Unit 3
Writing an introduction, and designing effective visual aids

Overview

An effective introduction is crucial in arousing your audience's interest, identifying a knowledge gap, and facilitating the understanding of the remaining writing or oral presentation. This unit will highlight its importance in writing with respect to its purpose, structures and language features which you can apply in your project. Specific to oral presentation, techniques and pitfalls of designing visual aids will also be discussed to help you to clearly and concisely convey the appropriate technical information.

Learning outcomes

By the end of this unit, you will be able to:

- identify the purpose and structure of writing an introduction in a report
- distinguish the language choices in an introduction
- draft an introduction for a progress report for your project
- design effective visual aids for an oral presentation
- design and present figures and tables effectively in an oral presentation

(Warm-up) Critique an introduction

Read the following introduction of an interim report written by a previous student. Discuss your comments with your partner.

Hint: Think about the purpose of an interim report as discussed in Unit 1.

© Centre for Applied English Studies, HKU

Text 1 [1]

TITLE: where2cut (a salon & hairstyle guide website)

INTRODUCTION

Hong Kong is taking one of the leading roles in digital and information technology. The consumption habit of peoples in Hong Kong becomes more digital with the widely use of Internet and digital devices. With the access to Internet, people now become rational consumers. Most of the Hongkongers often do research on the Internet before they consume. Internet makes different kinds of information accessible to everyone. For dinning, people will go to *openrice.com*; for shopping, *price.com.hk* may be their choice; for cosmetic & beauty products, people may surf *beautyexchange.com*. However, what about if you want to have a haircut? Which website will you go? You can hardly think of that, right? In fact, there is a lack of a strong, powerful and reliable network for Salons in Hong Kong. Although there are some small forums about salons, people still cannot figure out where to go since they do not know which one is trustworthy.

People Hong Kong will go to Salon every 1-2 month in average to have a haircut, hair treatment, styling, or other hair services. And with roughly estimation, Hong Kong has over 1000 salons spread over Hong Kong Island, Kowloon and New Territories. Under such high level of supply and demand for the Hair-Cutting Market, there is an essential need to provide a convenient and dependable platform for consumers to discuss and exchange information in order to locate their desirable salons and at the same time, for suppliers to promote their shops.

"Do you know which Salon is good?" "Where will you go for your hair cut?" "Which stylist is good?" "What are the costs for hair cutting, stylizing, and treatments?" "Which stylists are good?" "How can I tell the stylists what kind of hair-style I want to have?" These might be the questions that you and I have come across.

With the use of ***where2cut***, people no longer need to suffer from the above questions. What they need to do is just check on where2cut.com and search the salon that suitable for them. People can experience the salons, rate their favorite salons, comment on them or the stylists, as well as discuss and share their feelings with others, no matter it is good or bad.

© Centre for Applied English Studies, HKU

3.1 Purpose and structure of an introduction

An introduction

- is an overview of the background, with appropriate references, through which you show:
    - what has been done by others in the area
    - what opinions are given in the discussion
    - how these opinions interact
    - current status
    - future directions
- defines problem to be solved (or a research gap to be filled)
- justifies and explains the importance/benefits of your work as a solution to the problem (or to bridge the research gap)
- states the scope of your work (this can be a separate section if it is substantial)
- outlines deliverables (e.g., a computer program, technique, website, etc.)
- provides an outline for the remaining parts of the report

TASK 3.1 Identify the purpose and structure of an introduction

Work as a pair. Read an abridged version of an introduction of a technical report/paper below titled "Example-Centric Programming: Integrating web search into the development environment" [2]. Identify the features listed in the section.

Additional thoughts: Are you convinced of the benefits/motivations of the project? Is there sufficient general and technical background information?

