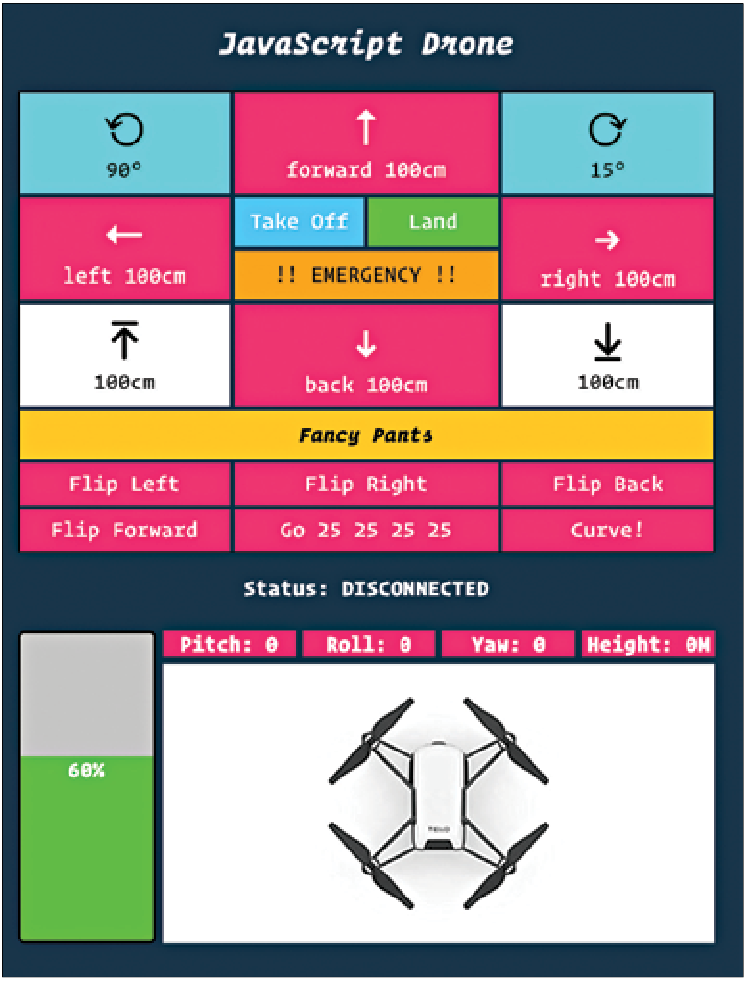
Latest JavaScript Trends

Some of the top Javascript trends you should keep in might are:

1. Front-end Frameworks

React.js, Vue.js, and Angular.js are the most popular JS frameworks.However, Svelte is catching up and doubled its user base last year. Many developers think that it will surpass Vue as it is a light and easy alternative to React.js and Angular.js.

As developers believe that Angular.js is hard to learn, fewer developers are pursuing it. It is used to develop large, complex projects.



Status: DISCONNECTED

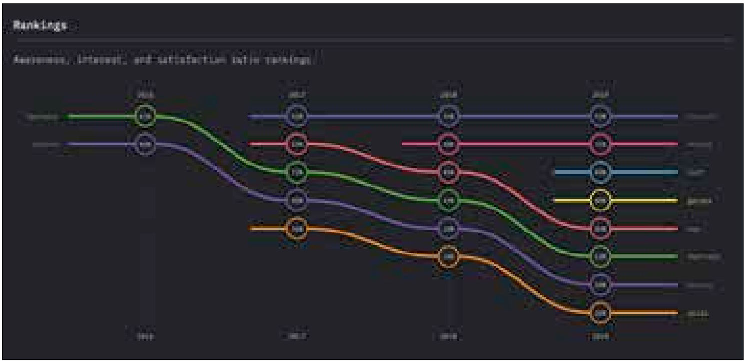
Pitch: 0 Roll: 0 Yaw: 0 Height: 0M

React.js is considered to be the top JS framework. It is constantly improving features and this ensures that developers and clients want to use it as their front-end framework. Features like hooks, component-based servers, and concurrent mode make it ideal for use in 2022.

2. Back-end Frameworks

Express.js considered a top back-end framework for the server-side of JavaScript development. However, new trends such as the rise of JAMstack- like site generators (Next.js and Gatsby) and headless CMS (Strapi) are important to keep track of.

Another interesting trend is the rise in popularity of Fastify as it provides a faster alternative for Express.js. It also has great features like hooks, decorators, and encapsulated plugins. They can help with facilitating the reusability and decoupling of code.



3. Web components as building blocks

Web components can be very versatile and used with plain JavaScript or any other frameworks. Some of the capabilities to take note of are:

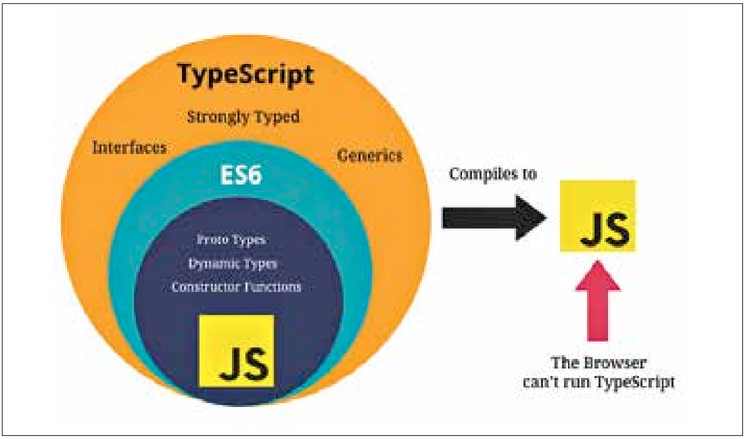
- Custom Elements – This can create new autonomous HTML tags that have fully-customizable behaviors.
- Shadow DOM – This helps with encapsulating DOM subtrees to web elements.
- ES Module – This helps determine JS modules and reuse them in other JS documents.
- HTML Template – This helps developers create markup fragments that are instantiated during runtime to prevent the load time from slowing down.

4. Typescript

TypeScript (TS) is a superset of JavaScript that has become one of the foremost picks as a programming language. This is because it allows for optional types that are not allowed in Javascript. It uses JavaScript syntax and semantics along with offering additional features that can help developers improve their efficiency and productivity.

Some of the main reasons why Typescript has become so popular include:

- Compile-time bug detection
- Static typing
- Large-scale solution support
- Simple instructions that developers can easily learn



How to get started?

As JavaScript has a host of applications, there are many different ways you can start using this language, based on what you want to create. The following are three basic ways that let you start easily:

1. Use the Console Tab in your Web Browser

As most popular web browsers have built-in JavaScript engines, you can run the language on your browser of choice.

To run JavaScript on a browser, you must:

- Open the desired browser (for instance, Google Chrome)
- Right-click an empty space to open developer tools. Select Inspect. OR, you can press F12 on your keyboard.
- Click the Console tab.
- Writer your code and click Run to execute it.

2. Use Node.js

Node.js is a framework that allows you to write code using Javascript. To use Nodejs, you must:

- Install an updated version of Node.js
- Install a text editor/IDE (such as Visual Studio Code)
- Create a new file
- Write your code
- Save your file with a .js extension.
- Open the terminal/command prompt
- Navigate to the file location
- Type node hello.js
- Hit Enter.

3. Use it by creating web pages

You an also start using Javascript by creating web pages. To do so, you must:

- Open VS Code
- Go to the File menu and select New File
- Write your code
- Save the file with a .html extension
- Create a similar file in node with a .js extension.
- Add the following script to your code: <script scr="main.js"></script>
- Open the HTML file using your web browser

What has changed?

ECMAScript 2022 has brought new changes to Javascript, making the language even more versatile.The .at() method helps with negative indexing and is compatible with Array, String, or TypedArray.

Basic Code

Try running these codes to practice

JavaScript Statements

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Statements</h2>

<p>A <b>JavaScript program</b> is a list of <b>statements</
b> to be executed by a computer.</p>
```

```
<p id=”demo”></p>

<script>
var x, y, z;  // Declare 3 variables
x = 5;     // Assign the value 5 to x
y = 6;     // Assign the value 6 to y
z = x + y;  // Assign the sum of x and y to z

document.getElementById(“demo”).innerHTML =
“The value of z is “ + z + “.”;
</script>

</body>
</html>
```

JavaScript Numbers

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Numbers</h2>

<p>Number can be written with or without decimals.</p>

<p id=”demo”></p>

<script>
document.getElementById(“demo”).innerHTML = 10.50;
</script>

</body>
</html>
```

CHAPTER #02

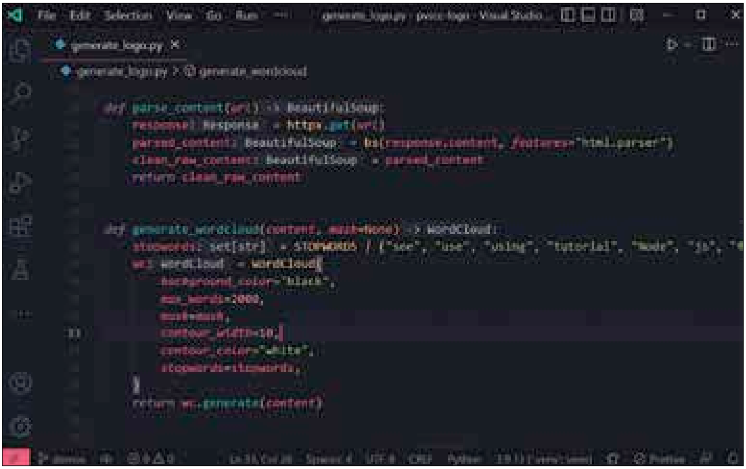
Python

Not the snake

Python is an interpreted, general-purpose high-level programming language that is mainly used for automation, website development, and data analytics.Python's initial development was led by Guido van Rossum all the way back in the late 198Os. Now, Python is managed by the Python Software Foundation.

Python is a general-purpose language. This means that is versatile. Moreover, it can be used to program different functions. As it is classified as interpreted language, it precludes compiling code before execution. Additionally, as it is a high-level programming language, the languagecan simply abstract details from code. Note that Python focuses so much on abstraction that the code itself be understood and implemented by most novice programmers.

When compared to compiled languages like C and C++, the Python code is generally short, however, it executes programs slower. The fact that it is


File Edit Selection View Go Run ···
generate_logo.py - pyssc-logo - Visual Studio ...
generate_logo.py ×
generate_logo.py > generate_wordcloud
Python

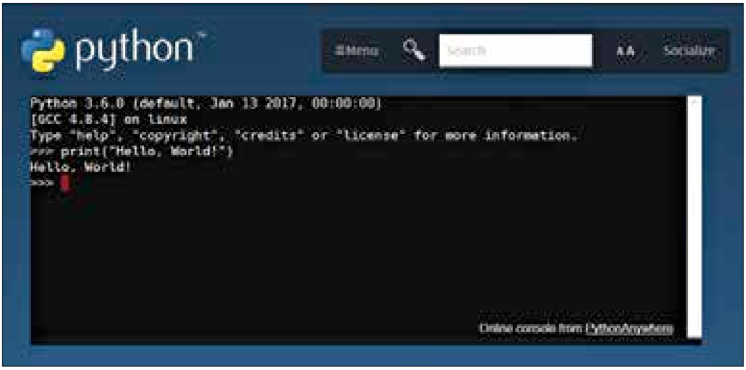
```
def parse_content(url) -> BeautifulSoup:
    response: Response = https.get(url)
    parsed_content: BeautifulSoup = bs(response.content, features="html.parser")
    clean_raw_content: BeautifulSoup = parsed_content
    return clean_raw_content


def generate_wordcloud(content, mask=None) -> WordCloud:
    stopwords: set[str] = STOPWORDS | {"see", "use", "using", "tutorial", "Node", "js",
    wc: WordCloud = WordCloud(
        background_color="black",
        max_words=2000,
        mask=mask,
        contour_width=10,
        contour_color="white",
        stopwords=stopwords,
    )
    return wc.generate(content)
```

user-friendly makes it a popular language for most developers who are working with machine learning algorithms using low-code no-code (LCNC) software applications.

Python has an easy syntax. The language is also known for its large community that actively contributes to its increasing software modules and libraries. To sum things up:

- **Python is Interpreted** – This languanged is processed at runtime by the interpreter. A developer does not need to compile the program before executing it.
- **Python is Interactive** – The developer can sit at a Python prompt and interact with the interpreter directly while writing the programs.



```
Python 3.6.0 (default, Jan 13 2017, 00:00:00)
[GCC 4.8.4] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> print("Hello, World!")
Hello, World!
>>>
```

- **Python is Object-Oriented** – This language supports Object-Oriented style. Tis is a technique of programming that encapsulates code within objects.
- **Python is a Beginner's Language** – It is great language for the beginner-level programmers. This language also supports the development of a broad range of applications from simple text processing to WWW browsers to games.

Some of the salient features of Python are –

- **Easy-to-learn** – The language only has a few keywords. It has a simple structure, and a clearly defined syntax. Anyone can, therefore, pick up the language quickly.
- **Easy-to-read** – This language is more clearly defined. As it is considered 'visible to the eyes', it is easier to spot mistakes.
- **Easy-to-maintain** – The source code is very easy to maintain.
- **Broad standard library** – The bulk of the library is very portable. It is also cross-platform compatible on OSs like UNIX, Windows, and Macintosh.
- **Interactive Mode** – The language supports interactive mode. As a result, it allows interactive testing and debugging code snippets.
- **Portable** – The language can run on a broad spectrum of hardware platforms. What's important is that it has the same interface on all platforms.
- **Extendable** – Developers can add low-level modules to the Python interpreter. These modules help developers further customise their tools to be more efficient.
- **Databases** – This language has a great interface on all major commercial databases.
- **GUI Programming** – The language supports GUI applications. These applications can easily be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.
- **Scalable** – This language offers a better structure and support for large programs when compared with shell scripting

What is Python Used For?

Python has a variety of uses across all industries, and can come in handy for various job roles. These include:

1. Web development

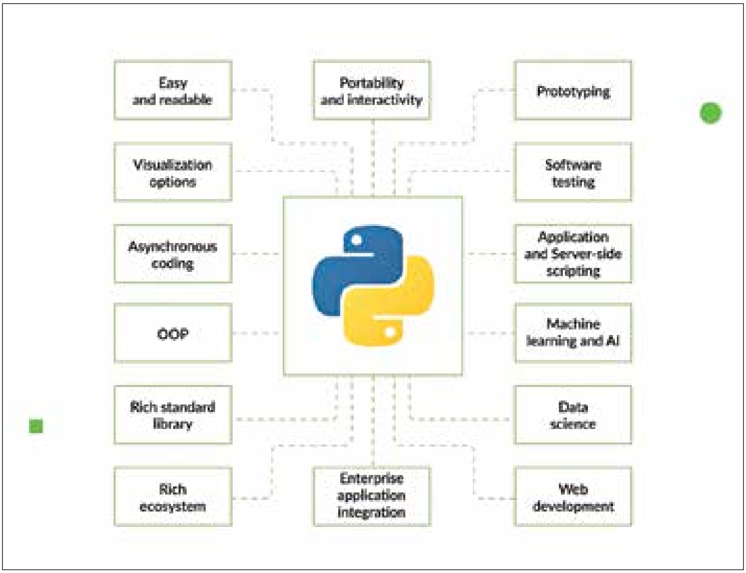
Before the launch of Django — which is basically a development framework that has been designed to support all projects created with Python — this language was more widely used by software programmers as opposed to web developers. But, Django's utility ended up transforming Python into a great web development too. Did you know that Pinterest and Instagram use this language for their back-end development?

2. Data analytics and data science

Python is the chosen language when it comes to data analytics (and tasks including accessing, organizing and visualizing data. Python-based tools such as SciPy and NumPy are widely used for drawing numerical insights from user data. Such tools are easy to use and scalable.

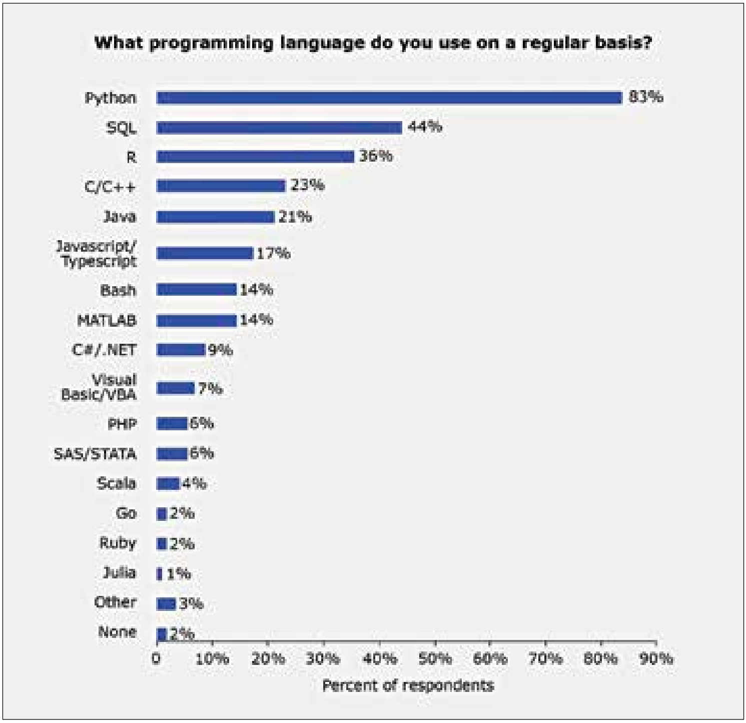
3. Artificial intelligence (AI) and machine learning (ML)

Certain Python libraries were basically designed to identify and analyse huge data sets. Such algorithmic capabilities are very important in the AI and machine learning industries. As a result of this, Python is a standard language when it comes to extrapolating and analysing data related to AI.



4. Fintech and finance

Did you know that Python is one of the most popular languages in the



fintech industry? Aspiring developers and programmers should learn Python as this industry is projected to grow further thanks to the rise in mobile banking and cryptocurrency.

Latest Industry Trends

As an aspiring programmer, you must stay abreast of the following industry trends that emerged in 2022.

1. Artificial Intelligence

One of the most significant trends for Python in 2022 is that this language is used widely in Artificial Intelligence. The language is very handy when it comes to developing different types of AI systems with heavy data requirements. This is mainly because AI needs a tool that allows programmers to work with structured and unstructured data. There are plenty of libraries for AI. . Some of the most popular libraries included Keras, OpenCV, NLTKas they help with functions such as natural language processing, scientific computing, computer vision, and much more.

