# **Proposal:**

**Summary**

* Add support for new rounding modes. Currently only *half even*, *celling* and *flooring* are supported
* There are at least two ways of extending DMN / FEEL:
  + One *round* function with 3 parameters (Java)
  + Several functions, one for each rounding mode (Python)
* This proposal adds several functions

**Details**

The changes are baselined on DMN 1.3 **dtc-19-12-06.pdf**.

**Position:**

**Page 162 Table 76 Semantics of numeric functions**

**Replace rows for** *floor* **and** *ceiling* **with the following**

|  |  |  |  |
| --- | --- | --- | --- |
| floor(*n*)  floor(*n, scale*) | number  number, number1 | Return *n* with given *scale* and rounding mode flooring*.*  If at least one of *n* or *scale* is **null** the result is **null**.  For floor(*n*) the *scale* is **0** | *floor*(1.5) = 1  *floor*(-1.56, 1) = -1.6 |
| ceiling(*n*)  ceiling(*n, scale*) | number  number, number1 | Return *n* with given *scale* and rounding mode ceiling*.*  If at least one of *n* or *scale* is **null** the result is **null**.  For ceiling(*n*) the *scale* is **0** | *ceiling*(1.5) = 2  *ceiling*(-1.56, 1) = -1.5 |

**Insert a new row after the entry that contains the definition of ceiling**

|  |  |  |  |
| --- | --- | --- | --- |
| round up(*n, scale*) | number, number1 | Return *n* with given *scale* and rounding mode round up*.*  If at least one of *n* or *scale* is **null** the result is **null**. | *round up*(5.5, 0) = 6  *round up*(-5.5, 0) = -6  *round up*(1.121, 2) = 1.13  *round up*(-1.126, 2) = -1.13 |
| round down(*n, scale*) | number, number1 | Return *n* with given *scale* and rounding mode round down*.*  If at least one of *n* or *scale* is **null** the result is **null**. | *round down*(5.5, 0) = 5  *round down* (-5.5, 0) = -5  *round down* (1.121, 2) = 1.12  *round down* (-1.126, 2) = -1.12 |
| round half up(*n, scale*) | number, number1 | Return *n* with given *scale* and rounding mode round half up*.*  If at least one of *n* or *scale* is **null** the result is **null**. | *round half up*(5.5, 0) = 6  *round half up*(-5.5, 0) = -6  *round half up*(1.121, 2) = 1.12  *round half up*(-1.126, 2) = -1.13 |
| round half down(*n, scale*) | number, number1 | Return *n* with given *scale* and rounding mode round up*.*  If at least one of *n* or *scale* is **null** the result is **null**. | *round half down* (5.5, 0) = 5  *round half down* (-5.5, 0) = -5  *round half down* (1.121, 2) = 1.12  *round half down* (-1.126, 2) = -1.13 |