Prima regola

Prendi tabella, per ogni riga con nome familiregister. Crea un familyregister per il metamodello di Ufamily. Nel do prendiamo il family register, in questo familyregister prendiamo tutti gli elementi “Family” con l’helper filterlist. Con l’helper checkcellsnot controlliamo se la famiglia è certa o meno. Se è certa fai rowtofamily, se è incerta rowtoUfamily.

Row to family

Generale crea la famiglia certa

Prende ogni riga della tabella e crea una famiglia del metamodello ufamily con l’attribiuto nome uguale al nome della famiglia nella riga. Nel do data una famiglia prendiamo tutte le righe rappresentanti tutti gli attributi di quella famiglia. “parseString” ci permette di eliminare le righe che finiscono con familyInverse e name. Eliminiamo anche la riga “FamilyFalnders” con il primo if.

PER MADRE E PADRE: A questo punto vediamo se è una madre, padre ecc. l’helper checkcells vede se per ogni cella il primo elemento è indefinito. Con il terzo if controlliamo se il membro è presente almeno in una cella. Nel quarto if controlliamo se l’id è uguale in tutti, ovvero se il membro è certo o incerto.

PER FIGLIO E FIGLIA:

nel prima if controlliamo se tutte le celle non sono vuote. Nell’helper checklistelement, prendo la prima lista non nulla tramite l’helper returnfirstlistnotnull. Il senso dovrebbe essere controllare che tutte le altre celle abbiano la stessa lista della prima. Se questo è vero abbiamo i figli certi, se non è vero abbiamo i figli incerti (però questo non funziona.

RowToUfamilies

In generale crea una famiglia incerta

Prendiamo le righe, creaiamo una UFamilydel metamodello Ufamily con l’attributo nome uguale al nome. Nel do prendiamo tutte le righe tranne quelle che finiscono con familyinverse e name. Nel primo if non prendiamo la riga “familyflanders.5”. se tutte sono vuote non fai nulla, se almeno una è non vuota faccio il checkID. Se gli ID sono tutti uguali allora il padre è certo se invece sono diversi allora il padre è incerto.