Why Should CPAs Code?

Growing up in the 1980s, Gerry Wilton liked computers and experimenting with coding. When he started working as a CPA in the 2000s auditing credit unions and insurance companies, that interest helped him see how he could use his understanding of programming languages (in this case, Visual Basic for Applications [**VBA**]) to automate the conversion of text files into Excel and reports that could be analyzed and used as audit evidence – a task that would take hours to complete manually, yet one that he was able to do with the press of a button.

"Most accountants are comfortable in Excel but they don't take it to the next level to be able to data mine and take advantage of the tools that are there," says Wilton, who is now director of operations at Mississauga, ON-based BAIWay, a boutique firm that provides training, implementation and consultation on document management systems to the public accounting industry.

In this data age, being able to code and becoming comfortable with programming languages, such as VBA, **Python** and **SQL**, will only grow in importance as technological advancements accelerate and drive the production of data. In *The Digitization of the World From Edge to Core*, IDC predicts the sum of the world's data will grow from 33 zettabytes (ZB) in 2018 to 175ZB by 2025, for a compounded annual growth rate of 61%. In this rapidly changing landscape, where data is the new raw material of business, *Data driven: What students need to succeed in a rapidly changing business world*, a report by PwC, puts coding, predictive analysis and machine learning high on the list of skills accountants will need to succeed.

"Today's accountants, particularly those in the corporate finance function, work with structured data and databases. They are using tools such as SQL to handle this data," says Alan Huang, associate professor of finance at the University of Waterloo's School of Accounting and Finance and cross appointed to the School of Computer Science. "In the future, accountants will be called on to link their understanding of business problems to data. To do that, you need coding skills."

> **"Learning to code and understanding the basics of programming are critical tools for accountants in the age of data."** – Alan Huang, associate professor of finance, *University of Waterloo*

Coding Defined

At its core, coding is the act of writing a script in a programming language a computer can understand. This script will tell the computer to carry out a specific task. Everyday actions such as sending an email or tapping a credit card to make a payment are only possible because someone has written a precise set of instructions (i.e., the code) in a language a computer can understand.

For Wilton, learning to code is about learning to speak the language of the application you are using. It is no longer just the domain of software engineers; it is relevant for anyone working with technology, because it provides control over the technology being used. For accountants already experienced at configuring formulas in Excel, coding is a natural next step.

> "Coding is a tool that allows us to communicate with the apps we use. If you can't communicate effectively, you won't be as efficient as you could be." – Gerry Wilton, director of operations, *BAIWay*

Benefits of Learning to Code

In addition to improving efficiency through the automation of redundant tasks and building the capacity for large-scale data analysis not otherwise possible with Excel, a growing body of knowledge makes the case that coding helps accountants become better logical, systematic thinkers. It requires logically approaching each problem or task as an engineering process (i.e., from identifying the requirements and constraints to developing possible solutions and testing). As well, being able to code and understanding the language of the data age provide a level of confidence and assurance that the results of tasks automated and conducted by a machine are reasonable. If there is a concern, having the skillset to see the logic underpinning the code also enables the appropriate modification of that code when necessary.

Coding for Automation, Data Analytics and Visualization

A firm receives a list of 15,000 T2 file names, but they are arranged haphazardly, with no easy way to sort them. Attempting to rename the files manually would take hours. Writing a script imposing a standard style format and sorting criteria into the naming convention of each file bypasses the redundant action of renaming the files to the machine. The operation takes 15 minutes.

In another example, a new version of CaseWare has just been released, and 30 workstations need to be updated. Rather than configuring each station individually, a script is written that pushes out consistent configuration settings throughout the firm at the same time.

These examples, provided by BAIWay, represent straightforward ways code can be employed to automate labour-intensive processes. Increasingly, CPAs are being tasked with combining, filtering and preparing data from different sources. Understanding the syntax of programming languages and how to write scripts eliminates constraints on what can be achieved. This is important for preparing data effectively and efficiently in order to arrive at crucial business insights.

Malik Datardina, governance, risk and compliance strategist for Auvenir is a certified information systems auditor (CISA). He routinely writes scripts within audit analytics software to automate data preparation for journal entries. "For example, we often receive data from one system that needs to be transferred to another system with different data configurations. I wrote a script to automate the processing of SAP and other data so it could be easily imported into QuickBooks. Done manually, entering the data would take the better part of a day. I was able to write a script in ACL (now Galvanize) and now it can be done within two hours."

Datardina also employs his coding knowledge and analytical tools, such as ACL, to manage and prepare data in order to easily identify specific exception items, generate statistical samples for analysis and reconcile different types of files. "Analytical tools have the natural ability to query and then easily recreate that analysis on a future basis," he says. Another advantage is that the source data does not change; that is a risk in Excel or other spreadsheet tools not designed for analysis.

Not surprisingly, a scan of skills requirements for current finance job postings commonly include:

- the ability to promote automation using basic programming and advanced data analysis tools

- strong knowledge of programming languages, such as VBA, Python, R and SQL, and data modelling tools

- the ability to collect, manipulate and interpret data to isolate key trends and summarize findings into concise and informative reports

- the ability to develop tools to support deep-dive analysis

"CPAs do not have to be data scientists, but we should get comfortable with data. This means we have to learn certain tools and programming languages that allow us to prepare data, such as ACL, Power BI, R and Python. For CPAs in business, this is an invaluable skill set. CPAs able to handle data on their own are more productive, achieve more strategic insights and add more value to their organizations because they are no longer relying on IT or any other business analyst to get the data." – Malik Datardina, governance, risk and compliance strategist, *Auvenir*

