

## Digital Marketer – Principles of Coding

Apprentices should be able to demonstrate knowledge and understands the principles of coding. Key minimum requirements are:

- Understands and has an appreciation of logic
- Understands, has an awareness of and recognises software languages
- Understands the compatibility of code on different platforms
- Understands the components involved to make the Web work

The format for the examination is a 30-minute multiple-choice examination consisting of 20 questions. The examination is closed book (*no materials can be taken into the examination room*). The pass mark is 13/20 (65%).

-----

### 1. [Appreciation of Logic](#) (10%, K2)

In this topic, learners will understand and develop an appreciation of logic. The successful Apprentice should be able to:

1.1 Understand the basics of logic in computation and of logic gates.

- And
- Or
- Not

### 2. [Programming Languages](#) (10%, K1)

In this topic, learners will become aware of programming languages and how they apply in building digital products. The successful Apprentice should be able to:

2.1 Identify the key characteristics and applications of the following programming languages:

- Hypertext Markup Language (HTML)
- JavaScript (JS)
- Java

### 3. [Interaction and Compatibility of Code on Different Platforms](#) (25%, K2)

In this topic, learners will gain an understanding of code compatibility on different platforms. The successful Apprentice should be able to:

3.1 Memorise the LAMP (Linux, Apache, MySQL, and PHP) and XAMPP stack. Describe the associated code compatibility with using alternative proprietary web stacks.

3.2 Discuss how the following file formats that can be shared across multiple digital platforms and issues that arise around compatibility:

- PDF
- HTML
- Image (GIF, JPG, PNG)
- Video; Mpeg
- Audio; MP3

3.3 Describe each stage required to generate or commission code. What considerations will be required to ensure code capability across multiple devices and the associated infrastructure limitations.

- Social media platforms feeds (called widgets) used on a new digital solution.
- Creating and protecting feeds (using API keys) for use by other organisations.

#### 4. Web components (55%, K2)

In this topic, learners will understand the components involved to make the Web work. The successful Apprentice should be able to:

4.1 Define the terminology for the following key internet protocols that enable the web to work:

- Hypertext Transfer Protocol (HTTP)
- Hypertext Transfer Protocol Secure (HTTPS)
- Transport Layer Security and Secure Sockets Layer (TLS / SSL)

4.2 Discuss the purpose of the following:

- Web and application server
- Hosting and serving
- Relational database management systems
- Content management systems

4.3 Describe the purpose of a web client; browsers and applications.

4.4 Describe how Search Engines operate in regard to the following:

How mark-up languages render hyperlinks.

- How the web crawler work.
- Displaying of search results.
- Factors that affect search engine optimization (SEO).

4.5 Explain the differences between a static and dynamic website.

- Written in code
- Written scripting language

4.6 Describe how local (cookies) or session data storage is utilised to share information for standard digital features.

- forms
- checkout
- registration

4.7 Identify the key roles of the following Web technologies governance groups.

- World Wide Web Consortium (W3C);
- Internet Engineering Task Force (IETF).

## Appreciation of Logic

Boolean algebra and truth tables can be used to describe logical expressions. Logic gates are an elementary building block of any digital circuits. It takes one or two inputs and produces output based on those inputs. Outputs may be high (1) or low (0).

In a computer, most of the electronic circuits are made up logic gates. Logic is about how we combine the values true and false to produce another value, that is also true or false. In a binary world we use one and zero in place of true or false.

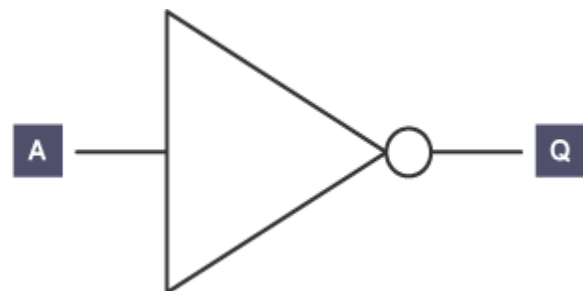
### NOT logic gate

NOT acts as an inverter. It takes only one input. If the input is 1, it will invert the result as 0 and vice-versa.

