CODING: AS A PROFESSIONAL ACCOUNTANT, WHY YOU SHOULD BE INTERESTED

About ACCA

ACCA (the Association of Chartered Certified Accountants) is the global professional body for professional accountants.

We're a thriving global community of 233,000 members and 536,000 future members based in 178 countries and regions, who work across a wide range of sectors and industries. We uphold the highest professional and ethical values.

We offer everyone everywhere the opportunity to experience a rewarding career in accountancy, finance and management. Our qualifications and learning opportunities develop strategic business leaders, forward-thinking professionals with the financial, business and digital expertise essential for the creation of sustainable organisations and flourishing societies.

Since 1904, being a force for public good has been embedded in our purpose. We believe that accountancy is a cornerstone profession of society and is vital helping economies, organisations and individuals to grow and prosper. It does this by creating robust trusted financial and business management, combating corruption, ensuring organisations are managed ethically, driving sustainability, and providing rewarding career opportunities.

And through our cutting-edge research, we lead the profession by answering today's questions and preparing for the future. We're a not-for-profit organisation.

Find out more at www.accaglobal.com

Contents

1. **Setting** the **scene**

ACCA offers a progressive forward-thinking accountancy qualification. Digital skills are vital to being prepared for the future and this paper explores one important aspect of this, namely coding.
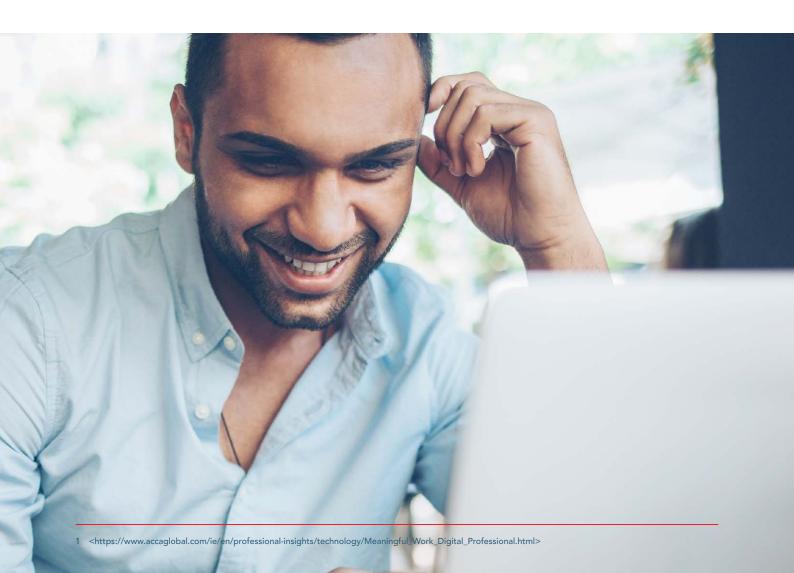
Coding refers to creating rules-based instructions that can be understood by a software program. The aim of coding is often to automate a task or analysis for efficiency, particularly if this needs to be done periodically and is a drain on human time and resources. It may also drive higher quality business insights particularly when dealing with large volumes of data that get periodically updated.

Research[1] among accountancy and finance professionals examining ten families of technologies has shown that digital adoption is expected to rise from 25% in 2020 to 87% by the end of the decade. It also highlights the varied nature of tasks performed by professional accountants with technology.

It sets out six clusters of tasks where accountants use technology including checking and validating information, transforming the delivery model, developing solutions using technology, looking ahead for strategic business insights, supporting existing workflows and specialist technical advice such as tax treatment for cryptocurrencies.

In reference to these clusters, coding is a task most associated with developing solutions using technology. In other words, it's one of the many ways by which professional accountants can leverage digital to enhance the value they bring.

Purpose-led careers look different for everyone. Increasing digital adoption, further reinforced by remote working in the pandemic, is an undeniable fact. Not every professional accountant may need to code, but even a basic understanding can add value to their organisations, help to differentiate themselves and open future career opportunities.