2. Framework Upgrades

Python developers tend to be in demand as they can help match the pace of change at various companies. The frameworks in Python are always rapidly changing changing technologies, and this helps developers stay abreast with the latest requirements. Some of the top Python frameworks include CherryPy, TurboGears, and Django. These frameworks are expected to reduce time and make coding easier in the coming years.

3. Web Application Development

Python is an immensely powerful programming language with many frameworks that help enterprises build high-quality applications. As a result, Python development will continue to be in-demand. Do keep in mind that Python is considered a leading programming language for prototypes that can enablecompanies to gain a first-mover advantage. The language also allows companies to stay relevant in their niche by keeping up with new changes.

4. Academic Growth

Did you know that Python is the most sought-after and taught programming language? While there are already a lot of colleges and university teaching this language, you can expect to see a rise in the number of educations institutes that offer such courses. It is interesting to note that many companies are going teh extra mile by training existing programmers and developers in Python.

5. Cloud Computing

Python is considered to be very suitable for cloud computing. This is mainly because need for data protection in the cloud is increasing. There are many different platforms that use Python in their stack to manage cloud computing servers. .Many cloud computing providers like AWS, Google Cloud, DigitalOcean use Python. This is because this language offers dynamic development capabilities.

How to get started?

Python is available on almost every platform you can think of, including Linux and Mac OS X. Here are some ways to set up your Python environment:

Local Environment Setup

Open a terminal window and type "python" to see if it has already been installed (and to identify the version installed). Python can be found in:

- Unix (Solaris, Linux, FreeBSD, AIX, HP/UX, SunOS, IRIX, etc.)
- Win 9x/NT/2000
- Macintosh (Intel, PPC, 68K)
- OS/2
- DOS (multiple versions)
- PalmOS
- Nokia mobile phones
- Windows CE
- Acorn/RISC OS
- BeOS
- Amiga
- VMS/OpenVMS
- QNX
- VxWorks
- Psion
- Python has also been ported to the Java *and .NET* virtual machines

Getting Python

You can head to the official website of Python (*https://www.python.org/)* for the most up-to-date and current source code, documentation, binaries, news, etc.

You can then download Python documentation. The documentation is available in various formats like HTML, PDF, and PostScript.

Installing Python

You just need to download the binary code applicable for your platform and install Python.

However, If the binary code for your platform is not available, all you need to do is get a C compiler to compile the source code manually. This can be beneficial. Compiling the source code gives you more flexibility in terms of the different features can add-on during your installation.

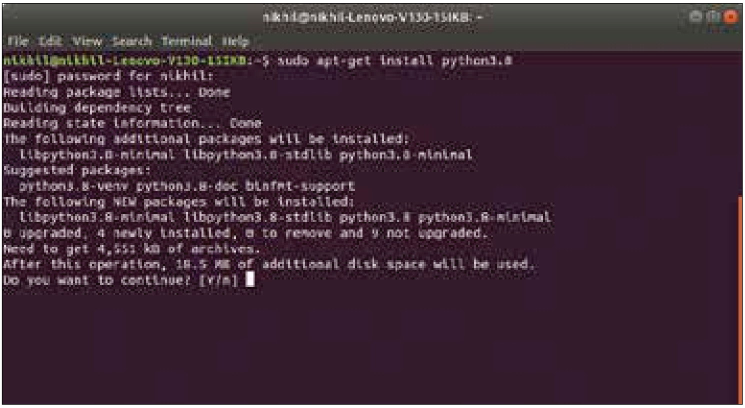
Here is a quick overview of installing Python on various platforms –

Unix and Linux Installation

To nstall Python on Unix/Linux machine, you must

- Open a Web browser and go to *https://www.python.org/downloads/.*

- Follow the link to download zipped source code available for Unix/Linux.
- Download and extract files.
- Edit the Modules/Setup file if you want to customize some options.
- run ./configure script
- make
- make install



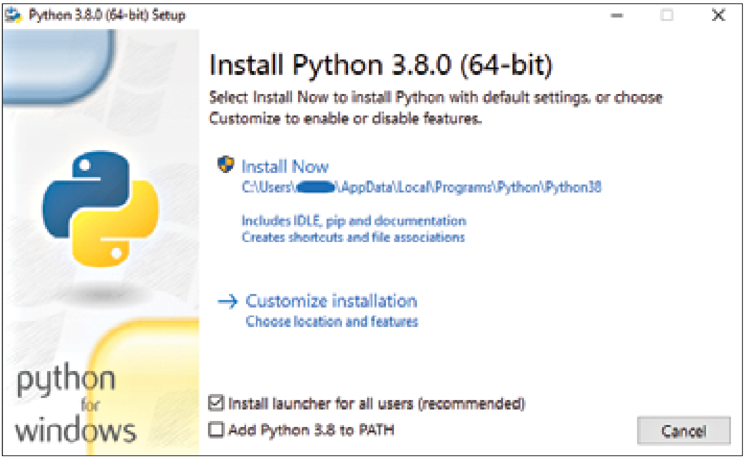
nikhil@nikhil-Lenovo-V130-15IKB: ~
File Edit View Search Terminal Help

```
nikhil@nikhil-Lenovo-V130-15IKB:~$ sudo apt-get install python3.8
[sudo] password for nikhil:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libpython3.8-minimal libpython3.8-stdlib python3.8-minimal
Suggested packages:
  python3.8-venv python3.8-doc binfmt-support
The following NEW packages will be installed:
  libpython3.8-minimal libpython3.8-stdlib python3.8 python3.8-minimal
0 upgraded, 4 newly installed, 0 to remove and 9 not upgraded.
Need to get 4,551 kB of archives.
After this operation, 18.5 MB of additional disk space will be used.
Do you want to continue? [Y/n] 
```

Windows Installation

To install Python on Windows machine, you must

- Open a Web browser and go to *https://www.python.org/downloads/.*



Install Python 3.8.0 (64-bit)

Select Install Now to install Python with default settings, or choose Customize to enable or disable features.

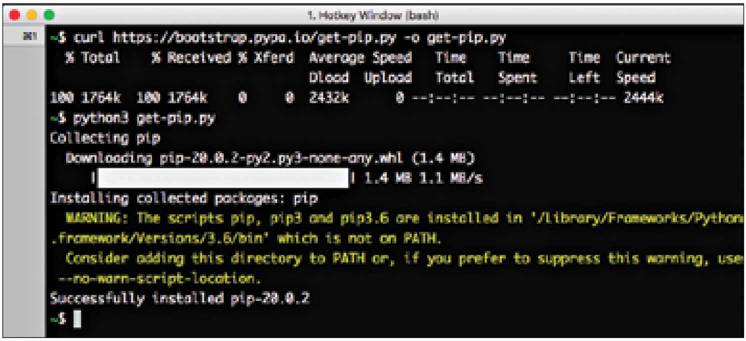
Install Now
C:\Users\[redacted]\AppData\Local\Programs\Python\Python38

Includes IDLE, pip and documentation
Creates shortcuts and file associations

→ Customize installation
Choose location and features

☑ Install launcher for all users (recommended)
□ Add Python 3.8 to PATH

- Follow the link for the Windows installer python-XYZ.msi file where XYZ is the version you need to install.
- To use this installer python-XYZ.msi, the Windows system must support Microsoft Installer 2.0. Save the installer file to your local machine and then run it to find out if your machine supports MSI.
- Run the downloaded file. This brings up the Python install wizard. Accept the default settings, wait until the install is finished, and will be done.



```
~$ curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 1764k  100 1764k    0     0  2432k      0 --:--:-- --:--:-- --:--:-- 2444k
~$ python3 get-pip.py
Collecting pip
  Downloading pip-20.0.2-py2.py3-none-any.whl (1.4 MB)
     |████████████████████████████████| 1.4 MB 1.1 MB/s
Installing collected packages: pip
  WARNING: The scripts pip, pip3 and pip3.6 are installed in '/Library/Frameworks/Python
.framework/Versions/3.6/bin' which is not on PATH.
  Consider adding this directory to PATH or, if you prefer to suppress this warning, use
 --no-warn-script-location.
Successfully installed pip-20.0.2
~$ █
```

Macintosh Installation

While many Macs come with Python installed, the environment may be several years out of date. Head to *http://www.python.org/download/mac/* to access instructions that help with getting the current version along with additional tools to support development on the Mac. MacPython is available for older Mac OS's before Mac OS X 10.3 (released in 2003).

Setting up PATH

As programs and other executable files are typically available in many directories, operating systems offer a search path that lists the directories to make it easier to find the files. .

The path is stored in an environment variable. This is a named string that is maintained by the operating system. This variable contains information that is available to the command shell and other programs.

The path variable is named PATH in Unix or Path in Windows (note that Unix is case sensitive; however, Windows is not).

When it comes to the Mac OS, the installer handles the path details. To open the Python interpreter from any specific directory, you must add the Python directory to your path.

Setting path at Unix/Linux

To add the Python directory to the path for a particular session in Unix –

- In the csh shell – type setenv PATH "$PATH:/usr/local/bin/python" and press Enter.
- In the bash shell (Linux) – type export PATH="$PATH:/usr/local/bin/python" and press Enter.
- In the sh or ksh shell – type PATH="$PATH:/usr/local/bin/python" and press Enter.
- Note – /usr/local/bin/python is the path of the Python directory

Setting path at Windows

To add the Python directory to the path for a particular session in Windows –

At the command prompt – type path %path%;C:\Python and press Enter.

Note – C:\Python is the path of the Python directory

Changes in Python

Python 3.0 has a list of new features that enhance the efficiency of developers and programmers. These are:

New syntax features:

- Structural Pattern Matching: Specification
- Structural Pattern Matching: Motivation and Rationale
- Structural Pattern Matching: Tutorial
- Parenthesised context managers are now officially allowed.

New features in the standard library:

- Add Optional Length-Checking To zip.

Interpreter improvements:

- Precise line numbers for debugging and other tools.

New typing features:

- Allow writing union types as X | Y
- Explicit Type Aliases
- Parameter Specification Variables

Important deprecations, removals or restrictions:

- Require OpenSSL 1.1.1 or newer
- Deprecate distutils module.

- Deprecate and prepare for the removal of the wstr member in PyUnicodeObject.
- Remove Py_UNICODE encoder APIs
- Add optional EncodingWarning

Basic Code

Hello World

```
print("Hello, World!")
```

Adding two numbers

```
# This program adds two numbers
num1 = 1.5
num2 = 6.3
# Add two numbers
sum = num1 + num2

# Display the sum
print('The sum of {0} and {1} is {2}'.format(num1, num2, sum))
```

Create a variable

```
x = 5
y = "John"
print(x)
print(y)
```

Output both text and variable

```
x = "Python"
y = "is"
z = "awesome"
print(x, y, z)
```

Add a variable to another variable

```
x = "Python "
y = "is "
z = "awesome"
print(x + y + z)
```

CHAPTER #03

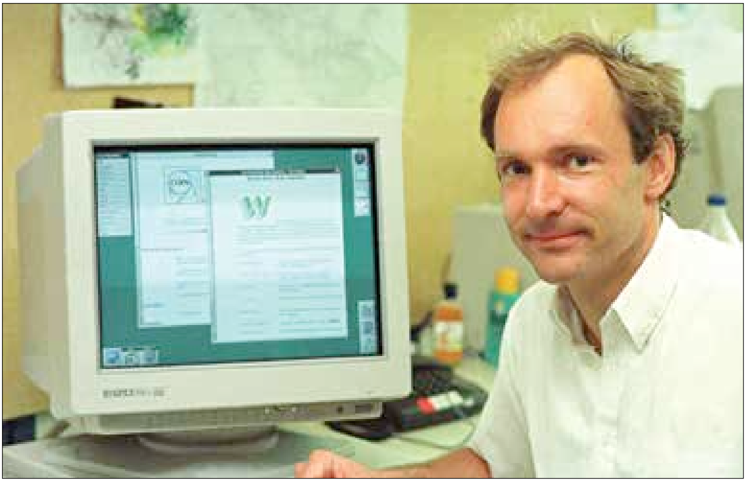
HTML

Literally everywhere

HTML is an acronym that stands for “Hypertext Markup Language,”. It is a language that is used for creating, structuring and designing websites. The term “hypertext”in the full form is important to note as it refers to HTML’s capabilities to link out to relevant content. Since it is a markup language, the tags used in HTML help with creating the overall structure and design of the webpage. They are also used to define the content elements within it.

When it comes to the most basic functionality — that is, prior to any extra styling and dynamic elements that you may choose to apply — HTML tags are used to inform your web browser the items that must be displayed on the screen along with the right order. When you are coding for the web, you must enclose or wrap various aspects of the content to make sure that it displays exactly where, when and how you desire. Many basic HTML tags can help set the font size as well as different formatting choices that impact how content is perceived, like bold or italics.

Though the type of content, dynamism and design that is typically used on the web has transformed significantly in the last few decades, HTML has still held its place as the fundamental framework used to power web design. Even the most dynamic websites that are generated by more advanced scripting languages (such as Javascript or PHP) depend on HTML, at least to some extent. As a result of this, learning to code in HTML is still essential for modern web developers. In today's times, HTML knowledge is essential for a developer who aspires to build, improve or manage websites — and this holds true for developers, marketers and even small-time business owners.



Tim Berners Lee

HTML was invented by Tim Berners-Lee in 1993. However, the language was not properly used for coding until 1999. This is the year when HTML 4.01 turned into the official standard for web development. The language was updated into HTML5 in 2012, and this version came with expanded capabilities.

It is interesting to note that modern HTML codebases are now shorter than what they used to be. A huge number of interactive and design elements that programmers rely on to code manually can now be handled with the help of CSS (Cascading Style Sheets) or JavaScript.

What is HTML used for?

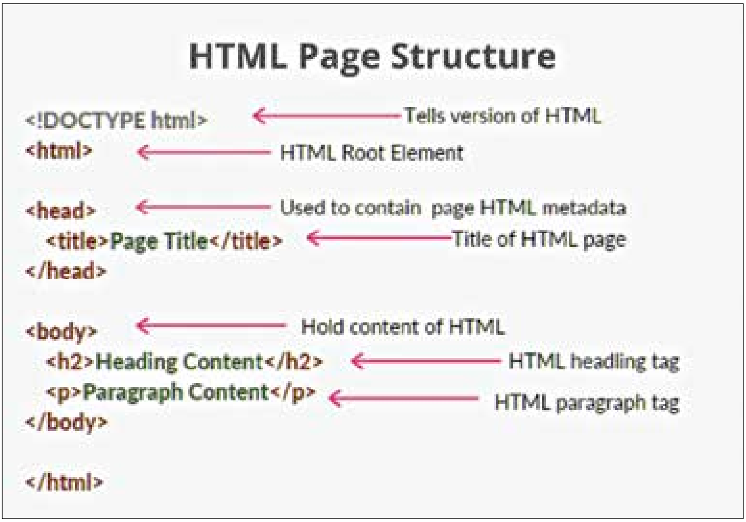
1. Structuring web pages

HTML is commonly used to structure web pages. By using certain tags and elements, developers can categorise the headings, paragraphs, and other contents. How is works is simple. Most browsers have their own built-in stylesheet that is used to visually differentiate different elements. For example, content that is surrounded with <h1> tags will have its own look in terms of bold fonts and line spacing. Similarly, if there is text that is surrounded by <h2> tags, it will look different from the content surrounded by <h1>, and so on. It is important to note that when these tags are not used, the text on the webpage looks the same.

2. Navigating the internet

Did you know that its much easier to navigate the internet thanks to HTML. Without HTML, you would have to manually enter the URL of every single website you want to open. This is basically because of the use of anchor tags. These tags help us to link pages to and from each other by using a specific attribute known as the href attribute. For example:

- `<a href=”https://www.google.com”>Google</a>`



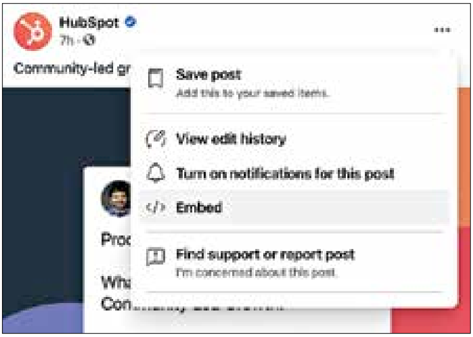
```
<!DOCTYPE html>
<html>

<head>
  <title>Page Title</title>
</head>

<body>
  <h2>Heading Content</h2>
  <p>Paragraph Content</p>
</body>

</html>
```

The code snippet that has been shown above creates a hyperlink to Google. You can also use hyperlinks to link to a specific section of a webpage.


HubSpot
7h ·
Community-led gr
Save post
Add this to your saved items.
View edit history
Turn on notifications for this post
Embed
Find support or report post
I'm concerned about this post.

3. Embedding images and videos

Most webpages today have images that break apart large chunks of content. This is achieved by using HTML. This language does not only allow you to not only embed images on the desired page but also change their width, height, and position. You can also change the way they're rendered. Before HTML, web developers would have to use Flash in order to embed any video or images. Now, you can just use the <video> tag. Additionally, there are many HTML tags that let you adjust thumbnails, video controls, autoplay, timestamps, and more.

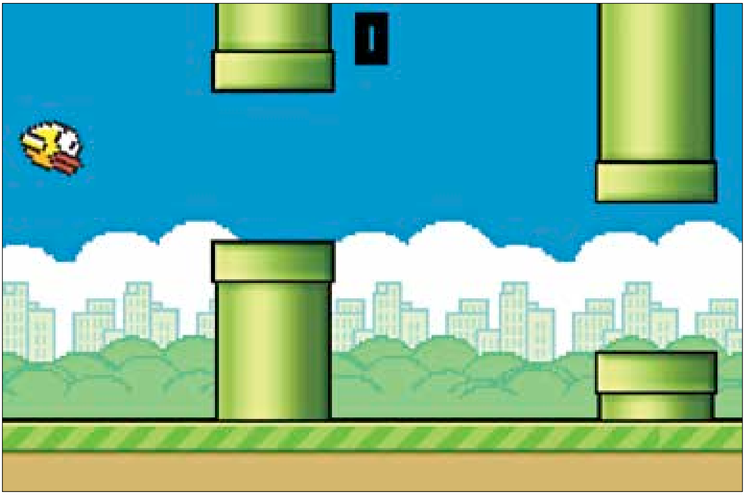
4. Improving client-side data storage and offline capabilities

With HTML5, we have also seen better offline capabilities. Many websites rely on cookies to store data about users. This data can be retrieved later if they revisit the site. Such functionalities work well for small data like authentication tokens and usernames. However, HTML5's addition of localStorage and IndexDB has been quite helpful as it allows browsers to store more complex data.

