**MainScript\_UsingStableIDs**

* Script Execution
* Introduction and purpose of the script
* Concept Implemented
* Break down of the script
* Validations and Exceptions
* List of Objects

**Script Execution:**

This is a normal simple perl script which can be executed in the command line with the below parameters,

***perl xxxxx.pl arg1 arg2 arg3***

arg1->(1-22,x,y)

arg2->(Starting\_Coordinate)

arg3->(Ending\_Coordinate)

*Example : perl xxxxx.pl 12 63840524 80294266*

**Introduction and purpose of the script:**

The purpose of the script is to convert or map the GRch38 co-ordinates to the older assembly (GRch37) to its corresponding region. The perl script receives the co-ordinates in the above format and finally returns the mapped co-ordinates which refers to the older assembly (GRch37)

**Concept Implemented:**

The script works on the base line of the **stable identifiers** present for Genes, Transcripts and Exons. At first the script validates the input and places the region covered by the coordinates in a Slice. Then the Slice is being fetched up for the stable identifiers in terms of Genes, Transcripts and Exons. The StableIDs of Genes, Transcripts and Exons are stored in different array variables.

We now connect to the older assembly GRch37 database and get the seq\_region\_start and seq\_region\_end for all the fetched stable IDs (Genes, Transcript and Exons). All values of seq\_region\_start and seq\_region\_end is stored in two different arrays. Finally the minimum value of the seq\_region\_start and the maximum value of the seq\_region\_end gives the mapped coordinates or corresponding region in GRch37.

**Break down of the script :**

* Connecting to GRch38 Registry
* Creating Slice using the input
* Creating adaptors for Gene, Transcript and Exons
* Fetching and storing the Stable\_IDs in arrays
* Clearing the GRch38 registry

We initially connect to GRch38 assembly using the registry. We create a slice for the input and also create separate adaptors for Gene, Transcripts and Exons. Using the adaptors we fetch Stable\_IDs in a loop and store them in the arrays. We clear the GRch38 registry because running multiple registries within single script at a time can cause an impact.

* Connecting to GRch37 assembly
* Creating adaptors for Gene, Transcript and Exons
* Fetching seq\_region\_start and seq\_region\_end
* Manipulation of starting and ending coordinates

Now we connect to the GRch37 assembly and create separate adaptors for Gene, Transcript and Exons. Using the adaptors we now get the seq\_region\_start and seq\_region\_end for all the StableIDs obtained from the slice. We have 2 arrays now, one is for seq\_region\_start and another for seq\_region\_end. The minimum value of the seq\_region\_start is obtained using min function and the maximum value of the seq\_region\_end is obtained by max function which are considered as the mapped target coordinates

**Validations and Exceptions:**

*Input Validation :* The input is validated and throws a alert message for invalid input

*Array Validation :* The arrays which store StableIDs are validated by the if statement before it enters the loop or manipulation. If the variable is empty it does not enter the particular loop ignoring it.

The arrays which store seq\_region values are also validated by if statement. If the array is empty it throws message as “Region does not contain any Gene/Transcript/Exon in GRch37 assembly”

*Exceptions :* We might get some exceptions when running the loops in GRch37 assembly. This is because the entire object itself (Gene/Transcript/exon) might be missing in the target assembly. We catch such exceptions using eval for further analysis and proceed up with the loop.

**List of used objects :**

*Adaptors* : Slice, Gene, Transcript, Exon

*Methods* : fetch\_all\_by\_Slice(), fetch\_by\_region, get\_adaptor(), stable\_id(), fetch\_by\_stable\_id(), seq\_region\_start(), seq\_region\_end(), min(), max()

*Array Variables :*

@Gene\_Stable\_IDs - Stores the StableIDs of Genes in GRch38

@Transcript\_Stable\_IDs - Stores the StableIDs of Transcripts in GRch38

@Exon\_Stable\_IDs - Stores the StableIDs of Exons in GRch38

@tgt\_start – Stores the seq\_region\_start() for all StableIDs in GRch37

@tgt\_end - Stores the seq\_region\_end() for all StableIDs in GRch37