---

1   <https://www.accaglobal.com/ie/en/professional-insights/technology/Meaningful_Work_Digital_Professional.html>
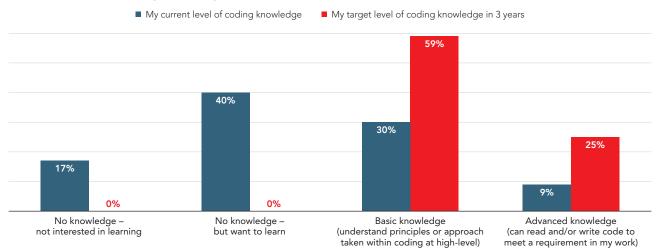
2. The **basics**

Though coding is an often-used word there can be variations in how well it is understood, as a survey of 992 ACCA members in November 2020 illustrates (Figure 1). Although 57% of respondents had no knowledge of coding 40% expressed an interest in learning. Looking ahead 3 years, none of the respondents wanted to remain without any knowledge of coding. And over that time frame, 6 in 10 wanted to develop a basic knowledge, and a further 1 in 4 sought advanced knowledge in coding.

Coding is done using a programming language, of which there are many types. These languages are on a spectrum of human readability with low-level languages being closer to machine code (language of 0s and 1s) and high-level languages being closer to human readable code, ie in natural language. It is the latter that is of relevance here.

Even within high-level programming languages there is a spectrum. Third generation languages like C, C++ and Java are extensively used in back-end application development and are popular with IT and programming specialists.

Of particular interest are fourth generation programming languages like Python where the code is expressed in a very intuitive way similar almost to writing a sentence in natural language. These types of languages can lend themselves to use cases like data analysis, data visualisation and scripts for customised reports to reduce low-value repetitive tasks, which are relevant for accountancy and finance professionals[2].

ALTHOUGH **57%** OF RESPONDENTS HAD NO KNOWLEDGE OF CODING **40%** EXPRESSED AN INTEREST IN LEARNING.

FIGURE 1: Level of coding knowledge

Legend: ■ My current level of coding knowledge ■ My target level of coding knowledge in 3 years

- No knowledge – not interested in learning: 17% / 0%
- No knowledge – but want to learn: 40% / 0%
- Basic knowledge (understand principles or approach taken within coding at high-level): 30% / 59%
- Advanced knowledge (can read and/or write code to meet a requirement in my work): 9% / 25%

2   <https://towardsdatascience.com/why-accountants-are-great-fit-for-programming-6033927047e6>

3. **Benefits** and **barriers**

Survey respondents saw a range of benefits in learning to code (Figure 2). A better understanding of the approach to data was a key factor (88% of respondents), ie how it is organised, analysed, and flows through the organisation. And it was seen to assist with having more informed strategic conversations on technology use and adoption due to a better understanding of what needs to happen in the background. A superior risk lens was mentioned as a benefit in terms of asking the right questions, or better questions, when using information produced by systems (87%).

There is a wider trend of more partnering between accountants and other professionals, eg multi-skilled audit teams with experts in data science or cyber.

Better understanding of coding is seen as assisting communication between accountants and those trained in other disciplines (86%). Coding is also seen as a valuable skillset that improves market value and career opportunities (85%).
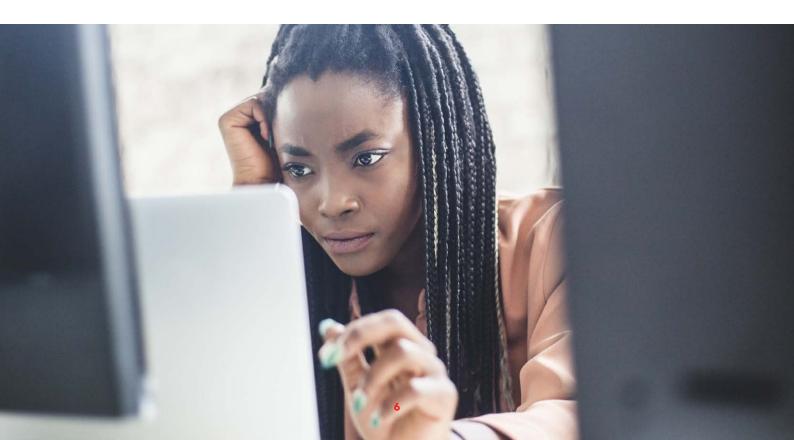
Despite these benefits from learning to code there were also some perceived barriers to building skills in this area (figure 3). The most often cited one was simply the lack of time (84%). This is perhaps not surprising given that many may not need coding immediately in the day-job, so it gets pushed back in the queue of priorities. As Figure 1 earlier showed, a significant proportion of 2 in 5 have no knowledge but want to learn. Converting intention to action will require making time for it.