Additionally, you can reference application cache and cache manifest files. These files help with letting you browse a website when you are offline as it retains the data on the machine. The biggest benefit of this is that it lets your web app continue functioning even when the connection drops.

5. Game development

Even though developers cannot create video games just with HTML, a certain attribute known as the <canvas> element helps build video games in the browser while using CSS and JavaScript. Before HTML, a developer could only do this with the help of Flash or Silverlight. With modern HTML, developers can create 2D and 3D games that can be played in the browser.



Remember this?

6. Interacting with native APIs

Did you know that HTML can interact with your operating system and not just with the web browser? Features like this make it easy to drag files onto a web page so that you can upload them, make a video display on full-screen, and more.

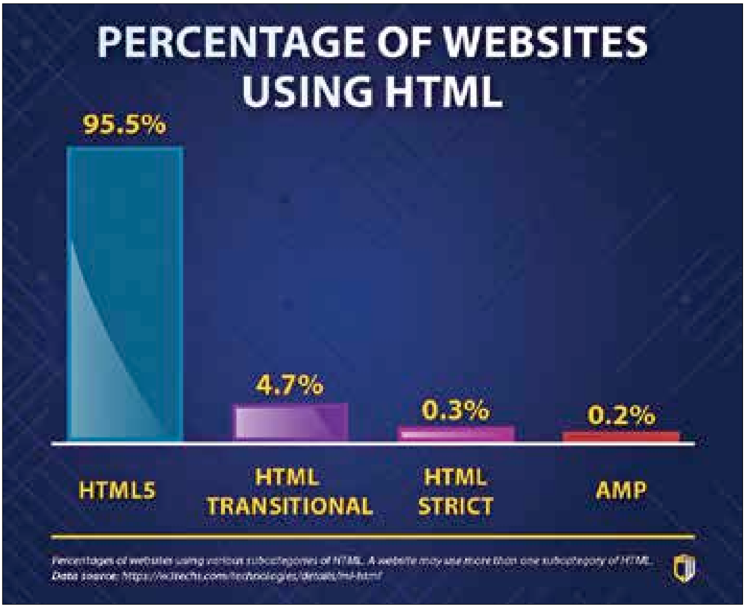
Latest HTML Trends

1. 3-D Illustration

Designers now want to add realism along with organic shapes. These are being made with the purpose of changing the boundaries between physical and digital worlds. 3-D web models will be more and more popular in the coming years. The idea will be to enhance the customer experience.

2. Machine Learning in the Browser

Thanks to libraries like Brain.js and TensorFlow.js, running automatic learning in the browser is now a reality. Due to this, designers have the ability to run a pre-trained model directly or to build a model in the browser. Manual learning is easy to access. It works easily on all devices and is also considered quite interactive as it leverages its access to the camera, microphone,, GPS, mouse inputs and more.



3. Algorithmic Design

Designing is now automated. What this means is that traditional web designs are considered outdated. Algorithms are becoming a huge trend as they allow people to palm off the grunt work, to spend effort in big-picture needs. design workflows that can combine traditional and custom tools, specific to your project, thus allowing you to work much faster and have better results.

4. Websites that Respond to the User's Situation

Another trend that is soon going to emerge is that we will have websites that respond accurately to the user input, device, or environment, irrespective of whether it is to enhance usability or accessibility. An example of a functionality that corresponds with this trend is the Ambient Light Sensor, which is supported in Google Chrome. This will be increasingly implemented. These ambient light sensors can gauge the light in the room and then dim the brightness of the screen reduce eye strain. Developers can use APIs to create highly compelling stories on the web, and this can alter based on the light level of the room. Developers can also enhance the readability of text in low light by using this.

5. Privacy Protection

Protecting a user's privacy is incredibly important as it helps brands build trust. Privacy by Design is a trend to watch out for. There are many promising tools in the works that can help you learn more about the way your projects functions, like Commento or Fathom Analytics. Developers are also turning to privacy-focused tools such as Firefox, Keybase, etc.

6. Speculative Design

Speculative design refers to imaginative research. The concept represents an opportunity to build fresh signals and narratives. This can offer a new space where different digital practitioners can easily collaborate with artists, architects and so on. As a result of this, one may end up creating prototypes for new experiences. These prototypes can then spark very insightful conversations about web development and design.

How to get started?

You can get started with HTML quite easily as you don't have to install a separate program to do so. If you have a Windows OS, you will need to open Notepad. However, if you have a Mac OS, you will need to open TextEdit. To get started, you must:

1. Open the appropriate text editor based on your OS
2. Start writing the code
3. Save the file with a .html extension.
4. Open the file in your brower.
5. The page you have created will be displayed.

What has changed in HTML?

HTML5, which is the latest version of HTML, marks the biggest change in the world of HTML. This version has introduced many new features that designers should take note of. Some of these include:

New tags:

HTML5 lets you use new tags such as:

- <article>
- <footer>
- <header>
- <video>

New upgrades:

- Deprecated elements like strike, font, and center have been dropped

- There is more flexible parsing and compatibility with the improved parsing rules
- There are many new elements like nav, section, video, time, progress, meter, canvas and side
- There are new input attributes such as email, dates and time, and URL
- There are new attributes such as charset, async and ping
- There are new APIs that help with offline caching, along with drag-and-drop support
- You can use Silverlight or Flash for vector graphics
- The language now supports MathML to display mathematical notations
- The JS Web Worker API allows JavaScript to run in the background
- You can apply global attributes such as tabindex, repeat and id to all elements

Basic Code & Tags

Try writing this basic code and using tags to practise!

My first heading

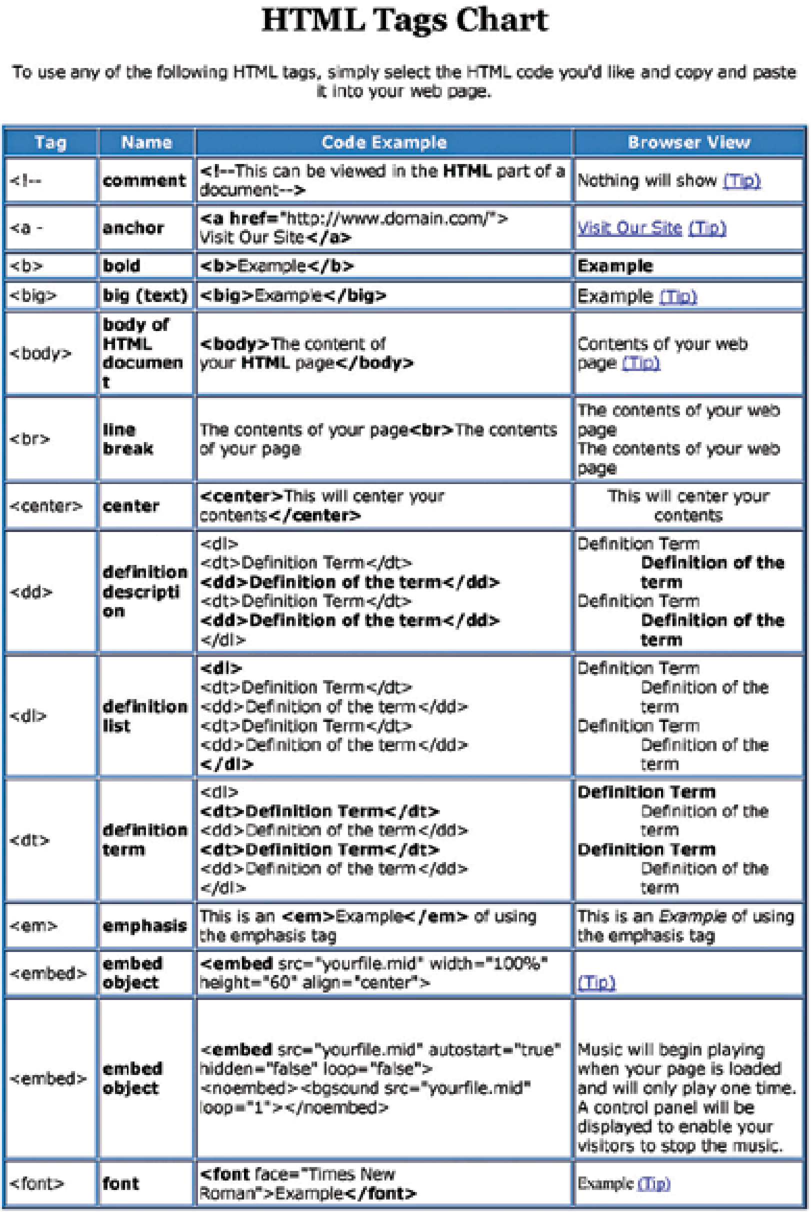
```
<html>
<body>

<h1>My First Heading</h1>

<p>My first paragraph.</p>
</body> </html>
```

HTML headings

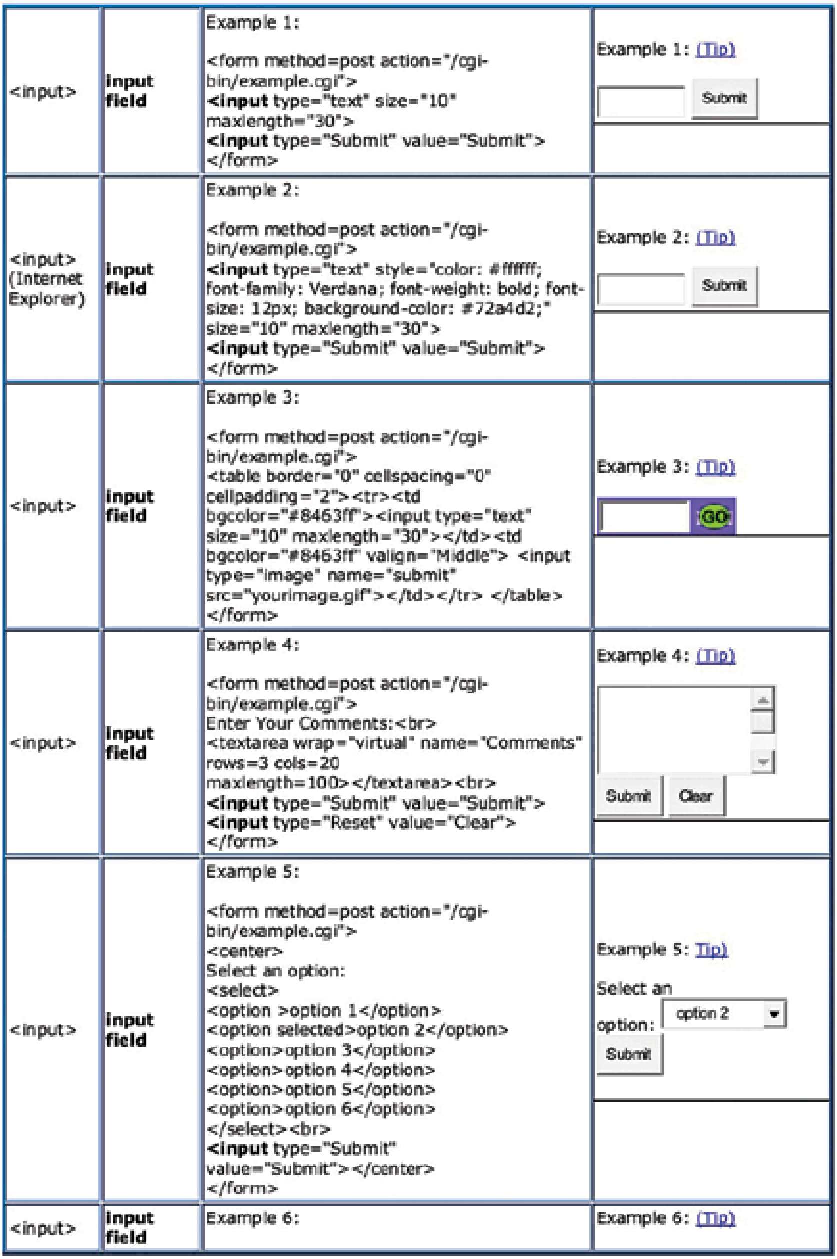
```
<!DOCTYPE html>
<html>
<body>
<h1>This is heading 1</h1>
<h2>This is heading 2</h2>
<h3>This is heading 3</h3>
<h4>This is heading 4</h4>
<h5>This is heading 5</h5>
<h6>This is heading 6</h6>
</body>
</html>
```



HTML Tags Chart

To use any of the following HTML tags, simply select the HTML code you'd like and copy and paste it into your web page.

Tag	Name	Code Example	Browser View
<!--	**comment**	**<!--**This can be viewed in the **HTML** part of a document**-->**	Nothing will show (Tip)
<a -	**anchor**	**<a href=**"http://www.domain.com/"> Visit Our Site**</a>**	Visit Our Site (Tip)
<b>	**bold**	**<b>**Example**</b>**	**Example**
<big>	**big (text)**	**<big>**Example**</big>**	Example (Tip)
<body>	**body of HTML document**	**<body>**The content of your **HTML** page**</body>**	Contents of your web page (Tip)
<br>	**line break**	The contents of your page** **The contents of your page	The contents of your web page The contents of your web page
<center>	**center**	**<center>**This will center your contents**</center>**	This will center your contents
<dd>	**definition description**	<dl> <dt>Definition Term</dt> **<dd>Definition of the term</dd>** <dt>Definition Term</dt> **<dd>Definition of the term</dd>** </dl>	Definition Term **Definition of the term** Definition Term **Definition of the term**
<dl>	**definition list**	**<dl>** <dt>Definition Term</dt> <dd>Definition of the term</dd> <dt>Definition Term</dt> <dd>Definition of the term</dd> **</dl>**	Definition Term Definition of the term Definition Term Definition of the term
<dt>	**definition term**	<dl> **<dt>Definition Term</dt>** <dd>Definition of the term</dd> **<dt>Definition Term</dt>** <dd>Definition of the term</dd> </dl>	**Definition Term** Definition of the term **Definition Term** Definition of the term
<em>	**emphasis**	This is an **<em>**Example**</em>** of using the emphasis tag	This is an *Example* of using the emphasis tag
<embed>	**embed object**	**<embed** src="yourfile.mid" width="100%" height="60" align="center">	(Tip)
<embed>	**embed object**	**<embed** src="yourfile.mid" autostart="true" hidden="false" loop="false"> <noembed><bgsound src="yourfile.mid" loop="1"></noembed>	Music will begin playing when your page is loaded and will only play one time. A control panel will be displayed to enable your visitors to stop the music.
<font>	**font**	**<font** face="Times New Roman">Example**</font>**	Example (Tip)



<input>	**input field**	Example 1: <form method=post action="/cgi-bin/example.cgi"> **<input** type="text" size="10" maxlength="30"> **<input** type="Submit" value="Submit"> </form>	Example 1: (Tip) Submit
<input> (Internet Explorer)	**input field**	Example 2: <form method=post action="/cgi-bin/example.cgi"> **<input** type="text" style="color: #ffffff; font-family: Verdana; font-weight: bold; font-size: 12px; background-color: #72a4d2;" size="10" maxlength="30"> **<input** type="Submit" value="Submit"> </form>	Example 2: (Tip) Submit
<input>	**input field**	Example 3: <form method=post action="/cgi-bin/example.cgi"> <table border="0" cellspacing="0" cellpadding="2"><tr><td bgcolor="#8463ff"><input type="text" size="10" maxlength="30"></td><td bgcolor="#8463ff" valign="Middle"> <input type="image" name="submit" src="yourimage.gif"></td></tr> </table> </form>	Example 3: (Tip) GO
<input>	**input field**	Example 4: <form method=post action="/cgi-bin/example.cgi"> Enter Your Comments: <textarea wrap="virtual" name="Comments" rows=3 cols=20 maxlength=100></textarea> **<input** type="Submit" value="Submit"> **<input** type="Reset" value="Clear"> </form>	Example 4: (Tip) Submit Clear
<input>	**input field**	Example 5: <form method=post action="/cgi-bin/example.cgi"> <center> Select an option: <select> <option >option 1</option> <option selected>option 2</option> <option>option 3</option> <option>option 4</option> <option>option 5</option> <option>option 6</option> </select> **<input** type="Submit" value="Submit"></center> </form>	Example 5: Tip) Select an option: option 2 Submit
<input>	**input field**	Example 6:	Example 6: (Tip)

CHAPTER #04

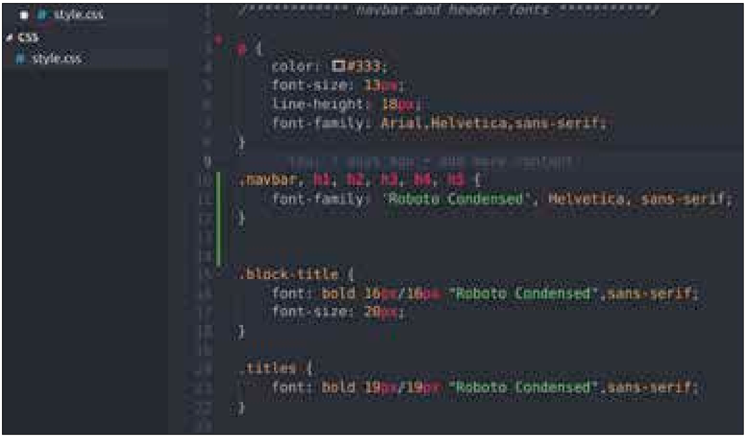
CSS

The cascading style is very interesting

CSS, or Cascading Style Sheets, is a very popular coding language, and is one is the most important ones to learn in today's times. This is because this is a tool that lets you set specific layouts, change fonts and colours, create effects and properly customise the look and feel of websites made with HTML. Before CSS had been introduced to developers, they had to rely on HTML only to change different visual aspects — and unfortunately, the results did not look great. However, today, HTML is only used to create the underlying structure of a site. Once that is done, developers use CSS to exercise control over the design choices.

It is important to note that CSS helps you customise nearly every part of formatting you can think of - starting from identifying the right fonts to altering table sizes and setting a text size.

