Testing v2.5

*Freddy’s Fast Food*

All my validation is built a top of each other so if one of them catches bad user input

The rest of them will be able to catch it as well (for their respective types, for example integer validation will also inherit the length requirement of string validation)

Note: The terminal colours are only supported on linux so they are not added on windows

Validation is done using exceptions named (ValidationError) and called functions which can be specified to the accept method and its child accept\_int, accept\_bool, and accept\_float methods

The Validation class has the following validators

* Int - This validator function ensures that the provided value is an integer number anything else will throw a validation error
* Float - This validation function ensures that the provided value is an integer number anything else will throw a validation error
* Min\_Max\_Int / Min\_Max\_Float - This validator function requires that the passed value is within the minimum and maximum value this validator depends on the Int validator and will throw a validation error if the number is less than or greater than the bounds
* Boolean - This validator requires the value to be acceptable a boolean value in Test 5 there is a table containing the acceptable values for this validation

[Test 1 - *Passing no user input*](#_kyzzsmwz89t2)

[Test 2 - *Passing a non number to an integer input*](#_65c1foqgu4zo)

[Test 3 - *Passing a float number to an integer input*](#_xw4wmmnzofh8)

[Test 4 - *Passing an integer value lower than the minimum*](#_pvzekxgpiw5n)

[Test 5 - *Passing an integer value higher than the maximum*](#_km5mrzyd279e)

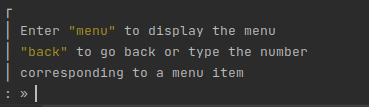
[Test 6 - *Passing an unexpected value to a bool input*](#_vdlczq6s7j13)

[Test 7 - *Passing a long float number to a float input*](#_h0hcyr66tjao)

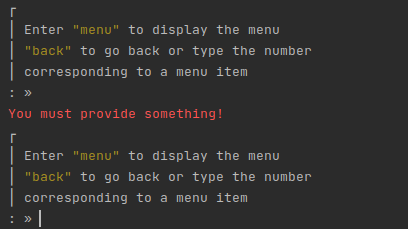
## Test 1

*Passing no user input*

The first test here is an attempt at providing no user input and just pressing enter



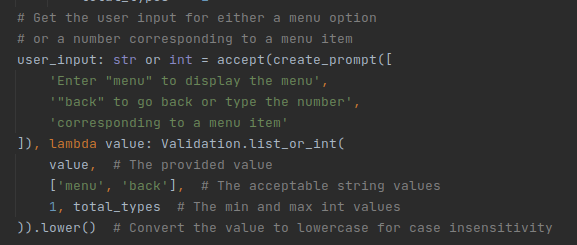
Upon pressing enter the user will be prompted with the following message



The user will be told “You must provide something!” then the same prompt is displayed again (This is because it is looped until the validation passes)

This specific input uses the Validation.list\_or\_int validator which depends on the

Validation.int and Validation.min\_max validators

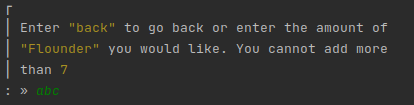


## Test 2

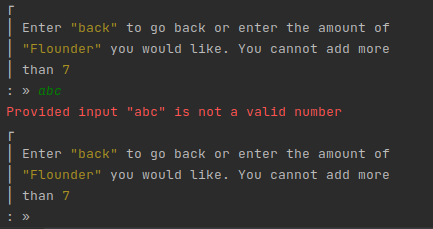
*Passing a non number to an integer input (accept\_int / Validation.int)*

# The input this was tested on requires a number between 1 and 7 (inclusive)

Here the user is attempting to provide the text “abc” to an input that expects an integer



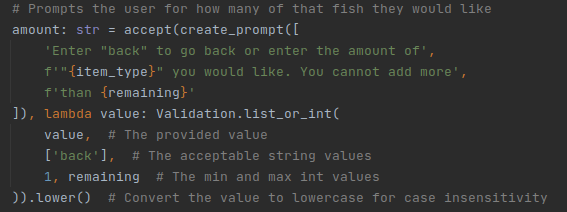
Upon pressing enter the user will be prompted with the following message



The user will be told “Provided input “abc” is not a valid number” then the same prompt is displayed again (This is because it is looped until the validation passes)

This specific input uses the Validation.list\_or\_int validator which depends on the

Validation.int and Validation.min\_max validators



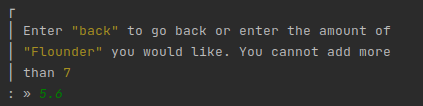
## Test 3

*Passing a float number to an integer input (accept\_int / Validation.int)*

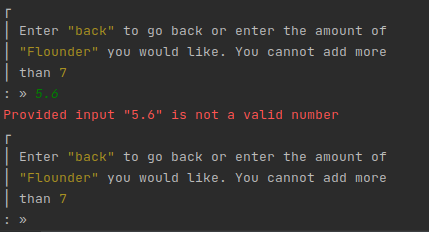
# The input this was tested on requires a number between 1 and 7 (inclusive)

But this only accepts integers

Here the user is attempting to provide the text “5.6” to an input that expects an integer



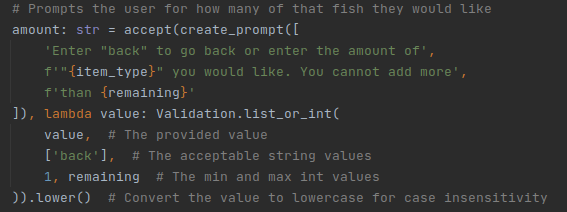
Upon pressing enter the user will be prompted with the following message



The user will be told “Provided input “abc” is not a valid number” then the same prompt is displayed again (This is because it is looped until the validation passes)