FIGURE 2: Benefits of developing expertise in coding

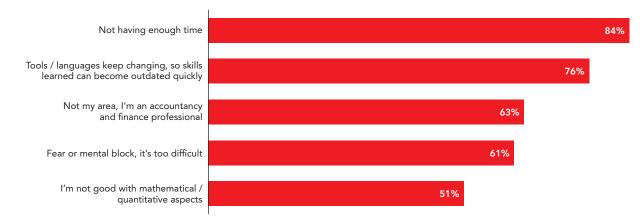|	
Better understanding of how data is organised, analysed and flows through my organisation	88%
Strategic conversations on technology use / adoption (due to better understanding of detail)	88%
Validating or interrogating information provided by a model / system / software output	87%
Explaining requirements to IT / Data Scientists in a way that they understand	86%
'In-demand' skillset, improves my market value	85%

The next big barrier was a perception that tools and programming languages keep changing so skills learned can get outdated quickly (76%). While at face value there is some validity in this observation, it doesn't recognise the distinction between logic and syntax. What is useful for accountants is to acquaint themselves with a programmatic way of thinking. Ie to understand the logic of how coding works, how decision-tree type analysis (if x happens then y) may be expressed in code and how that relates to business decision making or process flows. This is a more conceptual skill that is independent of the specific programming language being used.

Syntax on the other hand, refers to the specifics of how code instructions are written and can vary from one language to the next. It's needed to some extent to engage with a programming language, but learning the syntax is not the big-picture prize. Languages change and evolve, and while one language is popular today, it may be a different one in 5 years. So, the transferable skill here is understanding the logical constructs.

Other barriers mentioned related to a view that learning to code is 'out of scope' for an accountant (63%) or there is a reaction of fear (61%) driven by a view that it's just too hard. ACCA research on the emotional quotient[3] highlighted the pivotal role of the growth mindset. This is about exploring new areas, overcoming the fear of the unknown and adopting a learning mindset for success.

A self-perception of not being good with mathematical/quantitative concepts, though mentioned by just over half of respondents, was in fact far lower down the list compared to other factors. Furthermore, programming is becoming more user-friendly. The rise of low code/no code[4] platforms makes programming much more accessible and allows the ability to engage with intuitive interfaces like drag-and-drop menus to build code. This is increasingly giving rise to the citizen developer[5] where users can directly build simple applications in an agile way as business needs evolve without having to rely on a central IT function for everything which creates bottlenecks.

FIGURE 3: Barriers to developing expertise in coding

Not having enough time	84%
Tools / languages keep changing, so skills learned can become outdated quickly	76%
Not my area, I'm an accountancy and finance professional	63%
Fear or mental block, it's too difficult	61%
I'm not good with mathematical / quantitative aspects	51%

ACCA RESEARCH ON THE EMOTIONAL QUOTIENT[3] HIGHLIGHTED THE PIVOTAL ROLE OF THE GROWTH MINDSET. THIS IS ABOUT EXPLORING NEW AREAS, OVERCOMING THE FEAR OF THE UNKNOWN AND ADOPTING A LEARNING MINDSET FOR SUCCESS.

3 <https://www.accaglobal.com/uk/en/professional-insights/pro-accountants-the-future/emotional-quotient-in-a-digital-age.html>
4 <https://www.gartner.com/en/newsroom/press-releases/2021-02-15-gartner-forecasts-worldwide-low-code-development-technologies-market-to-grow-23-percent-in-2021>
5 <https://www.microsoft.com/en-us/itshowcase/citizen-developers-use-microsoft-power-apps-to-build-an-intelligent-launch-assistant>

4. Clean code

Code may need to be refreshed as circumstances change. Or it may be read and deployed by multiple individuals who had nothing to do with writing the original code. Indeed, the person who wrote the code may even have subsequently left the organisation. So, it needs to be well-constructed and usable clean code[6]. An asset that is human readable, avoids repetition, reduces complexity, and comes with documentation that clearly advises iterations and changes made.