However, CSS' offers far more than just basic formatting. In fact, as modern HTML can now take care of many of these needs, if CSS was only about design, it would have been outdated by now. fter all. Developers want to learn CSS because it the language can help extrapolate a specific style choice over many different pages. After the style is identified in a Cascading Style Sheet, it can be applied by default to any page that is linked with that particular CSS file.



```
/************* navbar and header fonts ************/

* {
    color: □#333;
    font-size: 13px;
    line-height: 18px;
    font-family: Arial,Helvetica,sans-serif;
}

.navbar, h1, h2, h3, h4, h5 {
    font-family: 'Roboto Condensed', Helvetica, sans-serif;
}


.block-title {
    font: bold 16px/18px "Roboto Condensed",sans-serif;
    font-size: 20px;
}

.titles {
    font: bold 19px/19px "Roboto Condensed",sans-serif;
}
```

As you can imagine, this is a feature that saves developers a lot of time and effort as they do not have to go through the trouble of defining every visual aspect of each page manually. This allows them to create a more even, cohesive design without much effort. The language also facilitates quick and easy changes. For instance, if one were to change the default font size, it can be carried out in ust a few moments by simply editing the style sheet.

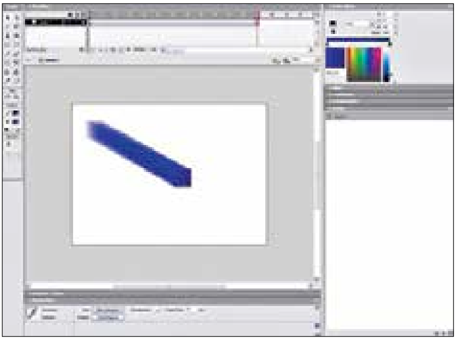
Håkon Wium Lie

CSS was established in October 1994 by Håkon Wium Lie, who was a Norwegian web pioneer. Lie was sick of the lack of styling options that were available to developers. The concept he campe up with was to 'cascade' formatting decisions over different pages. However, while the idea itself seemed popular, it was not launched immediately. This is because it took two years for Lie and Gijsbert (Bert) Bos to come up with a functional version of CSS. It still took 10 years after that for the language to become what it is today.

What is CSS used for?

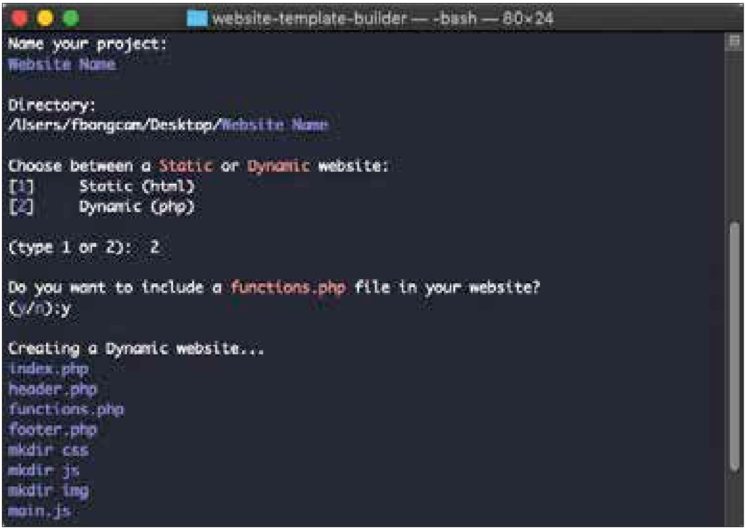
1. Controlling Effects and Flash Animation

CSS makes the process of handling or placing flash elements on websites so much easier. Developers can use CSS to make movies, update effects, and create animations by applying built-in Flash files, frameworks and styling sheets in CSS.



2. Managing Dynamic Web Templates

A dynamic web templates is basically a HTML-based parent copy of a



website-template-builder — -bash — 80×24

```
Name your project:
Website Name

Directory:
/Users/fbangcam/Desktop/Website Name

Choose between a Static or Dynamic website:
[1]     Static (html)
[2]     Dynamic (php)

(type 1 or 2):  2

Do you want to include a functions.php file in your website?
(y/n):y

Creating a Dynamic website...
index.php
header.php
functions.php
footer.php
mkdir css
mkdir js
mkdir img
main.js
```

web page. Such pages typically have many elements like text, graphics, styles, and page layouts. Developers can very easily add, manage, and edit dynamic elements to further style web pages by using different CSS frameworks and extensions.

3. e-Commerce

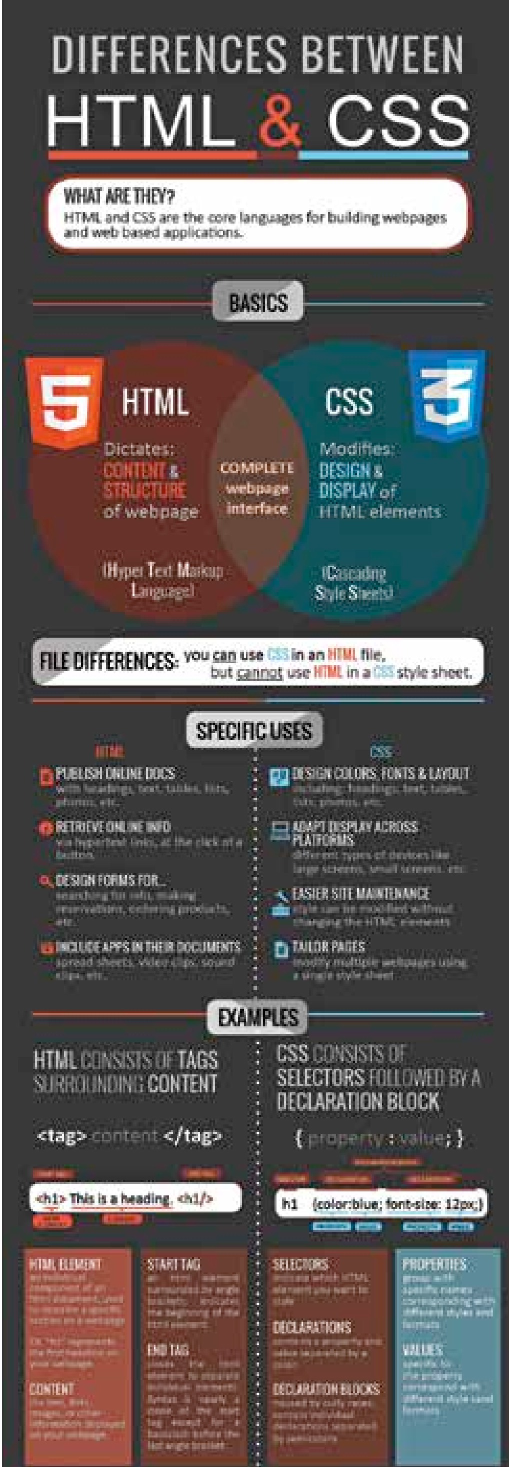
CSS can also help you design and develop an e-commerce website right from scratch. Almost every e-commerce platform uses CSS files in order to enhance the visuals of the website along with the user experience. You can create small sites as well as large portals.

4. Social Media

Did you know that some of your favourite social media platforms like Facebook, Instagram, and Twitter were made using Cascading Style Sheets? The language was used to create dynamic websites that look fantastic on any device. CSS helps developers customise the user interface and make design changes that are put in place from an end-user perspective.

5. Image File Management

Developers can use CSS frameworks to edit, format, and update



DIFFERENCES BETWEEN HTML & CSS

WHAT ARE THEY?
HTML and CSS are the core languages for building webpages and web based applications.

BASICS

HTML
Dictates: CONTENT & STRUCTURE of webpage
(Hyper Text Markup Language)

COMPLETE webpage interface

CSS
Modifies: DESIGN & DISPLAY of HTML elements
(Cascading Style Sheets)

FILE DIFFERENCES: you can use CSS in an HTML file, but cannot use HTML in a CSS style sheet.

SPECIFIC USES

HTML

PUBLISH ONLINE DOCS
with headings, text, tables, lists, photos, etc.

RETRIEVE ONLINE INFO
via hypertext links, at the click of a button.

DESIGN FORMS FOR...
searching for info, making reservations, ordering products, etc.

INCLUDE APPS IN THEIR DOCUMENTS
spread sheets, video clips, sound clips, etc.

CSS

DESIGN COLORS, FONTS & LAYOUT
including: headings, text, tables, lists, photos, etc.

ADAPT DISPLAY ACROSS PLATFORMS
different types of devices like large screens, small screens, etc.

EASIER SITE MAINTENANCE
style can be modified without changing the HTML elements

TAILOR PAGES
modify multiple webpages using a single style sheet

EXAMPLES

HTML CONSISTS OF TAGS SURROUNDING CONTENT

<tag> content </tag>

<h1> This is a heading <h1/>

CSS CONSISTS OF SELECTORS FOLLOWED BY A DECLARATION BLOCK

{ property : value; }

h1 {color:blue; font-size: 12px;}

HTML ELEMENT
an individual component of an html document, used to describe a specific section on a webpage

i.e. "h1" represents the first headline on your webpage

CONTENT
the text, links, images, or other information displayed on your webpage

START TAG
an html element surrounded by angle brackets; indicates the beginning of the html element

END TAG
closes the html element to separate individual elements. Syntax is nearly a copy of the start tag except for a backslash before the last angle bracket

SELECTORS
indicate which HTML element you want to style

DECLARATIONS
contains a property and value separated by a colon

DECLARATION BLOCKS
housed by curly races, contains individual declarations separated by semicolons

PROPERTIES
groups with specific names corresponding with different styles and formats

VALUES
specific to the property, correspond with different style and formats

different image formats. These frameworks can also help with creating watermarks, cropping images, and creating thumbnails. Before CSS was introduced to developers, editing, styling and updating image types was quite challenging.

6. Ease of Accessibility

CSS offers a lot of solutions that can help developers enjoy seamless delivery of web pages. From a business perspective, this has helped developers become more productive and efficient.

7. Website Maintenance

CSS is also used for website maintenance as it can make the process a lot quicker and simpler. Also, websites that are maintained with CSS files tend to be more responsive and flexible. This helps with improving the overall performance, as well as the look, and feel.

8. Server-Side and End-User Representation

CSS is also commonly used to provide end-user and server-side support and this is a critical use of this language. CSS files can alter the end-user and the server-side web pages for interface styling purposes.

Latest CSS Trends

CSS trends tend to be about the visuals, which is why it is important to stay abreast with the latest design trends. These tend to change from year to year, and it is essential to understand design as a concept to be able to create web pages that look current and trendy. For instance, in 2022, design trends like Memphis design, bold typographic designs, use of endless scroll, and clean fonts were hugely popular.

How to get started?

Before creating a CSS file, you need to create an HTML fille.
To create an HTML file, you must:

1. Open TextEditor or Notepad
2. Write the following code:

```
<html lang=”en”>
  <head>
    <meta charset=”utf-8” />
    <title>Getting started with CSS</title>
  </head>

  <body>
    <h1>I am a level one heading</h1>

    <p>
      This is a paragraph of text. In the text is a
      <span>span element</span> and also a
      <a href=”https://example.com”>link</a>.
    </p>

    <p>
      This is the second paragraph. It contains an <em>emphasized</em> element.
    </p>

    <ul>
      <li>Item <span>one</span></li>
      <li>Item two</li>
      <li>Item <em>three</em></li>
    </ul>
  </body>
</html>
```

3. Save the document as index.html
4. Create a similar file in the same folder as the index.html file. Save it as styles.css
5. Add this line to the head of the document - <link rel=”stylesheet” href=”styles.css” />
6. Add random styles to the styles.css document to see if its working

Changes in CSS

In 2022, the latest version of CSS addressed some specific needs that make it easier for developers to create beautiful websites. Some of these include:

1. Subgrid
2. Accent Colour
3. Scroll Snap
4. CSS Logical Properties
5. CSS Container Queries

Basic Code

You can try the following code for practise:

CSS Syntax

```
<!DOCTYPE html>
<html>
<head>
<style>
p {
  color: red;
  text-align: center;
}
</style>
</head>
<body>

<p>Hello World!</p>
<p>These paragraphs are styled with CSS.</p>

</body>
</html>
```

CSS Selectors

```
<!DOCTYPE html>
<html>
<head>
<style>
p {
  text-align: center;
  color: red;
}
</style>
</head>
<body>

<p>Every paragraph will be affected by the style.</p>
<p id=”para1”>Me too!</p>
<p>And me!</p>

</body>
```

```
</html>
```

ID Selector

```
<!DOCTYPE html>
<html>
<head>
<style>
#para1 {
  text-align: center;
  color: red;
}
</style>
</head>
<body>

<p id=”para1”>Hello World!</p>
<p>This paragraph is not affected by the style.</p>

</body>
</html>
```

Class Selector

```
<!DOCTYPE html>
<html>
<head>
<style>
.center {
  text-align: center;
  color: red;
}
</style>
</head>
<body>

<h1 class=”center”>Red and center-aligned heading</h1>
<p class=”center”>Red and center-aligned paragraph.</p>

</body>
</html>
```

Java

The OG is in the house!

Java is a programming language was that developed by Sun Microsystems in 1995. This language has grown into one that powers many of today's digital platforms. Java helps by offering a secure platform that can be used to build services and applications. In spite of many languages now existing, this language is still used to create innovative goods and digital services. It is incredibly versatile and future-proof. This language is fast, secure, and reliable. Developers use Java to construct applications in data centres, scientific supercomputers, laptops, game consoles, cell phones, and other devices. Java is the world's third most popular programming language, after Python and C.

Some of the salient features of Java include:

1. **Versatility:** Java has been the programming language of choice when it comes to developing Android apps, and software development tools like NetBeans IDE, Eclipse, and IntelliJ IDEA.

2. **Development tools:** One of the best features of Java is the Integrated Development Environment (IDE). The Java IDE is basically a selection of different debuggers, automation tools, and editors.
3. **User-friendliness:** Java is ideal for novices as it uses English-like grammar. Developers typically learn the language in two stages, core Java first, then advanced Java.
4. **Good documentation:** Java is an open-source programming language. This means that it is completely free. The language itself is well-documented and this is an important feature. It has a full guide that will help you navigate many issues that you may encounter while coding.
5. **A robust API:** There are only 50 keywords in Java. But, the Application Programming Interface (API) is comprehensive and wide. There are various methods that can be used in any code.
6. **A large community:** Java is immensely popular and this is because there is a lot of community support. It ihas the second-largest Stack Overflow community.

What is Java Used for?

Java has many different applications. Some of these are:

1. Video games

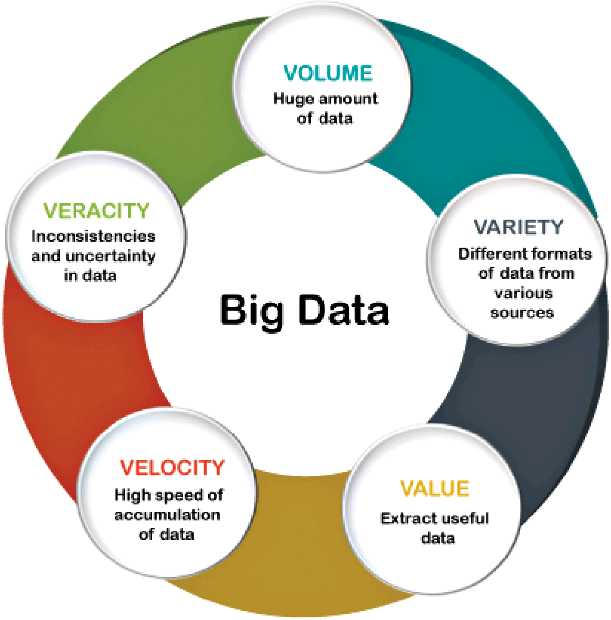
Java is a great language for those who want to study graphics and game development, particularly from the ground up. There are many different libraries and frameworks that are available, for instance, LibGDX and OpenGL.

2. Web-based programmes

Java is basically a server-side programming language. It is typically used for constructing web applications. There are many frameworks available for constructing web applications, like Spring, JSF, Hibernate, Struts, and Apache Hadoop. Websites like Amazon, Linkedin and AliExpress are made using Java.

3. Big data

When it comes to creating platforms that leverage Big Data, most developers like to use Java. This is mainly because many Big Data tools, like Hadoop and Deeplearning4j, use this language and it offers excellent community support.


VOLUME
Huge amount of data
VERACITY
Inconsistencies and uncertainty in data
VARIETY
Different formats of data from various sources
Big Data
VELOCITY
High speed of accumulation of data
VALUE
Extract useful data

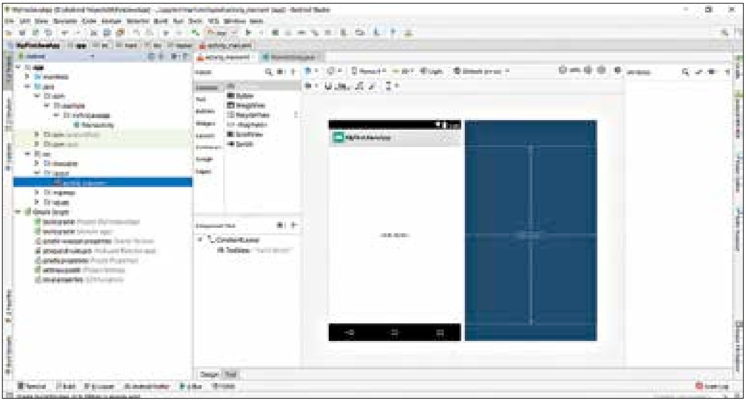
4. Internet of Things (IoT)

Many developers believe that Java is the best language for developing IoT devices. There are many reasons for this, for instance,, it is a platform agnostic programming language. What this means is that you only have to create your code one time. It will then operate properly on a variety of devices automatically. Java is also lauded for its adaptability and mobility.

It can be used for web development, desktop app development, Android development, and a variety of other things. Additionally, integrating multiple technologies is so much easier with Java. As a result, it is used commonly in automotive navigation systems, embedded electronics, and even various medical gadgets.

5. Android apps

Did you know that Java is the official programming language for Android development? 46.2% of Android applications are made with Java. This language is also widely used by Android Studio, and this is significant because it is the official IDE that develops Android apps.