© Centre for Applied English Studies, HKU

Text 2 [2]

TITLE: Example-Centre Programming: Integrating Web Search into the Development Environment

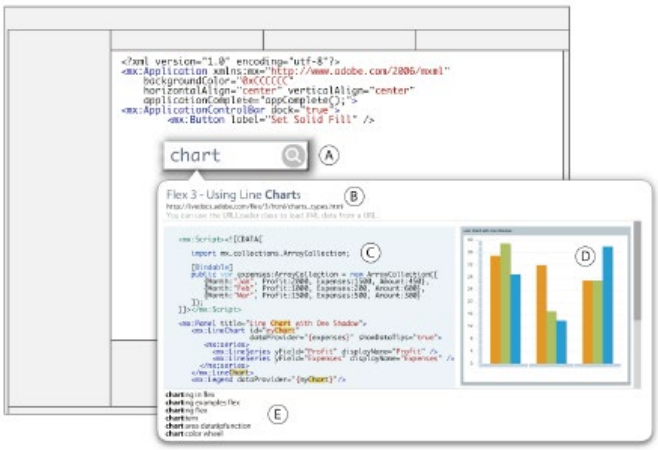


Figure 1. The *Blueprint* plug-in for the Adobe Flex Builder development environment helps programmers engage in example-centric development. A hotkey places a search box (A) at the programmer's cursor position. Search results are example-centric; each result contains a brief textual description (B), the example code (C), and often a running example (D). *Blueprint* also provides additional search suggestions (E).

INTRODUCTION

Programmers routinely face the "build or borrow" question [1]: should they implement a piece of functionality from scratch, or locate and adapt relevant existing code? Web search is fundamentally changing the cost structure of this question [2]. It is now possible to quickly locate example code that implements nearly any piece of routine functionality [3]. This enables programmers to opportunistically build applications by searching for, modifying, and combining short blocks of example code taken from the Web [4-6].

[text removed]

Finding appropriate code has become easier: there are many Web sites dedicated to example sharing (*e.g.,* the Flex Examples Blog [8]), online opensource code repositories (*e.g.,* Google Code [9]), and search interfaces for programmers [10, 11]. However, these *search tools* are still wholly separate from *editing tools*. Current development environments provide little support for example-centric development. Instead, they tacitly assume that programming begins *tabula rasa* and that code is either written by the programmer or imported as a library module.

Several difficulties arise from separate tools for editing and search. First, the important link between borrowed code and its source is lost. The programmer may not realize the code was borrowed from an

Put the corresponding function to the right of the text.

© Centre for Applied English Studies, HKU

online source; this can be valuable when debugging or modifying code. If they do know it was borrowed, but not the URL, they may have difficulty re-finding it if they would like to verify attributes or view additional code and commentary. Second, if the source example is later updated (*e.g.,* to fix a bug), the programmer will never know. Finally, to obtain relevant search results, programmers must manually specify contextual constraints in their query, such as languages and frameworks used. We hypothesize that there is significant value in integrating Web search with a code editor. More specifically, this paper proposes that *automatically augmenting queries with code context* and presenting an *example-centric view of search results* increases the speed, quality, and ease of programming by example modification. We introduce *Blueprint*, an extension to the Adobe Flex Builder development environment that manifests these ideas (see Figure 1).

This paper makes two contributions. First, it introduces a user interface that integrates searching for example code into a development environment. This search interface presents blocks of example code, augmented with *running examples* and *written descriptions* when available (see Figure 1). In a between-subjects comparison with 20 participants, we found that *Blueprint* enables participants to search for and select example code significantly faster than with a standard Web browser. Second, this paper introduces a technique for retrieving relevant example code from the Web for a user's query. To maximize speed, breadth, and ranking quality, the *Blueprint* server leverages a general-purpose search engine.

[text removed]

The remainder of this paper proceeds as follows. First, to motivate *Blueprint*'s interface choices, we offer background information on how programmers use the Web to inform our design. We then present a scenario enabled by *Blueprint* and describe its implementation and evaluation. Next, we offer a discussion of the design space of tools to support programmer Web use, and position *Blueprint* within this space to better understand its strengths and limitations. We close with a survey of related work and thoughts on future research directions.

© Centre for Applied English Studies, HKU

3.2 Language choice in an introduction – tense and voice

3.2.1 Use of tense

Different tenses are used in describing:

- general phenomena
- existing situation / current status (what has OR has not been done in the field)
- future plan / work to be done
- aim or organization of the report

Without strictly memorizing the use of specific tenses, try to recognize the tense used for each function.

TASK 3.2 Identify the use of tense

Look at the highlighted tenses and verb forms in the extract below.

