| **Session Objectives** | * Explain what a Python list is and why they’re used * Be able to create a list and populate it with elements * Add, remove, and modify items in a list * Apply a number of methods to manipulate or summarise lists |
| --- | --- |
| **Key Points** | * What is a list in Python * How do you declare and use a list * How do you edit and manipulate a list * What common methods apply to lists |
| **Assessment** | * Via practical challenges - see task sheet below |
| **Instructor Prep** | * **Note**: These session plans include sections where long explanations are provided simply because it is important to ensure that these tricky concepts are communicated as clearly as possible. However, it is not expected that the instructor will recite these verbatim, feel free to use your own refined and well-honed approach as long as the learning point is covered. The provided narrative is always available if needed. * **Learning and delivery** can be more effective if resources and tasks are personalised. If time permits, feel free to update slides with your own examples such as replacing cat examples with your own pets, or anything else. |
| **Materials** | [TIFC1-PF-3 - Introducing Lists - Slides](https://docs.google.com/presentation/d/1IusmQt0Ot8RQGL2-Ry4oK82d4DPwlMW5k9a6UPAxHqM/edit?usp=sharing)  [TIFC1-PF-3 - Introducing Lists - Tasks](https://docs.google.com/document/d/13KHnEXFd3h8lmo6Ah6bfyUGGtnP65r6uKt_2FCvwCe4/edit?usp=sharing)  Instructor Only: [TIFC1-PF-3 - Introducing Lists - Solutions](https://docs.google.com/document/d/11RcLWxtQztUBDkTEih--bVD9J9UQtLtiipJrOtHdeXg/edit?usp=sharing) |

| **Time** | **Activity** |
| --- | --- |
| 10 mins | **Slides 2 - 3 - Introducing Lists**  **Show Slide 2 - Introduce** your new ‘variable’   * **Show Slide 3 - Ask** if any learners can explain what the code in the example does? If some positive responses are forthcoming, thank the contributors then **explain**:   “*We’re now moving on to look at the next important feature of coding, which is lists. Note we didn’t say feature of Python, because most coding languages, and many environments that aren’t for coding, support creating and using lists*.” |
| 5 mins | **Slides 4 - 5**   * **Show Slide 4 - Read** key points about lists * **Show Slide 5 - Point** **out** or **call out** learner to identify key elements and syntax in the sample code |
| 5 mins | **Slides 6 - 7 - Accessing Elements in a List**   * **Show Slide 6 - Read** key points about accessing elements within lists * **Show Slide 7 - Point out** or **call out** learner to identify key elements and syntax in the sample code |
| 10 mins | **Slides 8 - 10 - Inserting and modifying Elements in a List**   * **Show Slide 8 - Explain** and **point out** the syntax for modifying elements within a list, i.e. recall the list name and index number of item in question, then assign a new value. * **Show Slide 9 - Explain** the following:   “*We’ve already used some methods to modify strings, well methods are actually just functions, made by someone else, who has written the code to provide the desired behaviour. In the future you will write your own functions which you can call in the same way. So far we’ve used some string methods, but there are many others, and you can import thousands of additional functions that you can call as methods to extend Python’s functionality.*  *Here we have a method called ‘append()’ and what it does is add an item to the end of your list.*”   * **Explain** the following:   “*Being able to explain what a piece of code does is an important skill, which can be very important in your search for a role. Many companies will include a coding challenge during recruitment, not a super-difficult one, but they want to know you can explain what you’re doing and the components, such as variables, strings, lists, loops, etc. that you’re using, and why*”   * **Ask** if any learners can talk through the first then second sample code examples * **Show Slide 10 - Read** statement on slide, and point out syntax in code example |
| 10 mins | **Slides 11 - 17 - Removing Elements from a List**   * **Show Slide 11 - Read** summary of three methods of removing elements from a list * **Show Slide 12 - Read** description of Del statement and **point out** or **call out** learner to identify correct syntax in code sample * **Show Slide 13 - Read** description of the pop() method and **point out** or **call out** learner to identify correct syntax in code samples on Slide 14 & 15 * **Show Slide 16 - Read** description of the remove() method and **point out** or **call out** learner to identify correct syntax in code samples on Slide 16 & 17 |
| 10 mins | **Slides 18 - 22 - Additional List Methods**   * **Show Slide 18 - 19 - Explain** the following:   “*There are many methods that can be used with lists, we’ll look at a few of the most common ones. Here the sort() method is used for sorting a list alphabetically or in ascending value.*  *However, use with caution, because it does re-organise the list.*  *If you need to retain the original index number, but still want to view the values in order then you can use another method.*”   * **Point out** or **call out** a learner to identify correct syntax in code sample on Slide 19 * **Show Slide 20 - Explain** that Sorted lets you view the contents of a list without changing the original order. Point out or call out a learner to identify correct syntax in code sample on Slide 20 * **Show Slide 21 - Explain** that the reverse() method will reverse the order of the original list. Point out or call out a learner to identify correct syntax in code sample on Slide 21 * **Show Slide 22 - Explain** that the len method will return the number of items in an input. Point out or call out a learner to identify correct syntax in code sample on Slide 22.   Explain that  “*The len() method can be used on inputs other than lists. Try to figure out how to apply it to a string during your next practical task.*” |
| 5 mins | **Slide 23 - Indexing Errors**   * **Show Slide 23 - Point out** or **call out** a learner to identify the cause of the problem. If correct responses received thank the contributors, else **explain**:   “*As we mentioned, the index number for the elements of a list starts at 0, therefore the index number of an item is usually 1 less than the item’s number if you were to count them out normally i.e. 1, 2, 3, 4,....*” |
| 5 mins | **Slide 24 - Example code**   * **Show Slide 24 - Call out** for a learner to identify whether the illustrated code would run without error. It should. Demonstrate by copying from Presenter’s Notes into VSC. * **Share** the code so that learners may try it during the following task. |
| 60 mins | **Slide 25 - Hands-On Challenges**   * **Share link** to task sheet, instruct learners that they now have 60 minutes to work through the task sheet. Once the time is up they should take a screenshot/snip of their last completed challenge and submit it on Canvas. * **Open breakout rooms** - instructor to select number/mix |