## Adding a new activity:

In order to demonstrate these changes, we show how to add an activity called "*NewType*" The changes required in the config file is a single line as shown below:

## Step -1: Changes in the config file

In the config file data/simulation.xml, we add an activity type as shown in green below

While the activity names are made configurable, the main types of activities are <u>NOT</u>. The types of activities as specified in the config file simrun\_Midterm.xml must be the same as the of activities as hardcoded in the dev/Basic/shared/StopType.hpp, as shown below:

```
StopType NULL_STOP = -1;
StopType WORK_ACTIVITY_TYPE = 1;
StopType EDUCATION_ACTIVITY_TYPE = 2;
StopType OTHER_ACTIVITY_TYPE = 3;
StopType WORK_BASED_SUBTOUR = 5;
```

## Step - 2: Changes in lua files:

dps.lua:

i) The activity type map should be updated: The new key for "NewType" is added as shown in green below:

ii) Inside function compute\_utilities(), add a new variable for the logsum of the new activity, shown in green.

```
local otherlogsum = params:activity_logsum(activity_types.Others)
local otherlogsum = params:activity_logsum(activity_types.Others)
local newTypeLogsum = params:activity_logsum(activity_types.NewType)
```

iii) The choice set should be altered. For example the choice set can be expanded to include 5 activity types as shown in green below:

Changes same as dps.lua

\_\_\_\_\_\_

dpb.lua:

i) The activity type map should be updated: The new key for "NewType" is added as shown in green below:

ii) Inside function compute\_utilities(), add a new variable for the logsum of the new activity, shown in green.

```
local otherlogsum = params:activity_logsum(activity_types.Others)
local otherlogsum = params:activity logsum(activity types.Others)
```

\_\_\_\_\_\_

isg.lua

```
i) Updating the choice vector. The additions are shown in green 
--choice set
```

```
--1 for work; 2 for edu; 3 for shopping; 4 for other; 5 for quit, 5 for NewType, 6 for quit

local choice = {
    1,
    2,
    3,
    4,
    5,
    6
}
```

ii) Inside compute\_utlities function(): The lines in red to be deleted and the lines in green should be added as shown below:

```
utility[4] = beta_cons_other + beta_work_tour_dummy_0 * 1 ......
utility[5] = beta cons other+beta work tour dummy 0 * 1.....
```

Earlier, The utility[5] represented "quit". But after adding a new mode, utility[6] represents for "quit".

```
utility[4] = beta_cons_other + beta_work_tour_dummy_0 * 1 ......
utility[5] = beta_cons_other+beta_work_tour_dummy_0 * 1.....
utility[6] = beta_cons_other+beta_work_tour_dummy_0 * 1.....
```

3) Inside the compute\_availabilities() function, the range of the for loop shoud be changed :

```
for i = 1, 5 do
for i = 1, 6 do
```

\_\_\_\_\_\_

## Step- 3: Changes in the logsum table

The logsum table should have ( N+4 ) columns, where N is the number of Activity types allowed. For example when we have a total of 5 allowed activities:

Work, Education, Shopping, Other, and New Type , the create table query of the logsum table should look as shown below:

```
CREATE TABLE demand.preday_logsum_test_new_activity_aditi
(
   person_id bigint NOT NULL,
   work double precision,
   education double precision,
   shop double precision,
   other double precision,
   newtype double precision,
   dp_tour double precision,
   dp_stop double precision,
   CONSTRAINT preday_logsum_test_new_activity_aditi_pkey PRIMARY KEY (person_id)
```

<u>Important Consideration:</u> The order of the Activity names must be in accordance with the order specified in the simrun\_MidTerm.xml config file