Present tense **modal verb** **present continuous** **past tense** **present perfect**

1. Go through each of the tenses highlighted in yellow. Why is this tense used rather than other tenses?
2. What other tense / verb form can be used in the final paragraph?
3. Go through each of the tenses highlighted in green. Why is this tense used rather than other tenses?
4. What are the differences in meaning between the pairs of sentences below:

    a. Web search **is** fundamentally **changing** the cost structure of this question

    b. Web search **has** fundamentally **changed** the cost structure of this question

    c. *Finding* appropriate code **has become** easier:

    d. *Finding* appropriate code **is becoming** easier:

5. Why use the modal verb in the first paragraph?

Text 3 [2]

TITLE: Example-Centre Programming: Integrating Web Search into the Development Environment INTRODUCTION Programmers routinely **face** the "build or borrow" question [1]: **should** they **implement** a piece of functionality from scratch, or **locate** and adapt relevant existing code? Web search **is** fundamentally **changing** the	**Comments**

cost structure of this question [2]. It **is** now possible to quickly locate example code that **implements** nearly any piece of routine functionality [3]. This **enables** programmers to opportunistically build applications by searching for, modifying, and combining short blocks of example code taken from the Web [4-6].

In 1993, Nardi **suggested** that "programming by example modification" **holds** significant latent value. An open question at the time **was** "how users will find appropriate example code [text removed]

[text removed] *Finding* appropriate code **has become** easier: there **are** many Web sites dedicated to example sharing (*e.g.,* the Flex Examples Blog [8]), online opensource code repositories (*e.g.,* Google Code [9]), and search interfaces for programmers [10, 11]. However, these *search tools* **are** still wholly separate from *editing tools*. Current development environments **provide** little support for example-centric development. Instead, they tacitly **assume** that programming **begins** *tabula rasa* and that code **is** either **written** by the programmer or imported as a library module.

We **hypothesize** that there is significant value in integrating Web search with a code editor. More specifically, this paper **proposes** that *automatically augmenting queries with code context* and presenting an *example-centric view of search results* increases the speed, quality, and ease of programming by example modification. We **introduce** *Blueprint*, an extension to the Adobe Flex Builder development environment that **manifests** these ideas (see Figure 1).

This paper **makes** two contributions. First, it **introduces** a user interface that integrates searching for example code into a development environment. This search interface **presents** blocks of example code, augmented with *running examples* and *written descriptions* when available (see Figure 1). In a between-subjects comparison with 20 participants, we **found** that *Blueprint* **enables** participants to search for and select example code significantly faster than with a standard Web browser. Second, this paper **introduces** a technique for retrieving relevant example code from the Web for a user's query. To maximize speed, breadth, and ranking quality, the *Blueprint* server **leverages** a general-purpose search engine.

[text removed]

The remainder of this paper **proceeds** as follows. First, to motivate *Blueprint*'s interface choices, we **offer** background information on how programmers use the Web to inform our design. We then **present** a scenario enabled by *Blueprint* and **describe** its implementation and evaluation. Next, we **offer** a discussion of the design space of tools to support programmer Web use, and position *Blueprint* within this space to better understand its strengths and limitations. We **close** with a survey of related work and thoughts on future research directions.

© Centre for Applied English Studies, HKU

3.2.2 Use of First Persons

One common question in technical writing is whether first persons such as 'I' or "we", should be used. While there may not be a universal and hard-fixed rule, and the alternative of passive voice is always available, the rationale depends on whether there is a need to highlight the author when compared to the associated event.

TASK 3.3 Identify and explain using and not using first person

Look at the highlighted phrases in the text and answer the questions below:

1. The subjects of clauses have been highlighted as follows: people, things, other. Which is the most common kind of subject? Why has the writer done this?

2. There is a phrase beginning with it in Paragraph 1. Can you think of any other phrases beginning with it?

3. What does this refer to in Paragraph 1? Can this be replaced with it?

4. In Paragraph 2 why is Nardi the subject of the sentence?

5. Why do all the other citations not mention the authors of the sources?

6. In Paragraph 3 there is a reduced relative clause:
   there are many websites (that/which are) dedicated to example sharing
   a. Why is it reduced?
   b. Can you find another example of a reduced relative clause in the text?