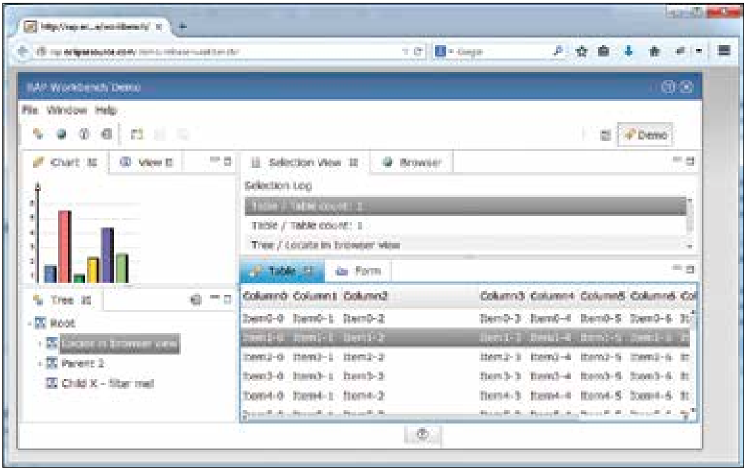
6. Desktop applications

Java can also help you create graphical user interfaces that are intended for desktop applications. For this, there are many Java Frameworks available (like Swing, AWT, JavaFX, and Griffon). As a result, GUI Application Development becomes a breeze.

Latest Java Trends

1. DevOps

One of the biggest trends to keep in mind is DevOps, or developer operations. This trend refers to the comping together of software development and information technology (IT) teams. With these teams merging, the DevOps teams are typically familiar with all aspects of application lifestyles.



Selection Log

Table / Table count: 1

Table / Table count: 1

Tree / Locate in browser view

Column0	Column1	Column2	Column3	Column4	Column5	Column6	Col
Item0-0	Item0-1	Item0-2	Item0-3	Item0-4	Item0-5	Item0-6	It
Item1-0	Item1-1	Item1-2	Item1-3	Item1-4	Item1-5	Item1-6	It
Item2-0	Item2-1	Item2-2	Item2-3	Item2-4	Item2-5	Item2-6	It
Item3-0	Item3-1	Item3-2	Item3-3	Item3-4	Item3-5	Item3-6	It
Item4-0	Item4-1	Item4-2	Item4-3	Item4-4	Item4-5	Item4-6	It

2. Git

Git is a technology that every Java developer should learn (but few do). There is a web-based counterpart known as GitHub that you can learn. Developers who can use this are always in-demand.

3. Big Data

Big data is one of the emerging trends in the IT world and it refers to large and complex data sets that are used for various purposes. The Big Data industry also refers to the methods data engineers generally use to deal with all the info. Very few programming languages are like Java are useful when it comes to Big Data.

4. Remote Access Solutions

With the COVID-19 pandemic having altered how professionals get their work done, this is the perfect time to learn Java. The language offers great security features. It also has really useful unique back-end capabilities, and because of this, enterprise-scale solutions that use Java tend to fare well.

5. Serverless Architecture

Serverless architecture (also known as serverless computing) refers to an application that executes without the developer having to manage a physical server. Cloud technology is an example of this. Java offers plenty of tools that are designed for cloud development, like Google App Engine, AWS SDK for Java, and Oracle Java Cloud Service.

How to get started?

To get started with Java, you first have to download it. You can do so for free by heading to *www.oracle.com.* Once it is downloaded, you need to install it on your computer.

1. Go to Control Panel
2. Select System and Security.
3. Select System.
4. Select Advanced System Settings.
5. Select System Properties
6. Click the Environment variables button under the Advanced tab
7. Select the Path variable in System variables and click the Edit button
8. Click the New button and add the path where Java is installed, followed by \bin. By default, Java is installed in C:\Program Files\Java\jdk-11.0.1
9. Click OK to save the settings.
10. Open Command Prompt (cmd.exe) and type java -version to see if Java is running on your machine.

Basic Code

You can try this code to practise on your own!

First program

```
// Your First Program

class HelloWorld {
    public static void main(String[] args) {
        System.out.println(“Hello, World!”);
    }
}
```

Single line comments

```
public class Main {
  public static void main(String[] args) {
    // This is a comment
    System.out.println(“Hello World”);
  }
}
```

Create a string variable

```
public class Main {
  public static void main(String[] args) {
    String name = “John”;
    System.out.println(name);
  }
}
```

Create an integer variable

```
public class Main {
  public static void main(String[] args) {
    int myNum = 15;
    System.out.println(myNum);
  }
}
```

Create a variable without assigning value

```
public class Main {
  public static void main(String[] args) {
    int myNum;
    myNum = 15;
    System.out.println(myNum);
  }
}
```

CHAPTER #06

SQL

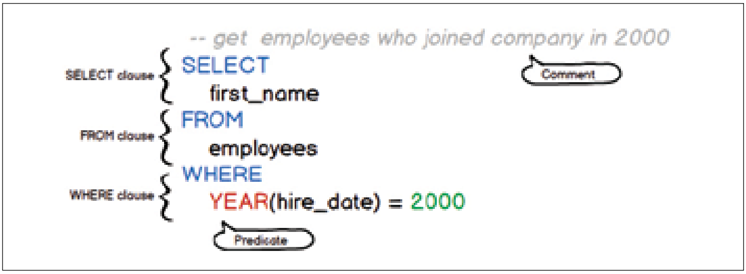
Managing the database

Structured Query Language (SQL) is one of the most popularly used programming languages. It is typically used in relational database or data stream management systems.

SQL was developed by IBM in the early 1970s. Now, it is an official standard that is recongised by the American National Standards Institute (ANSI) as well as the International Organization for Standardization (ISO). SQL is extremely simple to use. It has a highly effective mode for querying, manipulating, aggregating data as well as performing a wide range of functions that can take vast collections of structured data and turn it into usable information. This is why it is a popular choice for database users.

Due to its popularity and ease of use, the language has been incorporated into numerous commercial database products. Some of these products include Oracle, MySQL, SQL Server, Sybase, Postgres and others. There are a range of non-relational databases such as MongoDB and DynamoBD. These are known as NoSQL products as they lack SQL programming.

While different versions of this language may use their own syntax for vital operations, in general, basic syntax for commands like select, create, insert, and update are common to all SQL releases. Such syntax makes it very simple for programmers with just a little knowledge of SQL to perform tasks in many different environments.

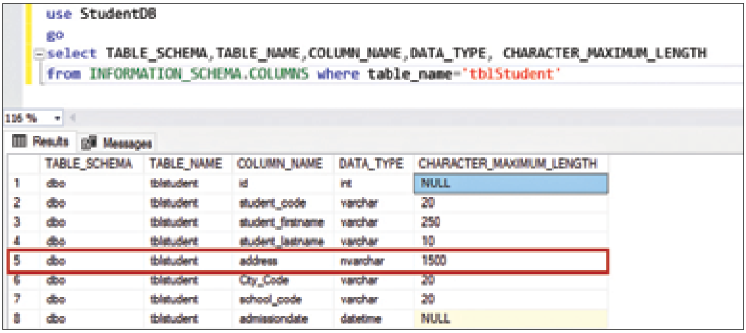


```
-- get  employees who joined company in 2000
SELECT
    first_name
FROM
    employees
WHERE
    YEAR(hire_date) = 2000
```

What is it used for?

1. Alter data within a table

SQL can be used to manipulate data that is presented in a table in a database. For example, developers can alter certain data points in table. Moreover, you can change every part of the table with updated data.



```
use StudentDB
go
select TABLE_SCHEMA,TABLE_NAME,COLUMN_NAME,DATA_TYPE, CHARACTER_MAXIMUM_LENGTH
from INFORMATION_SCHEMA.COLUMNS where table_name='tblStudent'
```

	TABLE_SCHEMA	TABLE_NAME	COLUMN_NAME	DATA_TYPE	CHARACTER_MAXIMUM_LENGTH
1	dbo	tblstudent	id	int	NULL
2	dbo	tblstudent	student_code	varchar	20
3	dbo	tblstudent	student_firstname	varchar	250
4	dbo	tblstudent	student_lastname	varchar	10
5	dbo	tblstudent	address	nvarchar	1500
6	dbo	tblstudent	City_Code	varchar	20
7	dbo	tblstudent	school_code	varchar	20
8	dbo	tblstudent	admissiondate	datetime	NULL

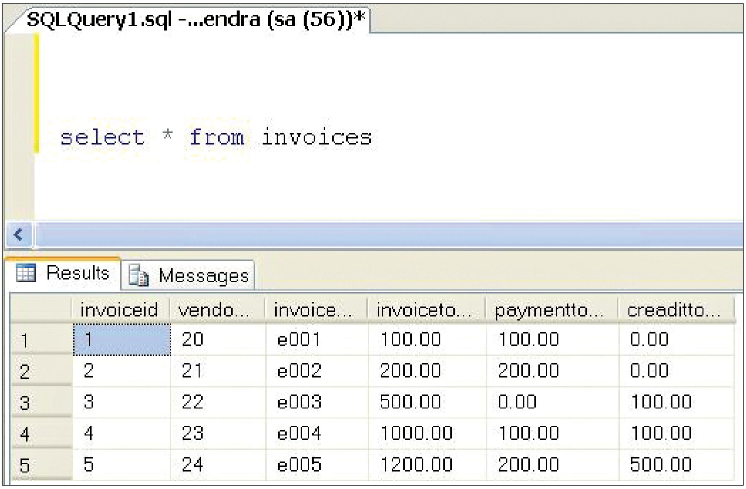
2. Create a table

You can use SQL to create a new table. Developers can easily create a brand new database. They just have to add new data in the table once it's created.

This function allows developers and programmers to add data to the system, and this can further analyse, store and manipulate data.

3. Retrieve data

Data scientists can also leverage SQL in order to retrieve data from the database. The process for this is known as DQL or Data Query Language. The syntax for this command is "SELECT." It will be easy to analyse and assess the data when you retrieve it.



```
select * from invoices
```

	invoiceid	vendo...	invoice...	invoiceto...	paymentto...	creaditto...
1	1	20	e001	100.00	100.00	0.00
2	2	21	e002	200.00	200.00	0.00
3	3	22	e003	500.00	0.00	100.00
4	4	23	e004	1000.00	100.00	100.00
5	5	24	e005	1200.00	200.00	500.00

4. Change data structure

SQL can also be used to change the structure of the data. This can be done by altering the database or table in which the data's stored. A programmer or developer can use SQL to achieve this purpose when there's new data that makes the current data outdated.

5. Define the database's schema

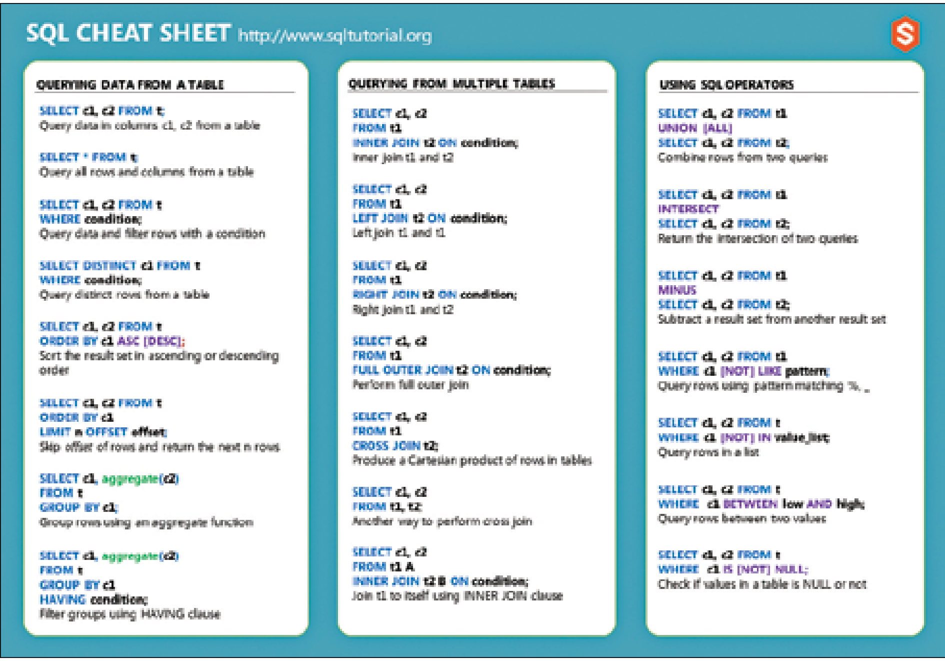
SQL can define the database's schema. This then adds to data descriptions and structuring. Some of the commands that can be used include "CREATE," "ALTER," "DROP," "COMMENT" "TRUNCATE," and "RENAME."

6. Manipulate the data

There is a feature known as DML that offers different features to store, delete, modify, or update the data. The commands for this are “INSERT,” “UPDATE” and “DELETE.”

7. Grant access to data

Developers can use SQL to provide access to other users. Developers use Design Control Language (DCL) for this. You can leverage certain commands “GRANT” and “REVOKE” to select which users have access.



SQL CHEAT SHEET http://www.sqltutorial.org

QUERYING DATA FROM A TABLE

```
SELECT c1, c2 FROM t;
```
Query data in columns c1, c2 from a table

```
SELECT * FROM t;
```
Query all rows and columns from a table

```
SELECT c1, c2 FROM t
WHERE condition;
```
Query data and filter rows with a condition

```
SELECT DISTINCT c1 FROM t
WHERE condition;
```
Query distinct rows from a table

```
SELECT c1, c2 FROM t
ORDER BY c1 ASC [DESC];
```
Sort the result set in ascending or descending order

```
SELECT c1, c2 FROM t
ORDER BY c1
LIMIT n OFFSET offset;
```
Skip offset of rows and return the next n rows

```
SELECT c1, aggregate(c2)
FROM t
GROUP BY c1;
```
Group rows using an aggregate function

```
SELECT c1, aggregate(c2)
FROM t
GROUP BY c1
HAVING condition;
```
Filter groups using HAVING clause

QUERYING FROM MULTIPLE TABLES

```
SELECT c1, c2
FROM t1
INNER JOIN t2 ON condition;
```
Inner join t1 and t2

```
SELECT c1, c2
FROM t1
LEFT JOIN t2 ON condition;
```
Left join t1 and t1

```
SELECT c1, c2
FROM t1
RIGHT JOIN t2 ON condition;
```
Right join t1 and t2

```
SELECT c1, c2
FROM t1
FULL OUTER JOIN t2 ON condition;
```
Perform full outer join

```
SELECT c1, c2
FROM t1
CROSS JOIN t2;
```
Produce a Cartesian product of rows in tables

```
SELECT c1, c2
FROM t1, t2;
```
Another way to perform cross join

```
SELECT c1, c2
FROM t1 A
INNER JOIN t2 B ON condition;
```
Join t1 to itself using INNER JOIN clause

USING SQL OPERATORS

```
SELECT c1, c2 FROM t1
UNION [ALL]
SELECT c1, c2 FROM t2;
```
Combine rows from two queries

```
SELECT c1, c2 FROM t1
INTERSECT
SELECT c1, c2 FROM t2;
```
Return the intersection of two queries

```
SELECT c1, c2 FROM t1
MINUS
SELECT c1, c2 FROM t2;
```
Subtract a result set from another result set

```
SELECT c1, c2 FROM t1
WHERE c1 [NOT] LIKE pattern;
```
Query rows using pattern matching %, _

```
SELECT c1, c2 FROM t
WHERE c1 [NOT] IN value_list;
```
Query rows in a list

```
SELECT c1, c2 FROM t
WHERE  c1 BETWEEN low AND high;
```
Query rows between two values

```
SELECT c1, c2 FROM t
WHERE  c1 IS [NOT] NULL;
```
Check if values in a table is NULL or not

8. Analyze data manually

SQL lets developer analyze the data within the system manually. This is essential when developers need to intervene manually. You can use SQL queries to find the necessary data and then use it for analysis.

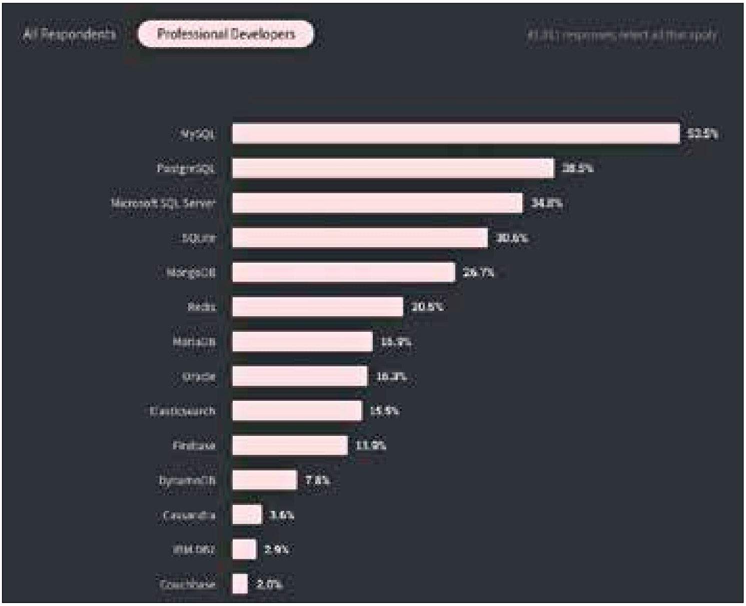
9. Combine datasets

The SQL Join function will let you simply combine data from two sets. Developers turn to this feature when they want to combine and compare two different tables.

Latest SQL Trends

1. SQL Is on the rise

Around 10 years ago, new NoSQL entrants were poised to supplant SQL-based DBMS. However, now, , it's believed that SQL will stay strong as a cornerstone of DBMS. In fact, new machine learning-based offerings have used SQL as the default querying language.Some examples of these are MindDB's ML framework and AWS Redshift ML.



2. ML-Driven Databases

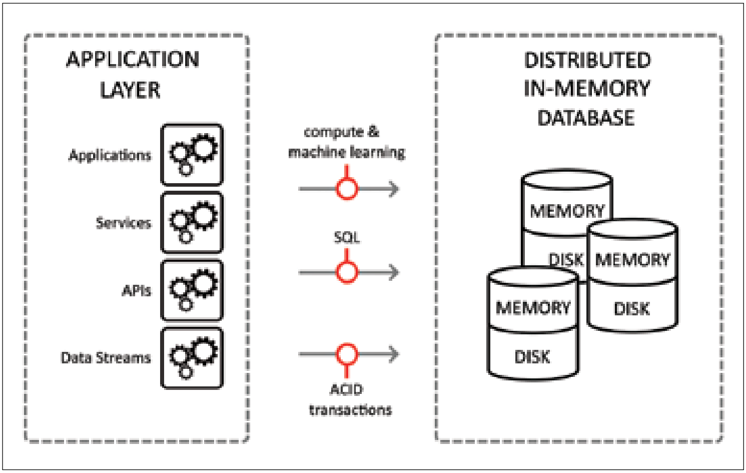
There is a rising trend of integrating ML models, especially in platforms wherein data lives are seen as a standard practice with vendors. There are many solutions such as Microsoft SQL Server Machine Learning Services and Oracle Autonomous Database on the enterprise side. There are also MindsDB and SingleStore startup offerings.