This specific input uses the Validation.list\_or\_int validator which depends on the

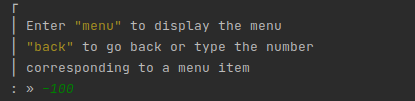
Validation.int and Validation.min\_max validators



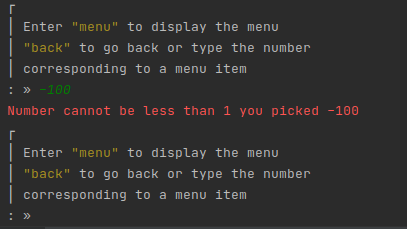
## Test 4

*Passing an integer value lower than the minimum (accept\_int / Validation.int / Validation.min\_max\_int)*

Here the user is attempting to provide the text “-100” to an input that expects an integer greater than or equal to 1



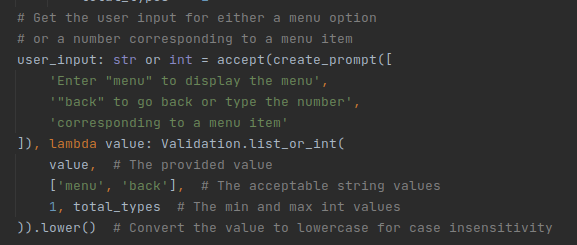
Upon pressing enter the user will be prompted with the following message



The user will be told “Number cannot be less than 1 you picked -100” then the same prompt is displayed again (This is because it is looped until the validation passes)

This specific input uses the Validation.list\_or\_int validator which depends on the

Validation.int and Validation.min\_max validators

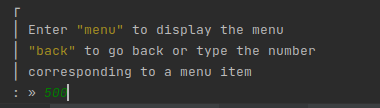


## Test 5

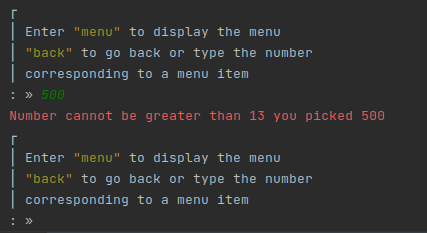
*Passing an integer value higher than the maximum (accept\_int / Validation.int / Validation.min\_max\_int)*

# Because min\_max\_int validation occurs after int validation there is no need to worry about the number being a float and going under the limit

Here the user is attempting to provide the text “500” to an input that expects an integer less than or equal to 13



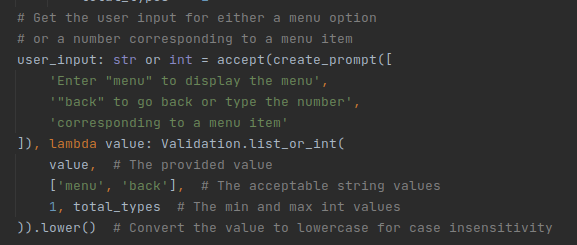
Upon pressing enter the user will be prompted with the following message



The user will be told “Number cannot be greater than 13 you picked 500” then the same prompt is displayed again (This is because it is looped until the validation passes)

This specific input uses the Validation.list\_or\_int validator which depends on the

Validation.int and Validation.min\_max validators



## Test 6

*Passing an unexpected value to a bool input (accept\_bool / Validation.boolean)*

The boolean validation (Validation.boolean) accepts the values in this table

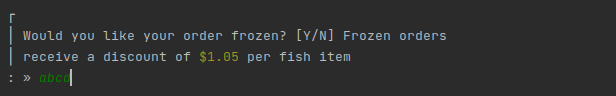
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Yes | y | yes | t | true | 1 |
| No | n | no | f | false | 0 |

*Anything not in this table will be considered invalid input*

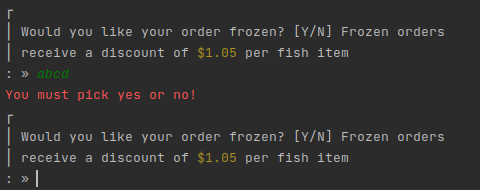
*(This table is case insensitive)*

# Because min\_max\_int validation occurs after int validation there is no need to worry about the number being a float and going under the limit

Here the user is attempting to provide the text “abcd” to an input that expects a value from the boolean table

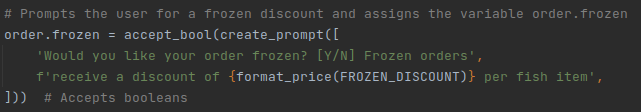


Upon pressing enter the user will be prompted with the following message



The user will be told “You must pick yes or no!” then the same prompt is displayed again (This is because it is looped until the validation passes)

This specific input uses the accept\_bool acceptor which uses the Validator.boolean validation

******

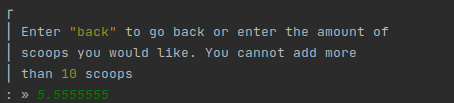
## Test 7

*Passing a long float number to a float input (accept\_float / Validation.float)*

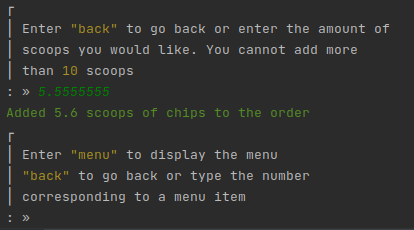
# In places where float values are accepted from the user rounding to 1dp is in place

To prevent long numbers from being stored

Here the user is attempting to provide the text “5.5555555” to an input that expects a float value



Upon pressing enter the user will be prompted with the following message



The user input will be accepted however the provided value will be rounded to 1 decimal place

This specific input uses the Validation.list\_or\_float validator which depends on the

Validation.float and Validation.min\_max validators

