**How to generate a creel survey schedule using the “CreelScheduleGenerator” Script**

*Background*

The creel schedule provides the sampling framework for collecting creel survey data. Overall, the schedule denotes the dates and times that should be sampled (i.e., creeled). While there are many approaches to generating a creel schedule, the “CreelScheduleGenerator” (hereafter “the Scheduler”) uses project-specific, user-supplied inputs to create sampling frames and then selects samples (i.e., dates and times) using a multi-stage, stratified, probabilistic sampling approach. The Scheduler follows the general methods used by Malvestuto et al. (1978), Pollock et al. (1994), and Hahn et al. (2000) to create a schedule for the purpose of collecting temporarily representative data that characterizes the true underlying patterns and fluctuations angler effort and catch rates to ultimately generate unbiased estimates of catch.

*Instructions*

The following steps provide an overview of how to generate a creel survey schedule using the “Creel\_Schedule\_Generator”. If additional guidance is needed, contact Kale Bentley ([kale.bentley@dfw.wa.gov](mailto:kale.bentley@dfw.wa.gov)) or your Study Design Lead.

1. **Download the “CreelScheduleGenerator” repository**
   * The most up-to-date version of the Scheduler is located on GitHub [here](https://github.com/wdfw-fp) and is stored as a [repository](https://docs.github.com/en/repositories/creating-and-managing-repositories/about-repositories)
   * To use the Scheduler, the repository must be downloaded (i.e., cloned) to your local computer (for further details see “[Clone the Repository with RStudio](https://resources.github.com/github-and-rstudio/)” sub-section). The repository can be located anywhere on your computer since you’ll interact with script (.R) files that are run within a RStudio project file (.Rproj) that uses relative filepaths.
   * Before a repository can be downloaded, a user must have a GitHub account (see “[Prerequisites](https://resources.github.com/github-and-rstudio/)” subsection) and have Git and RStudio installed on their computer (see “[Install Git and RStudio](https://resources.github.com/github-and-rstudio/)”)
2. **Launch the** **“CreelScheduleGenerator”**
   * Open the “Creel\_Schedule\_Generator” folder
   * Copy and paste the file “Creel\_Schedule\_Generator (Template Script).R” into the *user\_scripts* sub-folder
   * Re-name the .R file to your liking. Recommended structure *CreelSchedule\_Creel Project Name\_Start Date\_End Date* (e.g., *CreelSchedule\_Skykomish Summer Chinook and Game Fish\_May 2022\_July 2022*)
   * Open the “Creel\_Schedule\_Generator.Rproj” file located in the main folder and then open your newly created/named .R file within the user\_scripts sub-folder
   * If creating multiple schedules, feel free to develop sub-folders within the *user\_scripts* sub-folder (e.g., Year folder within a Waterbody folder)
3. **Review and update look-up tables (luts), as needed**
   * Open the *luts* sub-folder and open both files
   * Review the existing list of waterbodies and associated meta-data.
   * As needed, update each file for [missing] waterbodies that will be creel surveyed. For consistency, use the exact spelling of waterbodies and locations as specified in the creel database.
     + For the “lut\_water\_body\_lat\_long\_master.csv” file, GPS coordinates should correspond to the mouth of the waterbody for rivers and center of the waterbody for lake/reservoir
     + For the “lut\_water\_body\_locations\_master.csv” file, all index sites and census effort sections need to be listed. The field “surveyor\_num” denotes which creel technician will (typically) be surveying each section (for census counts) and site (for index counts). If only one surveyor will be completing the census and index count, simply enter “1” for every row within the “surveyor\_num” field. However, if >1 surveyor will be (simultaneously) conducting a count, denote each section/site to an individual surveyor (start at 1 and go up to the total number of surveyors that will be working together to complete an index count circuit).
4. **Interactively run the Scheduler script**
   * The Scheduler script is broken up into 10 sections.
     + Within each section, there are rows of code and most of which simply needs to be run (and thus NOT edited). However, see next bullet.
     + There is also code located within in *source\_files* and is run behind the scenes. Feel free to open these source files to understand how the Scheduler is working but should NOT be edited unless you plan on submitting (i.e., pushing) the changes back to the main repository and submitting a Pull Request
   * Six of the sections (#2, #3, #4, #5, #6, and #8) require users to supply inputs.
     1. Each user input is denoted using the prefix “ui\_” and specific instructions are provided
     2. There are also several “seed.number” objects throughout the script. These are used so that the randomized schedule can be duplicated (see “mysample” functions within the “01\_Load\_Functions” source file. Users can change any of the “seed.number[s]” as the actual numbers are irrelevant.
   * After the final line of the script is run, a creel schedule will have been generated and saved as an spreadsheet in the *output* sub-folder.
   * Similar to files created in the *user\_scripts* sub-folder, creel schedule output files can be re-named and organized within additional sub-folders as users see fit
5. **Review final schedule**
   * The Scheduler creates a schedule that denotes creel dates, start and end times (by shift), index count times, index count start locations/direction, and identifies census count survey date/times
   * The Scheduler does not attempt to assign individual creel technicians to the schedule. This must be done by the Project Lead.
   * There can be some flexibility in amending the final creel schedule but Project Leads should consult their Study Design Lead before doing so.