7. What does they refer to in Paragraph 3?

8. What does it refer to in Paragraph 5? Can it be replaced with this?

9. What does "we" refer to in Paragraphs 4 – 6?

10. If you have an individual FYP can you replace "we" with "I"?

© Centre for Applied English Studies, HKU

Text 4 [2]

TITLE: Example-Centre Programming: Integrating Web Search into the Development Environment

INTRODUCTION

Programmers routinely face the "build or borrow" question [1]: should they implement a piece of functionality from scratch, or locate and adapt relevant existing code? Web search is fundamentally changing the cost structure of this question [2]. It is now possible to quickly locate example code that implements nearly any piece of routine functionality [3]. This enables programmers to opportunistically build applications by searching for, modifying, and combining short blocks of example code taken from the Web [4-6].

In 1993, Nardi suggested that "programming by example modification" holds significant latent value. An open question at the time was "how users will find appropriate example code… [text removed]

[text removed] *finding* appropriate code has become easier: there are many Web sites dedicated to example sharing (*e.g.,* the Flex Examples Blog [8]), online opensource code repositories (*e.g.,* Google Code [9]), and search interfaces for programmers [10, 11]. However, these *search tools* are still wholly separate from *editing tools*. Current development environments provide little support for example-centric development. Instead, they tacitly assume that programming begins *tabula rasa* and that code is either written by the programmer or imported as a library module.

We hypothesize that there is significant value in integrating Web search with a code editor. More specifically, this paper proposes that *automatically augmenting queries with code context* and presenting an *example-centric view of search results* increases the speed, quality, and ease of programming by example modification. We introduce *Blueprint*, an extension to the Adobe Flex Builder development environment that manifests these ideas (see Figure 1).

This paper makes two contributions. First, it introduces a user interface that integrates searching for example code into a development environment. This search interface presents blocks of example code, augmented with *running examples* and *written descriptions* when available (see Figure 1). In a between-subjects comparison with 20 participants, we found that *Blueprint* enables participants to search for and select example code significantly faster than with a standard Web browser. Second, this paper introduces a technique for retrieving relevant example code from the Web for a user's query. To maximize speed, breadth, and ranking quality, the *Blueprint* server leverages a general-purpose search engine.

[text removed]

The remainder of this paper proceeds as follows. First, to motivate *Blueprint*'s interface choices, we offer background information on how

programmers use the Web to inform our design. We then present a scenario enabled by *Blueprint* and describe its implementation and evaluation. Next, we offer a discussion of the design space of tools to support programmer Web use, and position *Blueprint* within this space to better understand its strengths and limitations. We close with a survey of related work and thoughts on future research directions.	

3.3 Putting it all together

Now that you understand the purpose and structure of an introduction, and the rationale for the choice of tense and voice, you can apply them in your report.

TASK 3.4 Draft an introduction of your Progress Report (25 mins)

Discuss your project with a peer or small group. Identify the main elements based on Section 3.1 which are specific to your project. Write a brief introduction of your report.

Hint: It is ok that you may not have all the background information available or are aware of all the possible outcomes. You can just put in temporary placeholders. Also think of the purpose of your progress report which focuses on reporting progress, which is different from the final report shown. Subsections will help.

© Centre for Applied English Studies, HKU

3.4 Designing PowerPoint Slides

Your class teacher will cover some principles of effective visuals and ask you to complete some activities.

TASK 3.5 Improve a PowerPoint slide of your project plan

Select one slide (preferably with technical details) from the presentation of your project plan. Improve it with what you have learnt in this lesson. Present it to another group of students not in your project. Also give feedback to the other students using the template "Peer Review of Visual Aids" provided.

Peer Review Checklist of Visual Aids

Creators of the Visual Aids: _____

Assessor: _____

	Good	Need Improvement	N/A
Layout/Style			
❖ Slide no. large enough			
❖ An illustrative topic/title on each slide			
❖ A suitable number of points on a single slide			
❖ Not too wordy or lines of text (no double wrapping)			
❖ Clear color contrast of background against font (Use a light background)			
❖ No irrelevant/informal pictures			
❖ Graphs/charts/diagrams with captions and highlighted/annotated feature/keywords			
❖ Font size large enough			
❖ Roadmap			
Language			
❖ Parallel sentence patterns with bullet point			
❖ Accurate and consistent grammar and spelling			
❖ Formality			
❖ Accurate description of figures			

© Centre for Applied English Studies, HKU

3.5 Presenting Figures and Tables

Why is it important?

Figures and tables are fundamental to the readers' understanding of complex information. When presented well, they can organise, summarise, visualise, and consolidate a significant amount of data/results, refocus the readers after reading paragraphs of text, and provide new angles for further development or discussions. However, failure to do so may give readers negative impressions, such as a lack of preparation and inattention to details.

TASK 3.6 Compare two slides

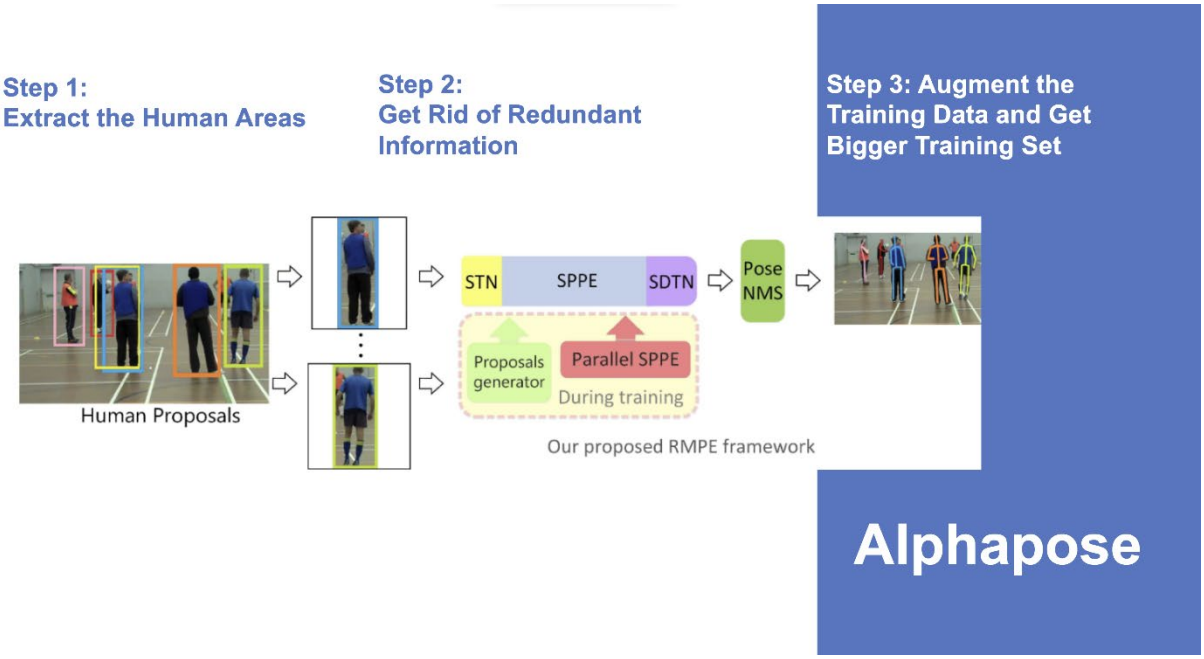
Watch the following YouTube video from (00:00-01:05) and compare Slide #1 and Slide #2. Discuss with a classmate which slide you prefer and why.

Animate Charts Properly in PowerPoint

TASK 3.7 Evaluate sample student slides

Below are two examples produced by past students. Discuss the visual aspects with a classmate. How can you improve them?

Sample Slide 1



Sample Slide 2

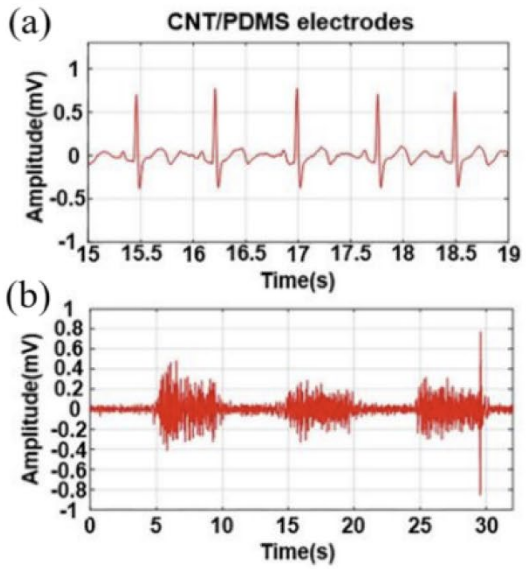
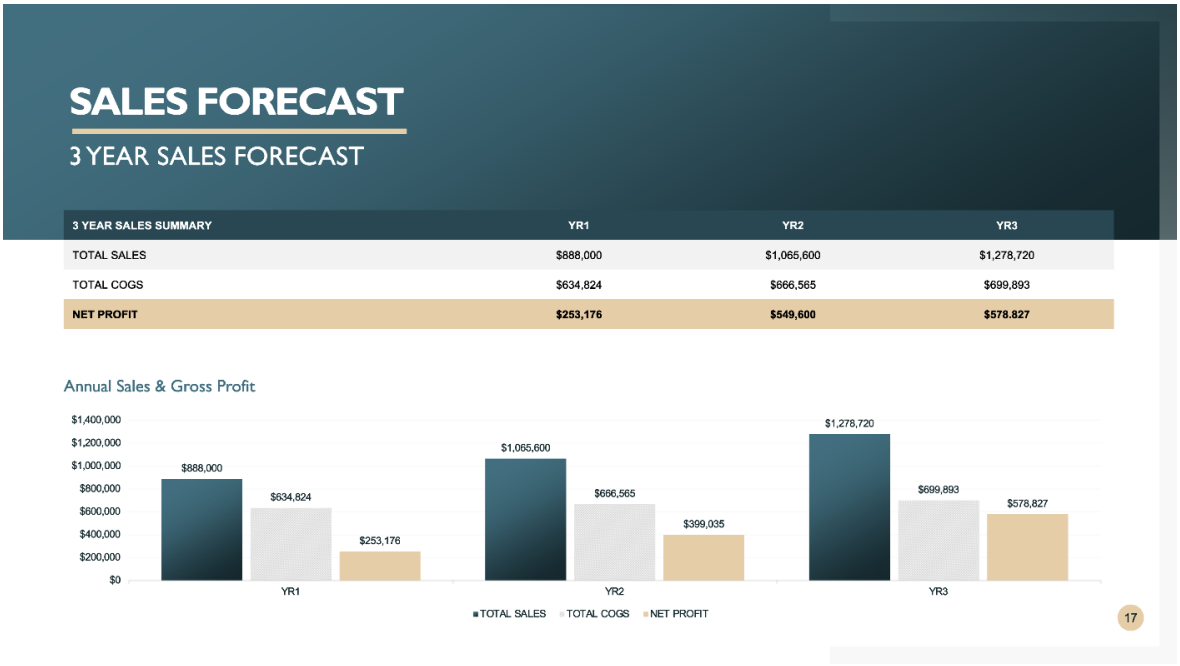


Figure 3.7.2 *Graphs of amplitude (i.e. Voltage) over time of carbon nanotubes/Polydimethylsiloxane (CNT/PDMS) electrodes showing (a): Electrocardiogram (ECM) signals and (b): Electromyography (EGM) signals [104]*

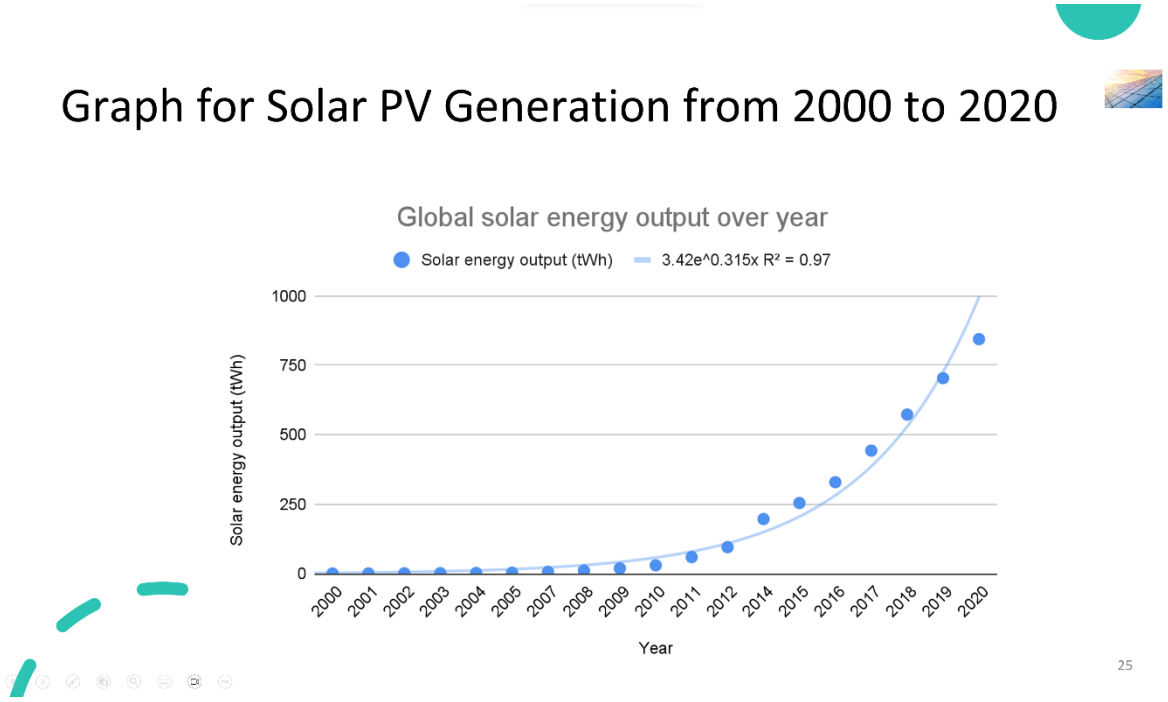
TASK 3.8 Evaluate and present the slides

Below are two examples produced by past students. Discuss the visual aspects with a classmate. Imagine you are the presenter; how will you present the information on the slides?

Sample Slide 3



Sample Slide 4



TASK 3.9 Improve and present your own figures and tables

Do you have a figure/table in one of your past reports/assignments? Show it to your classmate and discuss how the visual aspects can be improved. Then take turns presenting the figure/table to your classmate and offer him/her some useful feedback.

3.6 In summary, how are figures and tables presented?

There are diverse views on how figures and tables should be used in presentations (unlike in written reports or dissertations where there are more consistent 'rules' for the formatting of figures and tables). Some professional settings such as international conferences may have strict guidelines for formatting, while others are more flexible.

Here are several suggestions when presenting figures and tables in a presentation.

1. include either a title or a caption to briefly describe the figure or table,
2. ensure that all axes, legends, annotations, and lines are clearly labelled in charts/diagrams,
3. recreate tables on the slides instead of using screenshots,
4. ensure all visual elements within a figure/table are sharp and clear (e.g., no blurry lines, no block coding artifacts, use vector graphics, and readable on computer screen from at least 1m away),
5. add annotations to emphasize important data, if necessary (e.g., circle important numbers)
6. cite the source of a figure/table if the figure/table is not produced by you (e.g., images taken from the internet),
7. avoid packing a slide with too many figures and tables (i.e., usually maximum one main figure or table per slide).

Here are some optional approaches to further improve the look-and-feel of figures and tables in presentation.

1. Use coloured figures and tables instead of black-and-white ones,
2. Use consistent formatting style and sequential figure/table numbers throughout the slides,
3. Use animated figures/tables (e.g., animated GIF) or videos to show movements or trends in data/results.

3.7 Three steps for presenting figures and tables

When presenting a figure or table, use the following three steps:

Step 1: Say what the visual is about (e.g., title)
Step 2: Introduce the basic elements of the visual (e.g., axes, scale, legend, colours of lines),
Step 3: Talk about the main trends in the visual:

- Describe trends (e.g., highs, lows, peaks, troughs, clusters, slopes),
- Explain trends (e.g., why and how these trends appear),
- Describe the importance of these trends (e.g., how are these trends related to your research aims/questions/design?)

