Challenge April-2019
Recreational Fee

A solution with DT5GL[1] by Jack Jansonius – 8 May 2019

Problem description (from the website):
A city has created a decision table to determine appropriate usage fees for its recreational facilities based on length of usage and when the usage occurs:

What recreational fee should be charged for use of facilities?	*When does the usage occur?*			
How long is the usage?		weekday	weekend during off hours	weekend during peak hours
	< 1 hour	$2	$3	$8
	1-2 hours	$3	$5	$10
	> 2 hours	$5	$8	$15

The city also has the following behavioral business rule:
A senior citizen must not be charged a recreational fee for use of facilities.

Implementation of the decision tables in DT5GL:

```
Table 0: Recreational Fee
If:                                    | 0| 1| 2| 3| 4|
'Policy: Special treatment of seniors' | Y| Y| Y| N| N|
'senior'                               | Y| N| N| -| -|
calculatedprice > 0                    | -| Y| N| Y| N|
Then:
recreational_fee is special(senior)    | X|  |  |  |  |
recreational_fee is regular            |  | X|  | X|  |
recreational_fee is not_determined     |  |  | X|  | X|
# .......

Proposition: 'Policy: Special treatment of seniors'
Askable_using: "**?"

Attribute: calculatedprice
Summation_of: weekdayprice + weekend_off_price + weekend_peak_price

Table 1: Senior
If:                                    | 0| 1|
age >= 65                              | Y| N|
Then:
'senior'                               | X|  |
# .......

Attribute: age
Askable_using: "What is the years of age of the recreationist?"
```

[1] Powered by Python 3.6.0

```
Table 2: weekdayprice
If:                                  | 0| 1| 2| 3|
moment_of_usage is weekday           | Y| Y| Y| N|
usage_time <  60                     | Y| N| N| -|
usage_time < 120                     | -| Y| N| -|
Then:
weekdayprice = 2                     | X|  |  |  |
weekdayprice = 3                     |  | X|  |  |
weekdayprice = 5                     |  |  | X|  |
# .......

Attribute: moment_of_usage
Askable_using: "What is the moment of usage?"
Attribute: usage_time
Askable_using: "How long (in minutes)?"

Table 3: weekend_off_price
If:                                  | 0| 1| 2| 3|
moment_of_usage is weekend_off_hours | Y| Y| Y| N|
usage_time <  60                     | Y| N| N| -|
usage_time < 120                     | -| Y| N| -|
Then:
weekend_off_price = 3                | X|  |  |  |
weekend_off_price = 5                |  | X|  |  |
weekend_off_price = 8                |  |  | X|  |
# .......

Table 4: weekend_peak_price
If:                                  | 0| 1| 2| 3|
moment_of_usage is weekend_peak_hours| Y| Y| Y| N|
usage_time <  60                     | Y| N| N| -|
usage_time < 120                     | -| Y| N| -|
Then:
weekend_peak_price = 8               | X|  |  |  |
weekend_peak_price = 10              |  | X|  |  |
weekend_peak_price = 15              |  |  | X|  |
# .......

GoalAttribute: recreational_fee
Case: special(senior)
Print: "-----------------------------------------------------"
Print: "No recreational fee because it's a senior."
Print: "-----------------------------------------------------"

Case: regular
Print: "-------------------------------------------------------------"
Print: "Recreational_fee is: %s." calculatedprice
Print: "Based on:"
Print: "Charge for a weekday: %s." weekdayprice
Print: "Charge for a weekend during off ours: %s." weekend_off_price
Print: "Charge for a weekend during peak ours: %s." weekend_peak_price
Print: "-------------------------------------------------------------"

Case: not_determined
Print: "-----------------------------------------------------"
Print: "Recreational_fee could not be determined"
Print: "-----------------------------------------------------"
```

Some testruns

Session 1:

```
'Policy: Special treatment of seniors' (y/n)? *> y
"What is the years of age of the recreationist?"
> 66
-----------------------------------------------------
No recreational fee because it's a senior.
-----------------------------------------------------
```

Session 2:

```
'Policy: Special treatment of seniors' (y/n)? *> y
"What is the years of age of the recreationist?"
> 64

"What is the moment of usage?"
1. weekday
2. weekend_off_hours
3. weekend_peak_hours
4. otherwise
> 2

"How long (in minutes)?"
> 121
-------------------------------------------------------------
Recreational_fee is: 8.
Based on:
Charge for a weekday: 0.
Charge for a weekend during off ours: 8.
Charge for a weekend during peak ours: 0.
-------------------------------------------------------------
```

Session 3:

```
'Policy: Special treatment of seniors' (y/n)? *> n
"What is the moment of usage?"
1. weekday
2. weekend_off_hours
3. weekend_peak_hours
4. otherwise
> 3

"How long (in minutes)?"
> 59
-------------------------------------------------------------
Recreational_fee is: 8.
Based on:
Charge for a weekday: 0.
Charge for a weekend during off ours: 0.
Charge for a weekend during peak ours: 8.
-------------------------------------------------------------
```

NB. the option "otherwise" has been added by the tool and could easily be suppressed.