As we look ahead to a world coming to grips with the impacts of climate change, another feature of increasing relevance is to avoid complex code creating wastage of resources. While this may seem less relevant to the professional accountant at an individual level, when aggregated globally, inefficient code is a drag on global computing resources. Circuitous logic or unnecessarily repeating loops in the code create electricity (and hence energy) consumption that doesn't produce any additional insight. This is about striking the right balance between code that is quick to develop, and one that is produced in a thoughtful way for long-term value.

AN ASSET THAT IS HUMAN READABLE, AVOIDS REPETITION, REDUCES COMPLEXITY, AND COMES WITH DOCUMENTATION THAT CLEARLY ADVISES ITERATIONS AND CHANGES MADE.

6   <https://medium.com/swlh/the-must-know-clean-code-principles-1371a14a2e75>

5. **Coding** in **action**

(Edited with permission from inputs by Andrew Chong, ACCA Member and Senior Data Scientist, EY)

For coding in a commercial environment, practitioners must differentiate between 'hacking up' a fun project because you can; and delivering a product that adds value to existing processes and services. Professional accounting and consultancy firms, small, medium, and large, need targeted innovations to produce the highest return with as little disruption to the business as possible.

Innovation adoption can be slower for larger firms to begin with; but, once a concept is validated, there can be a quicker pivot given their resources and scale. Here we explore the role of practitioners who leverage coding within the internal innovation process, the internal entrepreneur, or Intrapreneur.

The framework set out below draws from Technology Readiness Levels (TRLs) for estimating the maturity of technologies developed at NASA in the 1970s. It enables consistent, uniform discussions of technical maturity across different types of technology[7]. We use Intrapreneurship Readiness Levels (IRL), which breaks the process into three stages of Design Thinking, Incubation and Ramp-Up/Go-Live (Figure 4).

We illustrate an example to set the coding skillset into context. This section is based on the experience of a hypothetical persona, who we name Alice, and who works in Assurance. She has imagined a new automated process for detecting anomalous/uncommon entries in the financial statements of clients. This is her journey of bring this product to life within her accounting practice. The purpose is to give a broad flavour based on common conceptual elements which are likely to be relevant across organisations.

A. Design Thinking

This is about the utility and feasibility of the idea. How well does it address an existing problem and how easily can a solution be developed? Design thinking principles guide Alice in approaching the endeavour by visualising the end-result and then reverse-engineering the requirements.

i. **Enablement:** Alice's journey started just with curiosity to learn and a dedication to upskilling. A few Ted Talks snowballed into the purchase of an e-learning course in Python basics and AI applications in the financial domain. She learnt methods for processing textual data and visualisations for assessing variations in data points. From here, an idea was formulated.

ii. **Problem and Solution:** Alice reviews client financial statements to flag anomalous entries needing further attention. Despite being repetitive, it's important given the risk of missing fraud. Her view was that with Python programming and natural language processing (NLP), she could automatically extract data points to examine for variations. And, visualising this intelligently could add insight as well as efficiency.

iii. **Prototyping:** Alice realised she could leverage textual algorithms such as spaCY[8] or Natural Language Toolkit (NLTK)[9] to process the textual entries in financial statements. She could build a distribution of concept occurrences within documents (n-grams and frequency counts). And group similar concepts with a word similarity sequence. By building a repository of these, she could compare incoming distributions to the existing steady state. For visualisations, she decided upon a simple scatter chart using Plotly[10]. By plotting average frequency occurrences of the stable state versus frequency occurrences of new documents, she could identify outlier concepts. She felt confident to take her idea to the next level.