If A is the input and Q is the output, the truth table looks like this:

| A | Q |
|---|---|
| 1 | 0 |
| 0 | 1 |

The Boolean expression is written as  $Q = \text{NOT } A$



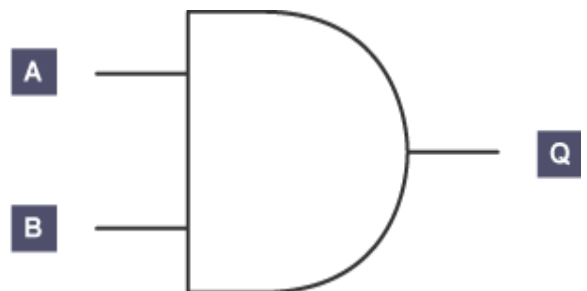
### AND logic gate

An AND gate can be used on a gate with two inputs. AND tells us that both inputs have to be 1 in order for the output to be 1.

If A and B are the input and Q is the output, the truth table looks like this:

| A | B | Q |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

The Boolean expression is written as  $Q = A \text{ AND } B$



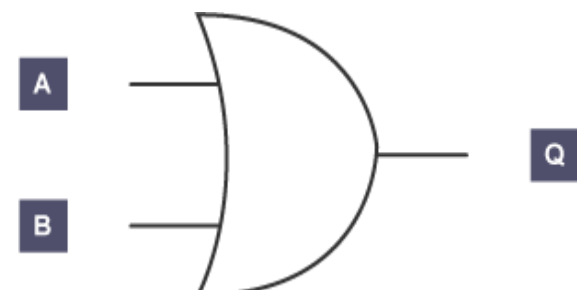
### OR logic gate

The OR gate has two inputs. One or both inputs must be 1 to output 1, otherwise it outputs 0.

The truth table would look like this:

| A | B | Q |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

The Boolean expression is written as  $Q = A \text{ OR } B$



Read the full article on Google Classrooms to see how we apply this in search refinement:  
[Principles of Coding: Platforms & Protocols - Basics of logic in computation & logic gates.](#)

## Programming Languages

Identify key characteristics and applications.

- **HTML**

HTML (the Hypertext Markup Language) and CSS (Cascading Style Sheets) are two of the core technologies for building Web pages. HTML provides the structure of the page, CSS the (visual and aural) layout, for a variety of devices.

HTML is the language for describing the structure of Web pages. HTML gives authors the means to:

- Publish online documents with headings, text, tables, lists, photos, etc.
- Retrieve online information via hypertext links, at the click of a button.
- Design forms for conducting transactions with remote services, for use in searching for information, making reservations, ordering products, etc.
- Include spreadsheets, video clips, sound clips, and other applications directly in their documents.

With HTML, authors describe the structure of pages using markup. The elements of the language label pieces of content such as “paragraph,” “list,” “table,” and so on.

**HTML makes static websites with basic functionality, but they would not be considered ‘dynamic’.**

- **JavaScript**

While the most common scripting language ECMAScript (more widely known as **JavaScript**) is developed by ECMA, a great many of the APIs made available in browsers have been [defined at W3C](#).

A script is program code that doesn’t need pre-processing (e.g. compiling) before being run. In the context of a Web browser, scripting usually refers to program code written in JavaScript that is executed by the browser when a page is downloaded, or in response to an event triggered by the user. **JavaScript powers the dynamic behaviour on most websites.**

- **Java**

Java is an object-oriented language similar to C++, but with advanced and simplified features. Java is free to access (it’s Open Source) and can run on all platforms. It is fast, secure, and reliable.

Java code is compiled into intermediate format, which can be executed on any systems for which the Java Virtual Machine is ported. That means you can **write a Java program once and run it on Windows, Mac, Linux or Solaris** without re-compiling. Hence the slogan “*Write once, run anywhere*” of Java.

The Java platform is designed with security features built into the language and runtime system such as static type-checking at compile time and runtime checking (security manager), which let you create applications that can’t be invaded from outside. **Java provides compatibility of code across different platforms.**

## Interaction and Compatibility of Code on Different Platforms

3.1 Memorise the LAMP (Linux, Apache, MySQL, and PHP) and XAMPP stack. LAMP stands **for Linux | Apache | MySQL | PHP**. Together, they provide a proven set of software components for delivering high-performance web applications. These components, arranged in layers support one another, making up the software stack.

The four traditional layers of a LAMP stack consist of free and open-source products. Linux, Apache, MySQL and PHP are the cornerstone of a free, non-proprietary LAMP stack. There are several variants of the four-stack model as well. These variants use alternative software replacing one or more of the traditional components.