"For the Finance function to effectively support our business, most finance professionals at Canadian Tire use SQL in order to draw information and insights from non-financial databases. Decision-making is facilitated by data and we need to understand how to analyze millions of pieces of data from a number of sources, including from our customers, products and transactions, in order to make relevant and informed recommendations." – Eric Kam, manager, financial planning and analysis, digital retail, *Canadian Tire*

Increasingly, accountants in the corporate finance function are called on to conduct comparisons across industries and to find peer com-panies to assess performance. This requires extracting insights from massive amounts of structured data warehoused in databases such as Standard & Poor's Capital IQ and Compustat, which contain:

- detailed financial information and statements

- sector-specific metrics

- hundreds of company-level fundamentals

- supplemental data items for thousands of active and inactive companies

For example, a CPA at a multinational firm might use data from Capital IQ and/or Compustat to prove to national tax agencies its transfer prices are reasonable based on data comparisons it has carried out across industries and peer firms. Understanding coding and programming languages, such as SQL, R and Python, will allow the user to link relational data sets and different elements within the same database to set appropriate parameters to find the right peer group for comparison – a task not otherwise possible if attempted manually.

Once the data is understood and extracted, visualization tools, such as Power BI and Tableau, can be used to map data and tell the story of the numbers in an impactful way. Using these tools requires a fundamental understanding of how they work in order to manipulate data to generate the right reports, refine results, build dashboards and create information systems that provide the insights necessary for decision-making. CPAs versed in programming languages, such as Python and R, can design their own tools that automate the data extraction to visualization processes and liberate time for preparing the business narrative around the numbers.

> **"The most important thing with coding is to get started and practise continuously. Find small tasks you can automate and start there. Once you've built your confidence and skillset, then scale up to more complex problems that could include machine learning and pre-dictive analytics."** – Michael Wong, principal, technology, Research, Guidance and Support, *CPA Canada*

The convergence of the Internet of Things, social media, and cloud and mobile computing are driving the production and collection of massive amounts of data. Businesses and organizations that can effectively analyze, interpret and use this data to anticipate trends and make crit-ical decisions will have a competitive advantage. As the professionals charged with extracting crucial insights from data, it is no longer enough for CPAs to just know how to use software. In this environment, learning how to code and understanding the language of machines will become a foundational aspect of a CPA's skillset.

Getting Started

Coding does not have to be overwhelming. Most CPAs already do basic programming with Excel formulas and VBA. For those looking to become more advanced with automation and data analytics, Python is a popular programming language frequently used in finance and data science. SQL is another supporting language often used to communicate with **relational databases**.

There are many resources available to help you start your coding journey. A popular way to learn is through coding bootcamps. While bootcamps generally require a larger time commitment, the presence of a live instructor and structured course content make them great options for individuals with little to no prior background in coding. Some examples for your consideration include:

- University of Toronto's Data Analytics Boot Camp

- General Assembly's Python Programming Course

- BrainStation's Python Programming Certificate

For individuals who are comfortable with self-learning and who possess some prior technical knowledge, there are many sites that offer free coding courses and tutorials, including:

- codecademy – online catalogue of courses to learn different programming languages

- Python for Beginners – Microsoft Developers Video Series

- SQLBolt – series of interactive lessons and exercises to learn SQL

- R Programming – Coursera course on how to program and use R for data analysis

Most importantly, learning to code is like learning a new language. You need to apply what you have learned and find every opportunity to practise. Do you have a report that needs to be refreshed every morning and sent to another team member? Is there a file that needs to be rolled forward manually every month? Are there public data points (e.g., interest rates and company share prices) that you regularly collect for making the assumptions that form the basis of financial forecasts? These are all potential opportunities where coding can be applied to improve efficiencies and analytics.

Back to the future

According to *How the U.S. Accounting Profession Got Where It Is Today: Part I*, the first business application of a computer was in the mid-1950s when Arthur Andersen & Co. designed and installed a payroll system for General Electric Company's Appliance Park facility in Kentucky. "Accountants were the natural programmers because that's where the information system was," says Malik Datardina. "In the long term, when you look at innovation and how we are going to move forward as a profession, we need to revive a tradition that's been buried. An accounting ledger is at the core of Blockchain yet it was invented by some-one outside the profession. Why? Strategically, we need people who can bring the worlds of accounting, auditing and financial management and technology together and innovate. We need people with that mindset who can make the future happen instead of waiting for it to happen – as we are seeing in audit technology companies."

Brief Breakdown of Coding Terms

API: API, or Application Programming Interface, is a set of functions and procedures that allows for the creation of applications that access the features or data of an operating system, application or web service.

Code editor: A code editor is a text editing program specifically designed for editing code. It makes writing and reading code easier by differentiating the various elements of code and providing auto-complete capabilities. Popular free code editors include Visual Studio Code, Atom and Sublime Text.

Compiler: A compiler is a program that converts source code written in one programming language into machine-code that can be read and executed by a computer.

Git: Git is an open-source version control system used for tracking changes in source code during software development. GitHub is a popular service based on Git that is used by programmers to co-ordinate work and collaborate on software development.

Python: Python is a widely used general purpose, high-level programming language designed to emphasize code readability. It is frequently used in the fields of data science and finance.

R: R is a programming language and environment for statistical computing and visualization. It is primarily used for statistics problems and data analysis.

Relational database: A relational database is a collection of data sets organized by tables, records and columns. These data items often have predefined relationships among themselves that form the structure of the relational database. For example, relational databases will have rows in a table called "a record" with a unique ID called "the key." The columns of the table hold attributes of the data, and each record usually has a value for each attribute, making it easy to establish the relationships among data points.

SQL: SQL, or Structured Query Language, is a programming language specifically used to communicate with relational databases.

VBA: VBA, or Visual Basic for Applications, is a programming language designed to be used in the Office suite of applications.