3. Microservice Integration

Today's modern software engineering teams are now accustomed to design and build applications by turning to a microservices approach. What this means is that they create applications that are marketed as a series of smaller, API-driven services. This improves scalability and agility.

4. In-Memory Databases

Modern mission-critical software solutions rely on minimal database latency in order to offer optimal performance. But, traditional DBMS have incredibly sluggish disk read/write operations when it comes to storing data to media (some examples are hard disk drives and solid-state drives). Due to this, in-memory databases are now seen as viable alternatives for these vital use cases. As records are retrieved and stored directly from memory (RAM), developers can expect a more reliable and faster performance.



5. Stronger Database Security Layers

In today's times, cyber attacks and data breaches are a common reality that must be mitigated. As a result, there is a high demand for securing the data layer of the software application. Due to this, many service providers are trying to augment their offerings with stronger security features.

How to get started?

In order to get started, you must:

1. Download MySQL. To do this, you need to download MySQL Workbench and MySQL Community.
2. Install both and open the MySQL Workbench application.

3. Create a database.
4. Click the Lightning icon to run the code.
5. Head to the Query tab.
6. Click Create Table
7. Add data by using the Insert Command

Basic Code

Select all records from a specific table

- `SELECT * FROM Customers;`

SQL Select

- `SELECT CustomerName,City FROM Customers;`

Select*

- `SELECT * FROM Customers;`

Select Distinct

- `SELECT DISTINCT Country FROM Customers;`

Combining where, and, or

- `SELECT * FROM Customers`
- `WHERE Country=’Germany’ AND (City=’Berlin’ OR City=’München’);`

NoSQL

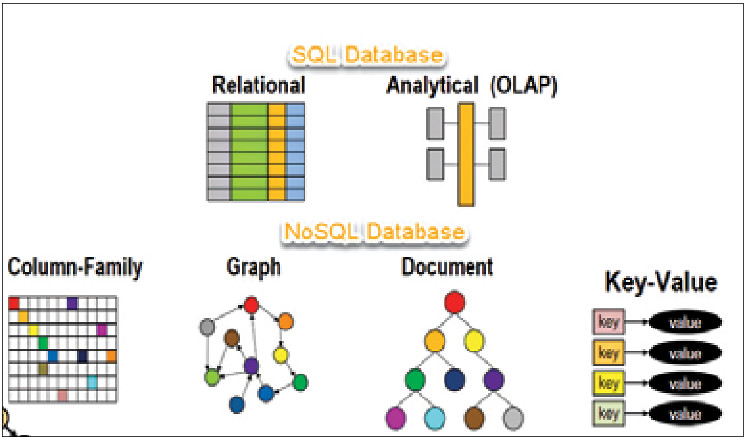
Well, if you say so. NO SQL...

NoSQL, also known as “not only SQL” as well as “non-SQL”, is a well-known approach to database design. It helps by enabling storage as well as the querying of data and this is done outside the traditional structures that are generally found in relational databases. Though the language can store data that is found within relational database management systems (RDBMS), it actually stores it in a different way when you compare it to an RDBMS.

In other words, NoSQL is a database that doesn’t stick to the widely used relational database management model. What this means is that NoSQL databases are not mainly built on tables, and they do not use SQL to manipulate data – this is where the name comes from. However, it is important to note that NoSQL was created to function as a support system for SQL, not as its replacement. The language is based on a model that is considered to be fairly less stringent. Also, it does not follow a fixed schema.

NoSQL may also refrain from sticking to the ACID properties, and, unlike in most of the RDBMSs, there is no such thing as JOIN.

It is important to understand that NoSQL is kind of distributed database. As a result, the information is added and stored on differed servers, and these servers can be remote or local. Due to this, you can expect the availability and reliability of data. Even if a few chunks of the data go offline, the remaining chunks of the database will still run.



In today's times, companies have to be able to manage large data volumes while offering high speeds. This is needed along with the ability to scale up quickly so that they can run modern web applications no matter which industry its related to. In a time when the cloud, big data, and mobile and web applications, are highly prevalent, NoSQL databases offer a lot of speed and scalability. It is therefore a popular choice for their ease of use and performance.

What is it used for?

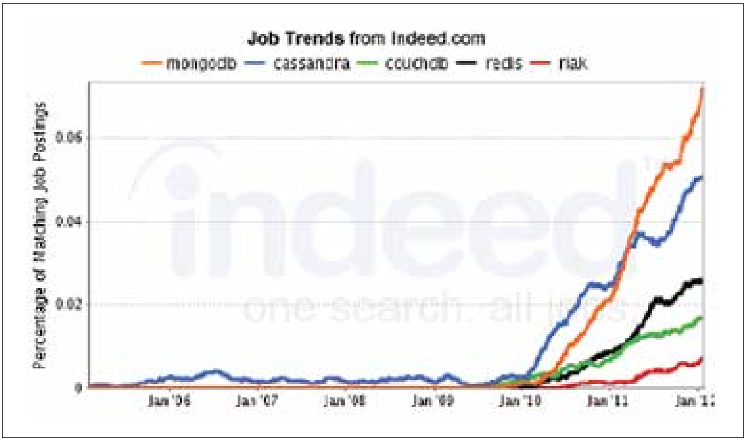
Here are some specific uses for various types of NoSQL databases.

1. **Managing data relationships:** Managing the challenging collection of data and the relationships between different data points is easily handled by using a graph-based NoSQL database. In this, the system uses many different knowledge graphs, fraud detection applications, recommendation engines, and social networks, where connections are established between different consumers who are using various data types.

2. **Low-latency performance:** A high throughput for real-time data management is essential for ad technology, home fitness applications and gaming. Such types of infrastructures help by providing a lot of value to the consumer, and this can range from updates regarding market bidding or returning highly relevant ads. Web applications need different types of in-memory NoSQL databases. It allows them to ensure a rapid response time while also managing spikes in usage. All this is done without the lag that is associated with disk storage.
3. **Scaling and large data volumes:** E-commerce relies heavily on data. Such platforms need to have the ability to deal with major spikes in usage. This is applicable for one-day sales as well as the holiday shopping season. Key-value databases are typically used in e-commerce applications as such databases have a very simple structure that can be easily scaled up when there are times of heavy traffic. This feature is very important for gaming, adtech, and Internet of Things (IoT) applications.

Latest NoSQL Trends

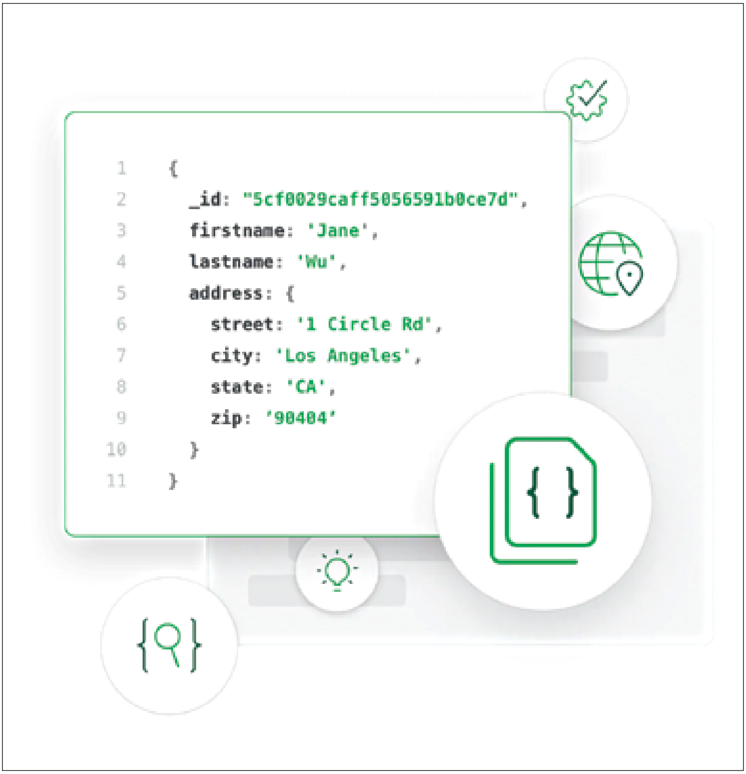
The biggest trend in the NoSQL realm is microservices. Large companies now need to offer services without latency. In order to scale their services quickly, microservices are offered to clients.The growth of microservices has led to companies examining the kind of database that should be used for different applications.



Source: Indeed

Job trends of five NoSQL Databases

In fact, most companies have realised that it does not make the most sense to use a single, relational database for every component of an application. This is because it has limitations, and this is especially true when you consider that there are many better alternatives for specific components. Microservices are a great option. This is mainly because they get rid of the need for a single, shared data store that can be used for an entire application. Now, the application has different independently deployable services. Each of these services come with their own database and data model and database. They also have their own integrated via API gateways or an iPaaS. Polyglot persistence, which is the practise of using multiple databases in just one application, has also ended up creating a space for NoSQL databases to thrive.



```
{
  _id: "5cf0029caff5056591b0ce7d",
  firstname: 'Jane',
  lastname: 'Wu',
  address: {
    street: '1 Circle Rd',
    city: 'Los Angeles',
    state: 'CA',
    zip: '90404'
  }
}
```

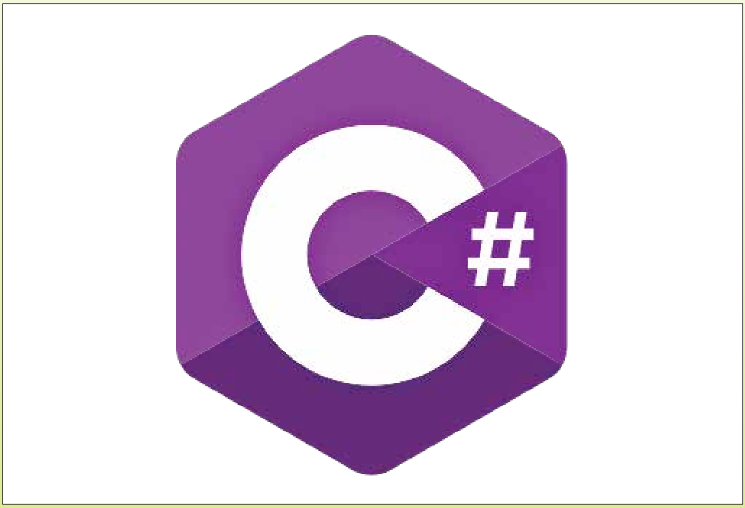
MongoDB

How to get started?

Based on your requirements, you can either download:

1. MongoDB Customers Page
2. Powerd by Haddop
3. Neo4J Customers Page

After installing these programs, you can run the code to execute the required commands.



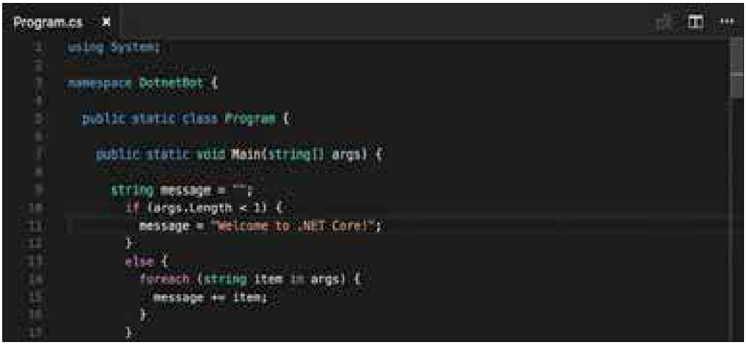
C#

Not your average C language

C# (sometimes spelled as C Sharp or C-Sharp) is known as a object-oriented programming (OOP) language that is mainly used for networking and Web development. This language is categorised as a Common Language Infrastructure (CLI) language.

In January 1999, Anders Hejlsberg (Dutch software engineer) decided to form a team to create C# to complement Microsoft's NET framework. In the beginning, C# was created as C-Like Object Oriented Language (Cool). However, the name was later changed to prevent potential trademark issues. NET was then released as C# in 2002. The NET framework is known for promoting multiple Web technologies.

Did you know that the term's # character is derived from the musical sharp key? This symbol is used to denote a one semitone increase in pitch. C# is known for improving and updating many C and C++ feature. Some of these updates include:



Program.cs

```
using System;

namespace DotnetBot {

  public static class Program {

    public static void Main(string[] args) {

      string message = "";
        if (args.Length < 1) {
          message = "Welcome to .NET Core!";
        }
        else {
          foreach (string item in args) {
            message += item;
          }
        }
```

Sample C# code

- C# has a strict Boolean data variable type, such as bool. On the other hand, C++ bool variable types can be returned as integers or pointers. This is done to avoid common programming errors.
- C# uses a garbage collector to automatically manage inaccessible object memory. , This will eliminate developer concerns as well as memory leaks.
- C# type is safer than C++ . This language also has safe default conversions only (for example, integer widening). These are implemented when the compilation is carried out (or at runtime)
- No implicit conversions between enumeration members, Booleans, integers (apart from O) can be converted to an enumerated type. User-defined conversions have to be specified as implicit or explicit. However in C++, there are default implicit conversion operators and copy constructors.

	Id	Name	DateOfBirth	Occupation	MaritalStatus	HealthStatus	Salary	Numb
▶	[illegible]	Gregory	20/09/1987	Doctor	Married	Excellent	12000	2
	18	Leonardo	20/09/1988	Professor	Married	Good	20000	2
	20	[illegible]	07/06/1964	Professor	Married	Excellent	23000	1
	22	Hemsworth	21/09/1979	Professor	Single	Good	12000	
	24	[illegible]	21/09/1980	Engineer	Married	Excellent	10000	1
	25	Tarun	22/09/2012	Doctor	Married	Excellent	200000	3
	26	[illegible]	22/09/2012	Doctor	Single	Excellent	10000	2
	27	Parull	05/07/1988	Engineer	Single	Excellent	23000	0
	28	Woodhouse	05/07/1989	Engineer	Married	Good	18000	1
	30	Jim Alan	15/02/1990	Doctor	Married	Excellent	9000	2
	32	John	26/12/1979	Engineer	Married	Excellent	4200	2

Name * Gregory
Date of Birth * 20/09/1987
Occupation * Doctor
Salary 12000
Marital status * Married
Health status * Excellent
No. of children 2

What is it used for?

Some of the many uses of C# are:

1. Desktop application development

C# is commonly used for desktop application development as the language offers a ton of versatility and ease of use.

2. Web application and service creation

C# is used for web application and service creation as it offers features related to scalability and efficiency making it ideal for client-related projects.

3. Microsoft application development

As C# was made to supplement *Microsoft's .NET*, it is mainly used for application development for Windows OS.

4. Game development

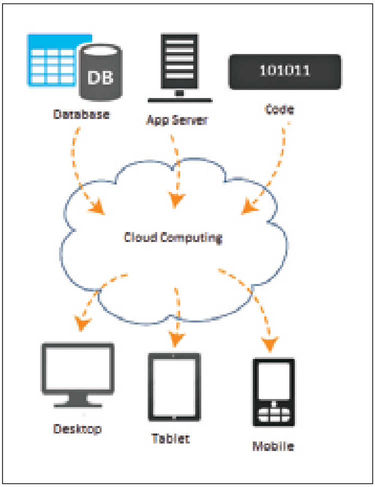
C# is also a popular language when it comes to game development as it has a vast library that users can access when working on projects. The C# community is also a supportive one.

5. Mobile application development

C# can be used for mobile application development across all platforms.

6. Cloud-based service development

C# can also be used to create software and applications that leverage Cloud computing for clients across all industries.



Latest C# Trends

1. .NET 5

Net 5, is the latest version *of .NET*, and it includes UWP, WPF, and Winforms. Here are some features that have contributed to the rise of.NET 5

- The entity framework core, EF Core, will eventually replace EF 6.
- ASP.Net Core MVC is a blend of ASP.NET MVC and Web API.
- MSIX: This is a new desktop application packing tool. It superseded the MSI package installer.
- JsonDocument: The Json Document library replaces System.Text.Json. JsonDocument.
- 3.2 Soaring Open-Source Platforms
- .NET 5 governs apps with an open-source platform.

2. Machine Learning .Net 1.4

One of the most important updates *in .NET* app development that you should pay attention to is the introduction of a Machine learning element. This feature allows you to use a command-line interface and model builder program. With this, you can build and own an automated Machine learning model for your client's business.

3. Azure Kubernetes Service

Azure Kubernetes Services makes it easy for developers to create web applications. As a developer, you can also enjoy serverless Kubernetes along with continuous delivery knowledge. This can also help with governance and enterprise-grade security.

Note that Kubernetes is a developer-friendly environment. It lets developers use all functions from testing, designing, and deploying microservice-oriented solutions. It also leads to major amount of manual work.

4. Blazor Framework in C#

Blazor is a major emerging trend. Blazor has the following characteristics:

- Performs routing and develops simpler layouts
- Apply forms and validation
- Blazor also offers an Injection of dependability
- JavaScript compatibility
- Rendering on specifically Server-side
- It does not require any additional plugins
- Blazor is compatible with all contemporary web browsers, as well as mobile browsers
- 3.6 The Popularity of .NET Core
- The .NET Core, one of the .Net trends is anticipated to continue in 2020. This framework consists of features like AOT, Runtime, GC, Base Class

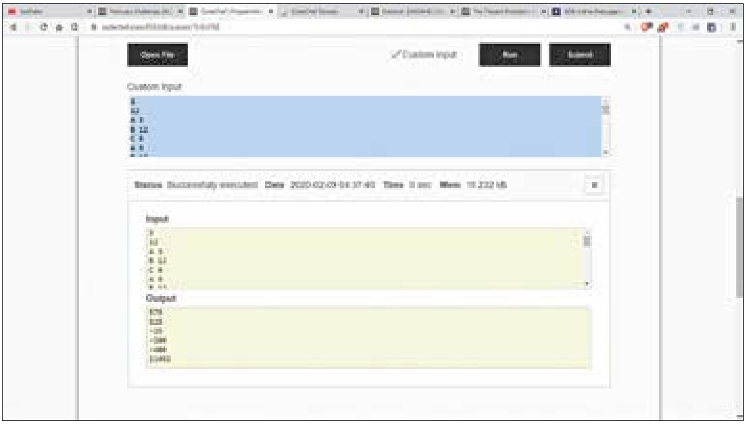
Library, ASP.NET, JIT, C#, VB.NET, Entity Framework, ML.NET, WPF, F#, WinForms, and Xamarin.

How to get started?

There are many ways to get started with C#. Some of these are:

1. Finding a Compiler:

There are various online IDEs such as CodeChef ide, GeeksforGeeks ide, etc. You can use these to run C# programs without installing anything.



Codechef IDE

2. Compilers for Windows

C# is developed *within .Net* framework and this was an initiative carried out by Microsoft. The company has various IDEs to run C# programs. Some examples are Microsoft Visual Studio, Visual Web Developer and Visual Studio Express.

3. Compilers for Linux

Mono can be used to run C# programs on Linux.

4. Programming in C#:

It is very easy to code and learn in C#, and this is because C# is fairly similar to other widely used languages syntactically. Programs can be written in C# by using any widely used text editor such as Notepad++, gedit, and so on.

Basic Code

Here’s some basic code for you to practise:

Hello People

```
// C# program to print Hello People
using System;

namespace HelloGeeksApp
{
    class HelloGeeks
    {
        // Main function
        static void Main(string[] args)
        {

            // Printing Hello People
            Console.WriteLine(“Hello People”);

            Console.ReadKey();
        }
    }
```

Single line comment before code

```
using System;

namespace HelloWorld
{
  class Program
  {
    static void Main(string[] args)
    {
      // This is a comment
      Console.WriteLine(“Hello World!”);
     }
  }
}
```

Create a string variable

```
using System;
```

```
namespace MyApplication
{
  class Program
  {
    static void Main(string[] args)
    {
      string name = “John”;
      Console.WriteLine(name);
     }
  }
}
```

Create an integer variable

```
using System;

namespace MyApplication
{
  class Program
  {
    static void Main(string[] args)
    {
      int myNum = 15;
      Console.WriteLine(myNum);
     }
  }
}
```

Creating a variable without assigning value

```
using System;

namespace MyApplication
{
  class Program
  {
    static void Main(string[] args)
    {
      int myNum;
      myNum = 15;
      Console.WriteLine(myNum);
      }
  }
}
```

Declare many variables

```
using System;

namespace MyApplication
{
  class Program
  {
    static void Main(string[] args)
    {
      int x = 5, y = 6, z = 50;
      Console.WriteLine(x + y + z);
    }
  }
}
```

Different data types

```
using System;

namespace MyApplication
{
  class Program
  {
    static void Main(string[] args)
    {
      int myNum = 5;                // integer (whole
number)
      double myDoubleNum = 5.99D;  // floating point number
      char myLetter = ‘D’;         // character
      bool myBool = true;           // boolean
      string myText = “Hello”;     // string
      Console.WriteLine(myNum);
      Console.WriteLine(myDoubleNum);
      Console.WriteLine(myLetter);
      Console.WriteLine(myBool);
      Console.WriteLine(myText);
    }
  }
}
```

User input

```
using System;

namespace MyApplication
{
  class Program
  {
    static void Main(string[] args)
    {
      // Type your username and press enter
      Console.WriteLine(“Enter username:”);

      // Create a string variable and get user input from
the keyboard and store it in the variable
      string userName = Console.ReadLine();

      // Print the value of the variable (userName), which
will display the input value
      Console.WriteLine(“Username is: “ + userName);
    }
  }
}
```

Addition operators

```
using System;

namespace MyApplication
{
  class Program
  {
    static void Main(string[] args)
    {
      int x = 5;
      int y = 3;
      Console.WriteLine(x + y);
     }
  }
}
```

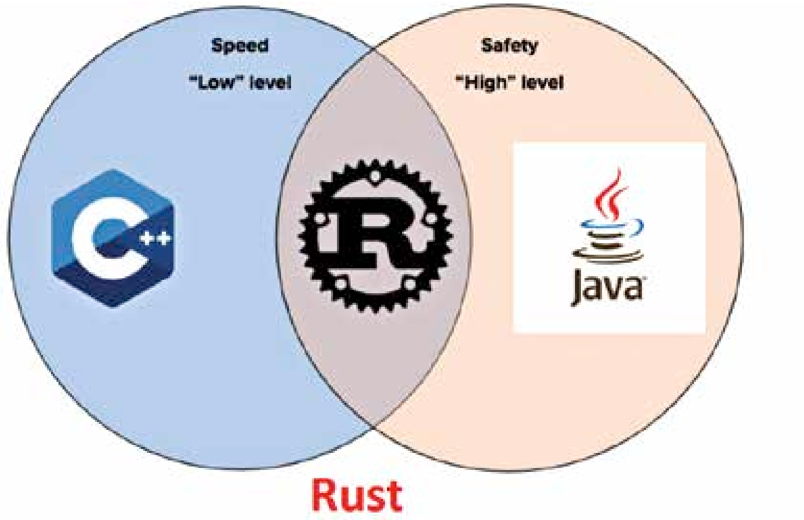
CHAPTER #09

Rust

Rusty not so much

Rust is a programming language that has object-oriented characteristics. It is often labelled as an alternative to traditional C++. Though it is similar to C++ in terms of syntax, it also has its own set of features that make it distinct from its counterpart. It is a multi-paradigm language that is a very popular one. Some of the practical benefits of Rust are a broad user base and low overhead. It also offers different functionality and can also be compiled into Web Assembly. This language often wins the StackOverflow survey, indicating that many developers prefer Rust over other languages. This is mainly because Rust prevents memory leaks, which means there would be fewer errors in the work. Rust also offers cross-platform use and faster compilation.

Rust, like many other programming languages, has its own software libraries. It also has a helpful compiler and is considered easy to read. This compiler is dual-licensed under Apache and MIT.



What is it used for?

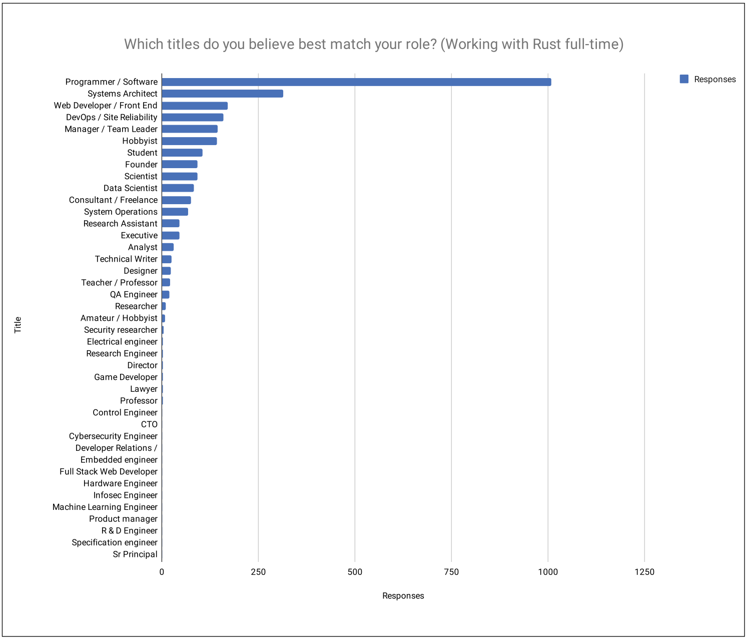
As Rust is also considered a rather low-level language, it really comes in handy when developers need to get more gains more out of the minimal resources they have. The language is statically typed and because of this, It helps you prevent certain types of bugs during compilation. As a result, developers will tend to use it when your resources are on the lower side, and in situations where is critical that your software runs smoothly. In contrast to this logic, high-level dynamically typed languages (for instance, Python and JavaScript) are considered to be better for tasks related to quick prototypes. Rust has many use cases like:

- Distributed online services.
- Powerful, cross-platform command-line tools.
- Systems programming, like browser engines and, perhaps, Linux kernel.
- Embedded devices.

Latest Rust Trends

Rust itself is an emerging trend in the programming world, even overtaking C+ in some instances. It is believed that Rust will continue to grow in popularity, and since the language has a robust community, we may see the language evolve quickly. Some of the reasons why Rust has grown as a trend are:

- Quick debugging and testing: The language itself is very fast and its known for supporting effective and quick debugging.
- Lower effort: The language is known for supporting very complex code, especially when compared to other languages. Developers can, therefore, achieve more in less code.
- Versatility: The language enables cross-platform development.
- Ease of integration: This language can be integrated with C, if needed, along with other programming languages.
- Safety: It is safer when compared to other programming languages.
- Community: There is a robust and excited community of developers who support this language and want to see it grow.



How to get started?

In order to start your journey with Rust, you must download the language from the official website or download the Rust Playground. The latter is an online tool that allows developers to run Rust code in order to practise coding and see how it turns out. After you have ensured that the Rust environment is ready, you can start writing code.

To create a new project, go to the directory you want the project to be in and select cargo new fizzbuzz. This will instruct Rust’s build manager to create a new project. Once you do that, go to the /src folder and open up main.rs. You can now write the code and run it to see how it works.

Basic Code

Here’s some basic code to practise:

Hello World

```
fn main() {
    // Statements here are executed when the compiled 
binary is called
    // Print text to the console
    println!(“Hello World!”);
}
```

Get the largest string

```
fn largest<’a>(str1:&’a str, str2:&’a str) -> &’a str {

if str1.len() > str2.len(){
str1
} else {str2}}
fn main() {
let str1 = “Hello”;
let str2 = “linuxhint”;
let result = largest(str1, str2);
println!(“Largest: {}”, result);
```

Primitives

```
fn main() {
    // Variables can be type annotated.
    let logical: bool = true;
    let a_float: f64 = 1.0;  // Regular annotation
    let an_integer   = 5i32; // Suffix annotation
    // Or a default will be used.
    let default_float   = 3.0; // `f64`
    let default_integer = 7;   // `i32`
        // A type can also be inferred from context 
    let mut inferred_type = 12; // Type i64 is inferred 
from another line
```

```
    inferred_type = 4294967296i64;

    // A mutable variable’s value can be changed.
    let mut mutable = 12; // Mutable `i32`
    mutable = 21;
    // Error! The type of a variable can’t be changed.
    mutable = true;
    // Variables can be overwritten with shadowing.
    let mutable = true;
}
```

Variable Bindings

```
fn main() {
    let an_integer = 1u32;
    let a_boolean = true;
    let unit = ();
    // copy `an_integer` into `copied_integer`
    let copied_integer = an_integer;
    println!(“An integer: {:?}”, copied_integer);
    println!(“A boolean: {:?}”, a_boolean);
    println!(“Meet the unit value: {:?}”, unit);
    // The compiler warns about unused variable bindings;
these warnings can
    // be silenced by prefixing the variable name with an
underscore
    let _unused_variable = 3u32;

    let noisy_unused_variable = 2u32;
    // FIXME ^ Prefix with an underscore to suppress the
warning
    // Please note that warnings may not be shown in a
browser
}
```

Casting

```
// Suppress all warnings from casts which overflow.
#![allow(overflowing_literals)]

fn main() {
    let decimal = 65.4321_f32;
    // Error! No implicit conversion
    let integer: u8 = decimal;
    // FIXME ^ Comment out this line

    // Explicit conversion
    let integer = decimal as u8;
    let character = integer as char;

    // Error! There are limitations in conversion rules.
    // A float cannot be directly converted to a char.
    let character = decimal as char;
    // FIXME ^ Comment out this line

    println!(“Casting: {} -> {} -> {}”, decimal, integer,
character);

    // when casting any value to an unsigned type, T,
    // T::MAX + 1 is added or subtracted until the value
    // fits into the new type

    // 1000 already fits in a u16
    println!(“1000 as a u16 is: {}”, 1000 as u16);

    // 1000 - 256 - 256 - 256 = 232
    // Under the hood, the first 8 least significant bits
(LSB) are kept,
    // while the rest towards the most significant bit (MSB)
get truncated.
    println!(“1000 as a u8 is : {}”, 1000 as u8);
    // -1 + 256 = 255
    println!(“  -1 as a u8 is : {}”, (-1i8) as u8);

    // For positive numbers, this is the same as the
modulus
    println!(“1000 mod 256 is : {}”, 1000 % 256);

    // When casting to a signed type, the (bitwise) result
is the same as
    // first casting to the corresponding unsigned type. If
the most significant
```

```
    // bit of that value is 1, then the value is negative.

    // Unless it already fits, of course.
    println!(“ 128 as a i16 is: {}”, 128 as i16);

    // 128 as i8 -> -128, whose two’s complement in eight 
bits is:
    println!(“ 128 as a i8 is : {}”, 128 as i8);

    // repeating the example above
    // 1000 as u8 -> 232
    println!(“1000 as a u8 is : {}”, 1000 as u8);
    // and the two’s complement of 232 is -24
    println!(“ 232 as a i8 is : {}”, 232 as i8);

    // Since Rust 1.45, the `as` keyword performs a *satu-
rating cast* 
    // when casting from float to int. If the floating point 
value exceeds 
    // the upper bound or is less than the lower bound, the 
returned value 
    // will be equal to the bound crossed.

    // 300.0 is 255
    println!(“300.0 is {}”, 300.0 _ f32 as u8);
    // -100.0 as u8 is 0
    println!(“-100.0 as u8 is {}”, -100.0 _ f32 as u8);
    // nan as u8 is 0
    println!(“nan as u8 is {}”, f32::NAN as u8);

    // This behavior incurs a small runtime cost and can 
be avoided 
    // with unsafe methods, however the results might over-
flow and 
    // return **unsound values**. Use these methods wisely:
    unsafe {
        // 300.0 is 44
        println!(“300.0 is {}”, 300.0 _ f32.to _ int _
unchecked::<u8>());
        // -100.0 as u8 is 156
        println!(“-100.0 as u8 is {}”, (-100.0 _ f32).to _ int _
unchecked::<u8>());
        // nan as u8 is 0
        println!(“nan as u8 is {}”, f32::NAN.to _ int _
unchecked::<u8>());
    }
}
```

Blocks

```
fn main() {
    let x = 5u32;
    let y = {
        let x _ squared = x * x;
        let x _ cube = x _ squared * x;
        // This expression will be assigned to `y`
        x _ cube + x _ squared + x
    };
    let z = {
        // The semicolon suppresses this expression and `()`
is assigned to `z`
        2 * x;
    };
    println!(“x is {:?}”, x);
    println!(“y is {:?}”, y);
    println!(“z is {:?}”, z);
}
```

CHAPTER #10

Perl

It's PERL, not PEARL!

Practical Extraction and Report Language (Perl) is basically a scripting language. Applications in Perl are also known as Perl scripts, and these scripts are generally more compact when you compare them with scripts written in other languages. One of the most common uses when it comes to Perl is extracting text file information as well as generating a report or other output format. It is important to note that this is an interpretive language as opposed to a compiled language such as C. Now most uncompiled programs need a lot more CPU run time. However, it is important to keep in mind Perl programs as they can actually save time for developers (as long as they have fast processors). Perl is used in many areas of web application development.

As far as the definition goes, this language is a high-level interpreted, general-purpose, and dynamic programming language. When it was first created, the language was initially created only for text handling and system

```
#!/usr/bin/perl
#This script assigns multiple values to multiple variable
#in one step
#Author: [illegible]
#Date: [illegible] 2019
($StudentName, $age, $BenchNumber) = ("Gaafar Mohamed", 19, 4356);
print "Student Name: $StudentName\n";
print ("Age: ", $age,"\n");
print("Bench Number: ",$BenchNumber,"\n");
```

management. However, the language was later given the ability to also handle regular expressions, network sockets, and so on. Currently, this language is known for its capabilities regarding handling Regex(Regular Expressions).

Features of Perl

There are many reasons for this language's popularity. Some of these are:

- Easy to start: This language is a high-level language. This means that it is similar to popular programming languages like C, and C++. As a result, it is much easier to learn, especially if you know these languages. .
- Text-Processing: You may have already guessed this from the full form of Perl, but it is mainly known for its ability to generate reports that are extracted from different text files easily.
- Great Features: Perl incorporates many of the features of popular languages such as C, sed, awk, and sh, etc. As a result, this language is considered to be useful and productive.
- System Administration: Perl makes it much easier to carry out tasks related to system administration. As a result, developers do not need to be dependent on numerous languages. Instead, they can just use Perl.
- Web and Perl: This language can easily be embedded into web servers to enhance the processing power. The language also comes with a DBI package, and this makes web-database integration simple.

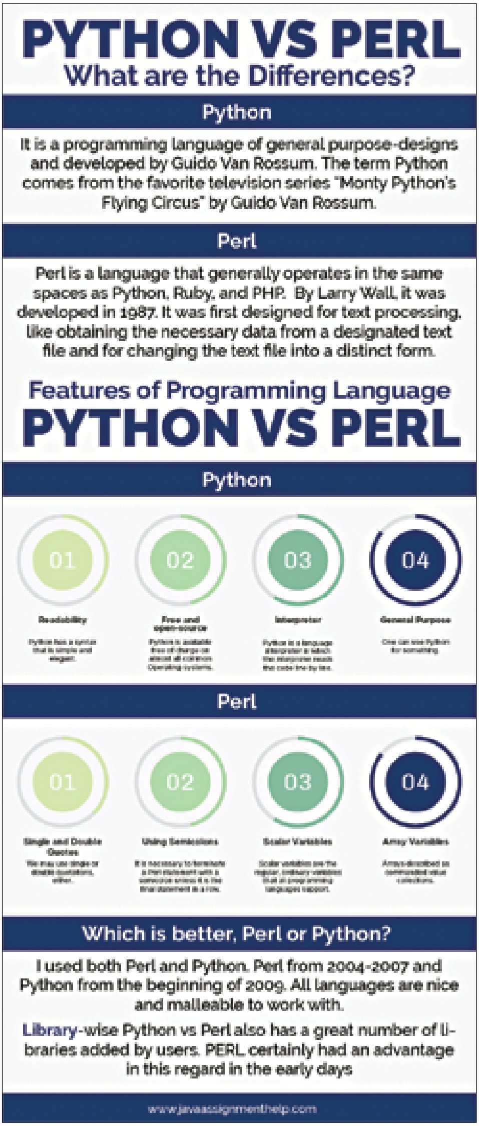
What is it used for?

Perl is an immensely popular language among programmers. This is because it has various applications, such as:

- When it was first introduced, most programmers leveraged this language to write write CGI scripts.
- This language is typically used as a connective tissue between dif-

ferent systems that are not similar or can inter-operate seamlessly due to operational differences.

- Most system administers rely heavily on this language. This is because they can simply put in a single command to reach a goal that would in other situations need a program to be written.
- Perl is easily portable as well, with some degree of personalisation allowed between Windows and macOS.
- Developers also use Perl to deploy and build different types of solutions.
- Perl is also used by many suppliers or software manufacturers for operations related to packaging and deploying the software commercially (including COTS and bespoke).
- Perl is also widely used in the field of bioinformatics and finance and bioinformatics as it can easily process and handle large volume data sets.



PYTHON VS PERL

What are the Differences?

Python

It is a programming language of general purpose-designs and developed by Guido Van Rossum. The term Python comes from the favorite television series "Monty Python's Flying Circus" by Guido Van Rossum.

Perl

Perl is a language that generally operates in the same spaces as Python, Ruby, and PHP. By Larry Wall, it was developed in 1987. It was first designed for text processing, like obtaining the necessary data from a designated text file and for changing the text file into a distinct form.

Features of Programming Language

PYTHON VS PERL

Python

01 Readability: Python has a syntax that is simple and elegant.

02 Free and open-source: Python is available free of charge on almost all common Operating systems.

03 Interpreter: Python is a language interpreter in which the interpreter reads the code line by line.

04 General Purpose: One can use Python for something.

Perl

01 Single and Double Quotes: We may use single or double quotations, either.

02 Using Semicolons: It is necessary to terminate a Perl statement with a semicolon unless it is the final statement in a row.

03 Scalar Variables: Scalar variables are the regular, ordinary variables that all programming languages support.

04 Array Variables: Arrays described as commanded value collections.

Which is better, Perl or Python?

I used both Perl and Python. Perl from 2004-2007 and Python from the beginning of 2009. All languages are nice and malleable to work with.

Library-wise Python vs Perl also has a great number of libraries added by users. PERL certainly had an advantage in this regard in the early days

www.javaassignmenthelp.com

How to get started?

Start by checking if Perl has already been installed in your computer. You can:

Head to Command line

- For Windows, search for cmd in the Run dialog (+ R),
- For Linux, open the terminal using Ctrl+Alt+T,
- For MacOS use Control+Option+Shift+T

Enter this command - perl -v

If the language is already installed, it will open. Otherwise you will get an error message indicating that there is something wrong with the command line.

If Perl is not already installed in your system, you must:

- Download Perl from perl.org
- Install the program by following the instructions displayed on the screen
- Use an online IDE to write or code, or open the Commandline to write your code
- Run your code

What’s has changed in Perl?

- Perl5 offers many different features that can help you support complex data structures. It also offers an object-oriented programming model and this model gives you many packages, directives, and references.
- Perl7 was announced in 2020, however, the language has not been released yet. Developers are looking forward to this update.

Basic Code

Here’s some basic code for you to run and practise!

Hello World

```
#!/usr/bin/perl

# Modules used
use strict;
use warnings;

# Print function
print(“Hello World\n”);
```

Addition

- `$a = 5;`
- `$b = 10;`
- `print $a + $b;`
- `Here Result will be 15`

Substraction

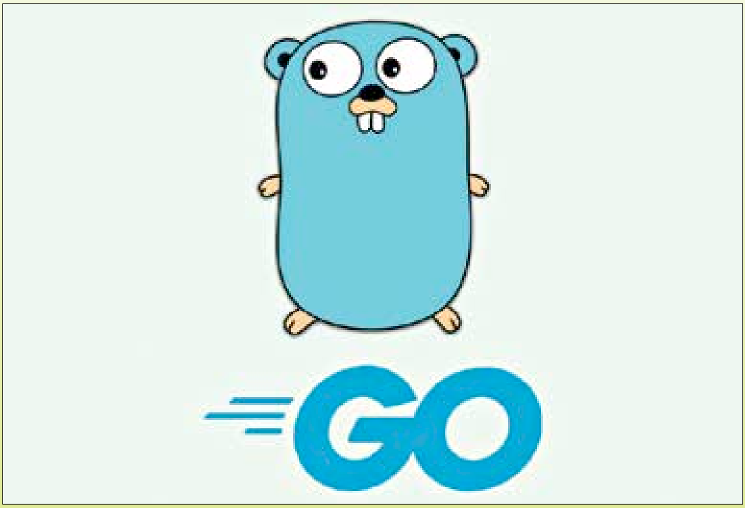
- `$a = 10;`
- `$b = 5;`
- `print $a - $b;`
- `Here Result will be 5`

Multiplication

- `$a = 5;`
- `$b = 10;`
- `print $a * $b;`
- `Here Result will be 50`

Division Operator

- `$a = 30;`
- `$b = 15;`
- `print $a / $b;`
- `Here Result will be 2`



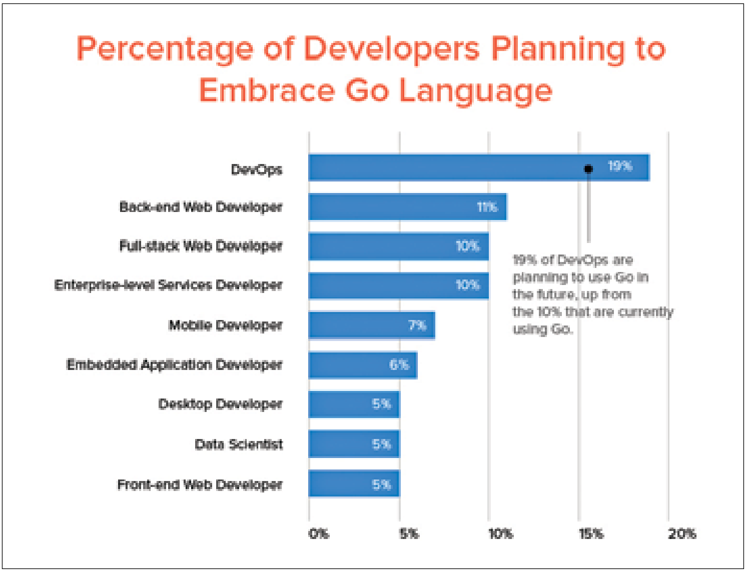
Go

Ready, set, GO...

Go (also known as golang) is the name of an open-source programming language that was developed at Google in 2007 by Robert Griesemer, Rob Pike, and Ken Thompson. Even though it has been developed very much in the same way as Algol or C, unlike some of the other languages, this particular language not dependent on C itself at all. If you're wondering whether you should learn Go, it is important to note that one of the defining feature of Go is concurrency. What this basically means is that many different processes can easily be executed simultaneously. This makes this language a highly efficient one. In addition to that, the language is also quite verbose and it displays extended information. Some of the other features to note are CSP-style programming, memory management, memory safety, and structural typing. Interestingly, this language started as an experiment. The main goal of this experiment was to address some of common criticisms pertaining to other programming languages while also making sure that the still retain their most useful features. Some of the objectives that people had in mind while creating Go were:

- The language should be able to scale to larger systems like C++ and Java
- The language itself should be dynamic and light and it should be able to adapt to changing requirements
- It should work with many supporting tools, but not actually be dependent on tools
- It should support networking and concurrency

Go was announced in November 2009, and it was been implemented in different places. The compiler of this language is known as gc and it has been created to be an open-source software. The language can be used on different platforms like Unix, OS X, Windows, BSD and Linux. Since 2015, this language also became compatible with mobile devices. Go offers fast compilation. This is a language that can enhance efficiency and even help with remote package management.



What is it used for?

Go has many different applications, such as:

1. Distributed Network Services

Concurrency works best in a networking environment. By leverages channels and goroutines, Go is highly suited for developing network services. These can include web servers, APIs, and mini frameworks that are designed for web applications.

2 .Cloud-Native Development

The networking and concurrency features that are associated high portability across various platforms are what have established Go as a great option for cloud computing. In fact, there are many different stand-out cloud platforms such as Kubernetes that have been built with Go. Go is also used by Google Cloud to enhance performance and scalability.

3. Replacements for Existing Infrastructure

Infrastructures that have become outdated or old can be rewritten by using Go in order to modernise anything that has grown obsolete over time. Go has powered the new version of the Network Time Protocol (NTP), which is basically an internet protocol designed for clock synchronization among networks

4. Utilities and Stand-Alone Tools

Go is a language that is fully functional and has little to no dependencies. The language is therefore considered to be well-suited for building small tooling items. Such tooling items can be deployed quickly and also be packaged for redistribution quickly and easily.

5. News Outlets

Go has a great capacity for multithreading This helps by maximusing CPU usage, which is essential for managing multimedia websites. In fact, known news outlets such as the BBC also use Go.

6. Media Platforms

One of the biggest advantages of Go is that it offers capable data processing. As a result, many media platforms such as Netflix, Youtube, and Soundcloud all use Go.

Latest Go Trends

Certain frameworks in Go are trending amongst developers. These include:

1. Gin

The gin framework is considered to be right at the top of the list of Go frameworks. It is an extremely popular framework as it has a minimalist framework and performance. This particular framework is basically used for building a REST API for the backend in a situation where the programmer aspires to create a single-page application by leveraging a frontend framework.

2. Beego

The beego framework is also a very popular one. This framework is mainly used for the rapid development of REST APIs, backend services, and web applications in Golang. This particular framework is considered to be similar to the Django web framework in Python. It also offers some specific features of Golang like struct embedding and interfaces.

3. Echo

The echo framework is ideal for developers looking for a minimalist, high-performance, and extensible framework in Go. The framework itself is a highly optimized HTTP router that has zero dynamic memory allocation and smartly prioritises routes. This framework is mainly used is to create robust and scalable REST APIs, and these APIs can easily be collected into groups.

4. Kit

The kit framework is a programming toolkit that is typically used to create robust, maintainable and reliable microservices in Golang. This framework contains a set of best practises and packages. These offer a trustworthy, comprehensive, and robust, means of creating microservices for enterprises of any size. Go is a great general-purpose language. But, having said that, it is essential to understand that these microservices need a certain amount of specialised support.

How to get started?

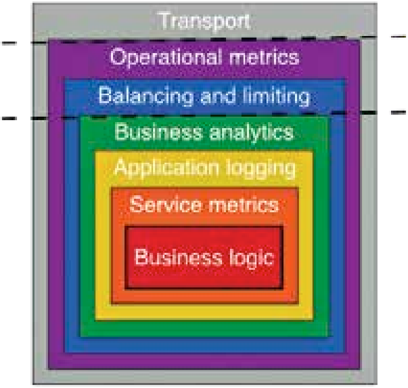
1. Download and run the installer for your platform from golang.org/dl.
2. To confirm everything is working, open a terminal and type the following:go version
3. You should see the following : go version go1.5 windows/amd64
4. If the system displays an error saying that the command is not recognised, try restarting your computer.
5. The Go toolset is made up of several different commands and subcom-

mands. You can pull up a list of those commands by typing: go help

6. Now you can write your Go program.

What's Changed in Go?

Go 1.7 was recently released. This updated has added support for the 64-bit ARM architecture on Windows (the windows/arm64 port). This port also supports cgo.



Basic Code

Here's a code for you to practise:

Hello World

```
$ go build hello-world.go
$ ls
hello-world    hello-world.go $ ./hello-world
hello world
c Code
```

What's More?

Well, there are some things left...

Start with Detective Work

Understanding which language you should pick can be a challenging task. After all, the choice is going to impact your career and your earnings. A great way to start is by learning as much as you can about the world of coding, as well as the new developments in different programming languages.

Many aspiring developers tend to believe that choosing the 'wrong' language will have catastrophic results. However, the truth is that there is no such thing as the wrong language. Programming languages can seem fairly different when you juts glance through them, but the fact of the matter is that they have a lot in common. Many different languages have similar structures and patterns. This means that no matter what language you decide to start your coding journey with, you'll still be working with integral key coding concepts. These concepts will be handy in the future as they will help you learn other languages. In other words, once you've learned one language, it starts becoming easier to learn others. Also, it is important to note that many developers and programmers tend to use different languages at different phases of their careers. The reason for this is that they are typically required to solve different types of problems that require certain languages for optimal solutions. This means that the first language you choose isn't one that will follow you for the rest of your career. With that in mind, you must understand that you need to get out of the mindset of believing that certain coding languages are better than others. Of course, some are more in demand than others, but if your goal is to learn multiple languages, then you can start anywhere. You aim should be to gain that foundational knowledge with the language you have chosen and then use that knowledge to grow.

Observe

A great way to understand more about the world of coding is by looking at the work done by developers before you. Let's say you want to create a social media website and you really like the way Facebook functions. It would be greatly beneficial to go through the code and see how different features are implemented, the kind of impact certain scripts have on the outcome, and the way the final product informs consumer behaviour.

This is something that you can do anytime. A great approach would be to download a compiler associated with the language you want to practise, and then to write script associated with websites or applications that you like. You can try to recreate the website in your spare time to see whether you have a hang of the language. You can even use projects like this to make improvements to the website basd on your own user experience, and then see how that works out. It's a great way to learn how to create different solutions for clients, while also practising the language in a low-stakes environment.

As a pro tip - save this code. You can use it as a portfolio project to show employers the level of dedication you have. They can evaluate your work and who knows, they may be impressed enough to give you a great job!

Open it up

As a beginner, you may find that working on open source projects can be a fantastic method to improve your skills.

There are many benefits to doing so, and you can end up:

1. Improving software you rely on

Many open source contributors begin their journey as users of software. As a user, you may come across a bug in the software. Why not try patching it yourself? If you are able to do so, you can contribute the patch to the project.

2. Improving existing skills

No matter what skill you want to improve, you'll be able to find something on an open source project platform. Many people learn a lot at such places and the community spirit is great.

3. Finding mentors

Working with others on a shared project can be a great experience for a beginner. You'll have to learn from those more experienced and may even end up building great relationships.

4. Growing your reputation

Open source work is public.You'll be able to show off your skills and grow your reputation.

5. Learning professional skills

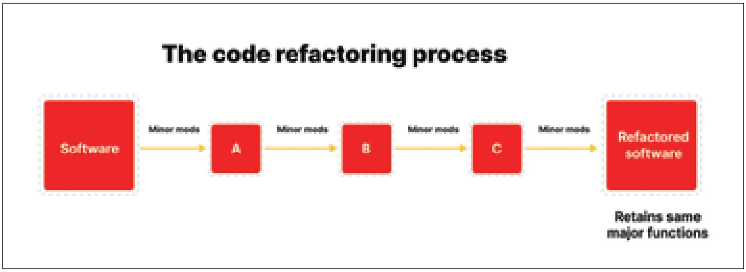
Contributing to open source projects can help you improve professional skills such as time management, conflict resolution, leadership skils and so on. Some of the top websites that offer open source projects include:

- GitHub Explore
- CodeTriage
- Open Source Friday
- SourceSort
- First Timers Only
- First Contributions
- Up For Grabs
- 24 Pull Requests

Refactor your code

Once you've written your code, it is essential to refactor it. Here are some tips and tricks that can help you:

- Always carry out code refactoring in basic steps. In other words, you should always make tiny changes in your program to prevent larger issues from cropping up. Make sure that all the tiny changes improve your code while also ensuring the application can run properly.
- After you make any small change, you should always run the test TDD and CI. If you don't do this, you might end up introducing bugs in the system.
- You should not create any features or functionality when you are carrying out the refactoring process. It is always better to refactor the code prior to adding the updates or new features as they can cause issues with your existing code.
- You should definitely get your testing team and QA teams involved in refactoring. This is because the process itself can impact test outcomes.
- As someone who is passionate about what you do you should remember that you might never be satisfied with your code. After a certain point,



even the refactored code will become outdated in near future and you might feel like you need to refactor it again.

Content is King!

Understanding the industry you're vying to join is crucial for any professional, especially when the industry is as dynamic as developing and programming. Trends keep changing, companies have new expectations, and customers outgrow older applications and what to turn to new ones that are more exciting and packed with new features. It is essential to always keep your finger on the pulse so that you are aware of the different changes that may be occurring in the industry.

There are plent of websites available for you to browse daily and get new information. Some website also post daily challenges so that you can improve your skills. Some of these include:

- CodeChef
- Codewars
- CodinGame
- Geektastic
- HackerRank
- LeetCode
- Project Euler.
- SPOJ
- TopCoder
- Exercism.io
- Codebyte

It is also important to learn to lean on books, even in this digital age. The information you may find on the internet may be compressed, or written with an assumption that the readers already know a little about coding or are studying it in school. You may miss out on key fundamentals by only relying on digital content. Here are some of the best books for coders to learn:

Top 10 Books That Every Programmer Must Read Once

Last Updated : 28 Feb, 2022

Read

Discuss

If we find a person with a rare intellect, we should ask him about the books he reads. A book can define a person's nature and intelligence. If you are a coder, you should be also a good reader because it develops the mind and the mind is your weapon. you have to train it daily. Before finding any solution to a problem Reading and understanding the problems is the most important if you are a coder or any problem solver. Without good reading and comprehension skills, a person takes more time to understand the problems before solving it.

Top-10-Books-That-Every-Programmer-Must-Read-Once

Below there is the list of top 10 books every programmer should read. These books not only teach the syntax and semantics of programming languages but also help you to think, organize, and become a good problem solver, which is indeed the most important lesson for a coder. These books are not only for mastering a particular programming language like C++, Java or Python but will help you to become a Better Programmer.

1. Clean Code: By Robert C. Martin
2. The Mythical Man-month: By Frederick Brooks
3. The Pragmatic Programmer: Your Journey to Mastery
4. Code Complete (2 Edition): By Steve McConnell
5. The Art of Computer Programming
6. Programming Pearls
7. Code: Charles Petzold
8. Introduction to Algorithms
9. Refactoring: Improving the Design of Existing Code
10. Design Patterns: Elements of Reusable Object-Oriented Software

Executing the Code

Now that you've read up on all the different programming languages, and how you can get started, it's time to embark on your journey as a developer. Remember, no matter what language you choose it is essential to have clarity about what you want to achieve in your life. Programming languages can be used to build the best video games, software, applications, websites, and platforms and it's up to you how you use your knowledge to change the world!

Of course, nothing is as easy as just wishing it. You must work hard and understand the ins and outs of your chosen language to be the best in your field.

To follow the latest in tech,
follow us on...

facebook.com/digitgeek

digit.in/facebook

RNI REGN. NO. DELENG/2001/03791