FIGURE 4: Coding and the innovation journey

Design thinking	Incubation	Ramp up / Go-live
Enablement ▶ Problem and Solution ▶ Prototyping	Business sponsor ▶ Scoping and Financing ▶ Development	Soft launch ▶ Release and Monitoring ▶ Scaling

---

7   Mihaly, Heder (September 2017). "From NASA to EU: the evolution of the TRL scale in Public Sector Innovation" (PDF). *The Innovation Journal.*

8   <https://spacy.io/>

9   <https://www.nltk.org/>

10 <https://plotly.com/>

B. Incubation

The focus here is to garner support and resources to action product development. Innovation and funding channels depend on the internal initiatives of the firm, so early stage buy-in is vital.

iv. **Business Sponsor:** This is likely to be an individual in the firm with high credibility and standing, such as a revenue-generating partner. As they are closely linked with the firm's business goals and objectives, they would be able to provide strong insight on feasibility and alignment. Alice was initially turned down by her immediate reporting partner who struggled to understand why an algorithm was needed to perform something that can be simply done by a junior associate. However, with the guidance of colleagues, she approached another partner with a focus on internal automation. This individual liked the idea and agreed to be the business sponsor on condition that Alice collaborate with one of his staff members, Johnny, who had also been examining applications for machine learning algorithms.

v. **Scoping & Financing:** As things get serious, to secure internal funding with the blessing of the business sponsor, Alice and Johnny needed to create a full proposal that detailed the end solution, the technology and technical expertise required to build and scale it, as well as projections on cost and expected value. This process took a month as Alice and Johnny refine their concept, scope out the development milestones, and source for additional expertise internally or externally (ie. solution architects, UI designers, business subject matter experts). Johnny had additional ideas on how the usage of Regular Expression (Regex) models or simple entity recognition models would improve accuracy of the grouped concepts, however this was shelved for later releases.

vi. **Development:** Finally, with sign-off from the internal innovation steering committee, and additional support from the firm's technology consulting team, development of the automation tooling was approved to run for two-months of chargeable time. A team of five including Alice and Johnny would dedicate 50% of their chargeable time to building out the tool. Additionally, there were additional processes with the internal Legal and Risk Management teams to ensure that the data that was being processed did not contain any information that could be directly linked back to clients. Provisions in engagement agreements specific to handling and usage of data would also need to be considered.

C. Ramp-up / Go-Live

At this stage, the product is ready for targeted roll-out on live engagements. Start small and validate how useful the users find the product.

vii. **Soft Launch:** This consisted of 8 pilot engagements that were conducted by Alice's and Johnny's department colleagues. Enablement sessions and focus groups were conducted to showcase the functionality and troubleshoot issues.

viii. **Release & Monitoring:** Following the positive feedback of 5 pilot engagements, approval was granted to begin using the tool on a larger number of engagements. The adoption goal was to have the tooling across 120 engagements over 6 months. To enable this, feedback mechanisms were set up to monitor adoption. Additionally, Alice and Johnny also began analysing engagements that reported negative feedback, primarily due to concepts that were not well-categorised by the algorithm or had been wrongly flagged. These would be evidence that can justify the need for more sophisticated Regex or Machine Learning models that could perform at a domain-specific level.

ix. **Scaling:** While the first release was ongoing, talks were underway to augment the functionality of the tooling for another use-case within the firm. Excitedly the team begin planning the roadmap for this next step!

PROFESSIONAL ACCOUNTING AND CONSULTANCY FIRMS, SMALL, MEDIUM, AND LARGE, NEED TARGETED INNOVATIONS TO PRODUCE THE HIGHEST RETURN WITH AS LITTLE DISRUPTION TO THE BUSINESS AS POSSIBLE.

6. **Conclusion**

Coding can be a valuable skillset and there are various levels to which one can get involved. From just a high-level awareness of what it is, to involvement as a user who can partner with specialists to more hands-on development of code itself.

ACCA offers CPD courses across this spectrum. For those interested in a high-level introduction, *Machine learning – an introduction for finance professionals* will provide a business oriented view of AI with some visibility of what Python coding screens and data visualisations look like. For those seeking more hands-on training with Python labs, *Machine learning with Python for finance professionals* will be more appropriate.

NOT EVERY PROFESSIONAL ACCOUNTANT MAY NEED TO CODE, BUT EVEN A BASIC UNDERSTANDING CAN ADD VALUE TO THEIR ORGANISATIONS, HELP TO DIFFERENTIATE THEMSELVES AND OPEN FUTURE CAREER OPPORTUNITIES.