**XAMPP: Any operating system | Apache | MariaDB (or MySQL) | PHP or Perl**

Read the full article on Google Classrooms: [Principles of Coding: Platforms & Protocols - LAMP](#).

3.3 Describe each stage required to generate or commission code. What considerations will be required to ensure code capability across multiple devices.

Responsive design is a core part of website development and code compatibility is a part of this. Code operates slightly differently on different browsers and browser versions. Want to test your website across multiple browsers - try BrowserStack:

<https://www.browserstack.com/>

Testing development code ensures compatibility and there are plenty of options to check this such as W3C Markup Validation Service:

<https://validator.w3.org/> and CodePen: <https://codepen.io/>

Ensure that you have your viewport set properly – this defines the area of the screen that the browser can render content to.

1. [https://www.w3schools.com/css/css\\_rwd\\_viewport.asp](https://www.w3schools.com/css/css_rwd_viewport.asp)

2. <https://www.uxmatters.com/mt/archives/2013/02/how-do-users-really-hold-mobile-devices.php> (not revision, but good read)



Adding the meta viewport tag to the head element tells the browser we know about resizing / responsive requirements

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

This is done for you in WYSIWYG systems like WordPress / Squarespace.

[https://developer.mozilla.org/en-US/docs/Mozilla/Mobile/Viewport\\_meta\\_tag](https://developer.mozilla.org/en-US/docs/Mozilla/Mobile/Viewport_meta_tag)

In coding you should use relative positions rather than absolute position to ensure that your website flexes across the wide range of different device sizes. So don't set an image to be '1000px', but set the width to be '100%'. This way the image will scale to be 100% of the available viewport. CSS example below:

`img {max-width: 100%;}` – CSS covers images

`img, embed, object, video {max-width: 100%;}` – CSS covers all images, videos, embedded items and objects

[https://www.youtube.com/watch?time\\_continue=31&v=F-akKxve2oE&feature=emb\\_logo](https://www.youtube.com/watch?time_continue=31&v=F-akKxve2oE&feature=emb_logo)

"max-width" actually overrides the "width" so you can input the actual size too.

Added information:

[https://www.youtube.com/watch?time\\_continue=70&v=sFzdU8Z1Wd8&feature=emb\\_logo](https://www.youtube.com/watch?time_continue=70&v=sFzdU8Z1Wd8&feature=emb_logo)

[https://www.youtube.com/watch?time\\_continue=95&v=AmHW9Hhm5ml&feature=emb\\_logo](https://www.youtube.com/watch?time_continue=95&v=AmHW9Hhm5ml&feature=emb_logo)

## Web components

Forming a massive 55% of your knowledge and understanding – this is the key area to revise. All the information you require is located in Google Classrooms:

[Principles of Coding: Platforms & Protocols](#)

[Principles of Coding: Search & Cookies](#)

Identify the key roles of the following Web technologies governance groups – see the [glossary](#) below.

## Glossary

|                                                                           |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|---------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>ECMA<br/>(European<br/>Computer<br/>Manufacturers<br/>Association)</b> | <p>Ecma is a standards organization for information and communication systems. It acquired its current name in 1994, when the European Computer Manufacturers Association (ECMA) changed its name to reflect the organization's global reach and activities.</p> <p>ECMAScript is a scripting-language specification standardized by Ecma International in ECMA-262 and ISO/IEC 16262. It was created to standardize JavaScript to help foster multiple independent implementations.</p>                                                                                                                                                                                                                                                               |
| <b>IETF (Internet<br/>Engineering<br/>Task Force)</b>                     | <p>The Internet Engineering Task Force (IETF) is an open standards organization, which develops and promotes voluntary Internet standards, in particular the standards that comprise the Internet protocol suite.</p> <p>The mission of the IETF is to make the Internet work better by producing high quality, relevant technical documents that influence the way people design, use, and manage the Internet.</p> <p>Some would say that the IETF looks after the architecture / structure of the Internet:</p> <ul style="list-style-type: none"><li>*The ART Area develops application protocols and architectures in the IETF.</li><li>*The Internet Area is responsible for specifying how IP will run over new link layer protocols.</li></ul> |
| <b>W3C (World<br/>Wide Web<br/>Consortium)</b>                            | <p>The World Wide Web Consortium (W3C) is an international community that develops open standards to ensure the long-term growth of the Web.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |