SEL2229 Assignment 1 Full Checklist

This checklist will allow you to cross off items each week and make sure you covered everything. Any specific information about exactly how you should do this will be included in the [weekly walkthrough](#).

▷ Week 1 (no later than 31/01/2025)

- ☐ Sign up for Perusall
- ☐ Join the class using the join code
- ☐ Explore week 1 reading
- ☐ Sign up for/Explore Posit
- ☐ Use a template to create assignment 1 Posit project, ensuring to make this a permanent personal copy.
- ☐ Complete week 1 reflections in `assignment_reflections.Rmd`. in Posit

▷ Week 2 (07/02/2025)

- ☐ Take the technology survey
- ☐ Do some math
- ☐ Create a simple script
- ☐ Upload a data file
- ☐ Do some stuff with the data file
- ☐ Try out Swirl
- ☐ Complete week 3 reflections in `assignment_reflections.Rmd`. in Posit

▷ Week 3 (14/02/2025)

- ☐ Browse documentation and cheat sheets
- ☐ Download `colourTidy.R` script and `colourData.csv`
- ☐ Upload new files, and organise all of your files
- ☐ Copy and modify the script to load the data using the tidyverse
- ☐ Add comments to each line of the script to explain what it's doing
- ☐ Complete week 3 reflections in `assignment_reflections.Rmd`. in Posit

▷ Week 4 (21/02/2025)

☐ Create a copy of the script for this week. Remember to include comments throughout

☐ Find unusable data points and clean them from the dataset.

☐ Use `mutate()` to create recoded, categorical versions of open-ended responses

☐ Use `group_by()` and `summarize()` to find the proportion of compound and non-compound colour terms for artists and non-artists

☐ Complete week 4 reflections in `assignment_reflections.Rmd`. in Posit

▷ Week 5 (28/02/2025)

☐ Create an account on surveyjs.io and create a new survey

☐ Add two pages, a title/description for each page, and a consent question on the first page.

☐ Add an acceptability matrix question and separate categorical acceptability questions

☐ Create questions that collect information about age and region.

☐ Create inputs that allow open-ended responses for region and duration of residence.

☐ Make questions required and change values/variable names so they are transparent

☐ Use validation to add branching for consent and region.

☐ Preview and copy edit your survey to make sure it works, then copy the JSON from Surveys.js into the `index.html` file in the `Week5_SurveyDemo` folder in your assignment project in Posit.

☐ Complete week 5 reflections in `assignment_reflections.Rmd`. in Posit

▷ Week 6 (07/03/2025)

☐ Create your script and load the tidyverse and the data file. Use `View()` and `summarize()` to understand each of the columns.

☐ Use the templates to create basic plots for each of the linguistic variables, with Region coded by colour.

☐ Use `scale_colour_manual()` to change the figure legend and colours used for region.

☐ Use `group_by()` and `summarise()` to calculate means and standard deviations, and use these to create additional plots for syntactic type and semantic likeness category.

☐ Complete week 6 reflections in `assignment_reflections.Rmd`. in Posit

▷ Week 7 (14/03/2025)

☐ Get the data file for the first experiment and the analysis script from the OSF repository for Sidhu, Pexman & Vigliocco (2019), and upload this to your Posit assignment project.

☐ Create your script and load the tidy verse and the datafile, using summary() and View() as needed to understand the structure of the data file and the relevant columns.

☐ Use filter() to clean the dataset and remove irrelevant rows.

☐ Recreate a version of Fig.1 from the paper using the template

☐ Complete week 7 reflections in `assignment_reflections.Rmd`. in Posit

▷ Week 8 (21/03/2025)

☐ Get the data file from the external R project and upload this to your own Posit project.

☐ Create a script for the week and read the data file into a tibble; use `View()` and `summary()` as needed to understand the data.

☐ Find the problem with the `ProducedLabel` column and use `filter()` to clean unusable responses from the dataset.

☐ Use the `stringdist` package and `mutate()` to add a column that calculates the Levenshtein distance between TargetLabel and ProducedLabel

☐ Use `str_length()`, `max()` and `mutate()` to add a column with normalised Levenshtein distance for each response. Use `summary()` to verify normalised distances fall within a reasonable range.

☐ Calculate means and standard errors for normalised distance by condition and test block, and integrate these into a plot.

☐ Complete week 8 reflections in `assignment_reflections.Rmd` in Posit.

☐ FINAL SUBMISSION: 25/03/2025, 3pm