Session 4:

```
'Policy: Special treatment of seniors' (y/n)? *> n
"What is the moment of usage?"
1. weekday
2. weekend_off_hours
3. weekend_peak_hours
4. otherwise
> 4
-----------------------------------------------------
Recreational_fee could not be determined
-----------------------------------------------------
```

Variations on a theme: the senior.

Table 1 states that a citizen is a senior when he is 65 years or older. For this rule, it is therefore easy to ask about the age:

```
Table 1: Senior
If:                                  | 0| 1|
age >= 65                            | Y| N|
Then:
'senior'                             | X|  |
# .......
```

This rule can be made stricter by asking for an identity card and reading the age of the card:

```
rTable 1: Senior
If:                                  | 0| 1|
'check based on an identity card'    | Y| N|
age_on_identity_card >= 65           | Y| -|
age >= 65                            | -| Y|
Then:
'senior'                             | X| X|
'remark'                             |  | X|
# .......
```

However, it is still possible to simply ask for the age without reading an identity card; in this case an additional message will appear on the screen (or in the database), by means of :

```
GoalProposition: 'remark'
Print: "Warning: Age is not determined on the basis of an identity card."
```

An example of a test run is then:

```
'Policy: Special treatment of seniors' (y/n)? *> y
'check based on an identity card' (y/n)? *> n
"What is the years of age of the recreationist?"
> 65
--------------------------------------------------
No recreational fee because it's a senior.
--------------------------------------------------
Warning: Age is not determined on the basis of an identity card.
```

But these rules can also be made stricter by taking over (or scanning) the registration number of the identity card and, on this basis, requesting the age in an internal or external identity card application:

```
rTable 1: Senior
If:                                   | 0| 1|
'check based on an identity card'     | Y| N|
registry_number > 0                   | Y| -|
age_based_on_registration_number >= 65| Y| -|
age >= 65                             | -| Y|
Then:
'senior'                              | X| X|
'remark'                              |  | X|
# .......
```

Now is a viable test run:

```
'Policy: Special treatment of seniors' (y/n)? *> y
'check based on an identity card' (y/n)? *> y
"What is the registry number on the identity card of the recreationist?"
> 748874547878          <-- Simulation: goes to the identity card application.

"What is the age based on the registry number of the recreationist?"
> 65                    <-- Simulation: comes from the identity card application.
-----------------------------------------------------
No recreational fee because it's a senior.
-----------------------------------------------------
```

These rules can also be made even stricter by stipulating that only special authorised persons can still bypass the check by means of an identity card:

```
rTable 1: Senior
If:                                             | 0| 1| 2|
'authority to disable check on identity card'   | Y| Y| N|
'check based on an identity card'               | Y| N| -|
registry_number > 0                             | Y| -| Y|
age_based_on_registration_number >= 65          | Y| -| Y|
age >= 65                                       | -| Y| -|
Then:
'senior'                                        | X| X| X|
'remark'                                        |  | X|  |
# .......
```

This will result in 3 test runs:

1:

```
'Policy: Special treatment of seniors' (y/n)? *> y
'authority to disable check on identity card' (y/n)? *> y
'check based on an identity card' (y/n)? *> y
"What is the registry number on the identity card of the recreationist?"
> 8375398739735

"What is the age based on the registry number of the recreationist?"
> 66
-----------------------------------------------------
No recreational fee because it's a senior.
-----------------------------------------------------
```

2:

```
'Policy: Special treatment of seniors' (y/n)? *> y
'authority to disable check on identity card' (y/n)? *> y
'check based on an identity card' (y/n)? *> n
"What is the years of age of the recreationist?"
> 65
-----------------------------------------------------
No recreational fee because it's a senior.
-----------------------------------------------------
Warning: Age is not determined on the basis of an identity card.
```

3:

```
'Policy: Special treatment of seniors' (y/n)? *> y
'authority to disable check on identity card' (y/n)? *> n
"What is the registry number on the identity card of the recreationist?"
> 387837438799

"What is the age based on the registry number of the recreationist?"
> 65
-----------------------------------------------------
No recreational fee because it's a senior.
-----------------------------------------------------
```