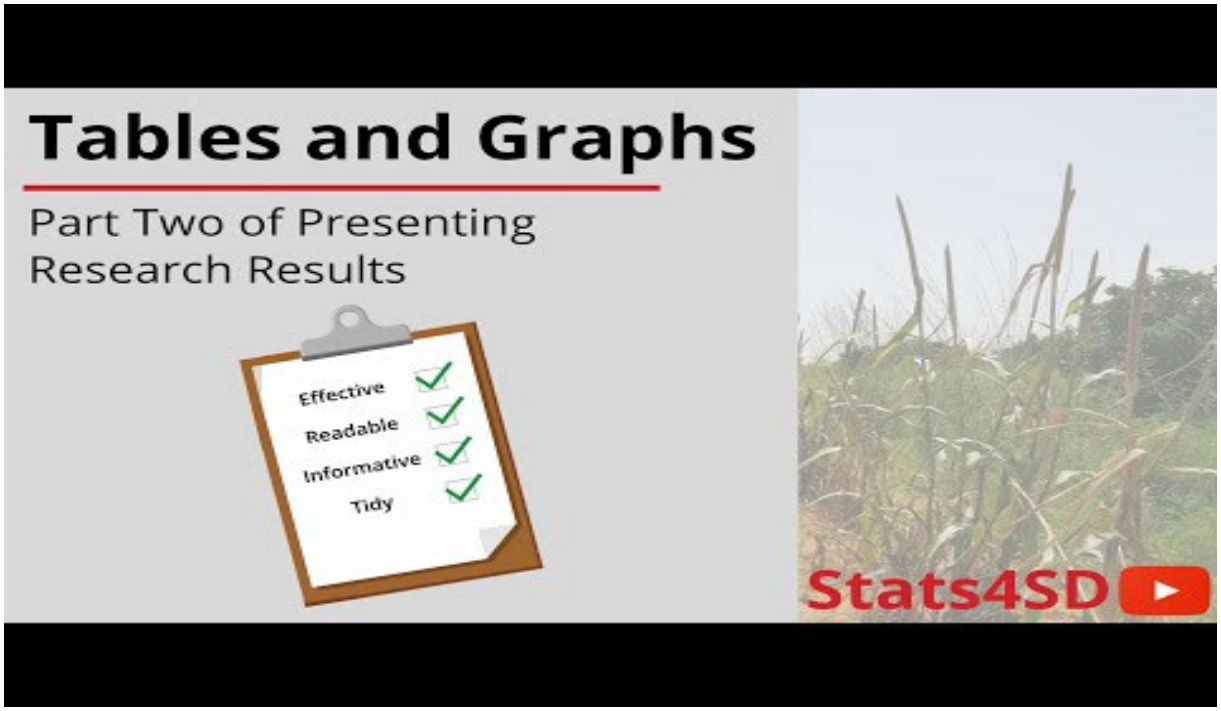
© Centre for Applied English Studies, HKU

3.8 Additional resources for presentation skills

Here are two more useful videos you should watch if you want to improve your presentation skills:

1. 6 Things You're Doing WRONG⚡in PowerPoint

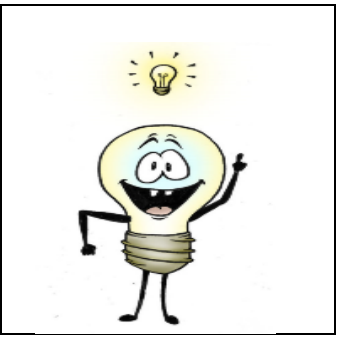
2. Presenting Research Results: Tables and Graphs



3.9 Over to you

Introduction does not only provide a so-called 'preview' of the rest of a piece of writing or presentation. Consider it the first opportunity to justify 'why' you are pursuing your research or study. Since it is generally more structured than other sections in a written or spoken text, try to have a checklist to remind yourself of the essential elements as listed in Section 3.1. Regarding effectively designing PowerPoint slides, see if you can use your slides to track a simple storyline. If you can, your slides are focused, clear, and concise enough.

TASK 3.10 Reflect on this unit



Key points to remember

- Purpose of an introduction: Arouse audience attention, identify motivation, and outline the remaining text
- Essential elements include: background, research gap/problem(s) to be solved, motivation of the study, scope, deliverables, and outline.
- Use past tense to highlight specific literature and past studies (report of findings)
- DO NOT indiscriminately use present tense and passive voice.
- Use first persons to highlight personal reference to specific literature.
- Characteristics of a PowerPoint slide design: focus, clarity, conciseness, aesthetics.
- Use a specific type of graph for a specific type of purpose
- Always 'speak' to refer and highlight content on a PowerPoint slide. Content on a slide does NOT speak for itself!

Homework and Preparation for the next session

- Prepare an introduction and background for your Progress Report

© Centre for Applied English Studies, HKU

References

[1] Adapted from student texts.

[2] Brandt J, Dontcheva M, Weskamp M, Klemmer SR. Example-Centric Programming: Integrating Web Search into the Development Environment. *Stanford Computer Science Technical Report.* Stanford University; 2009.

[3] Grussendorf M. *English for Presentations*. Oxford: Oxford University Press; 2007.

[4] Adapted from student texts.