Intermediate Designer 1: Special Qs, Validation, and C#

Overview

- Special question types (GPS, barcode, picture)
- Data validation
- Basic C# syntax for creating validation and enablement conditions

Learning Objectives

- Know what validation conditions and messages are
- Apply basic C# syntax for creating validation and enablement conditions

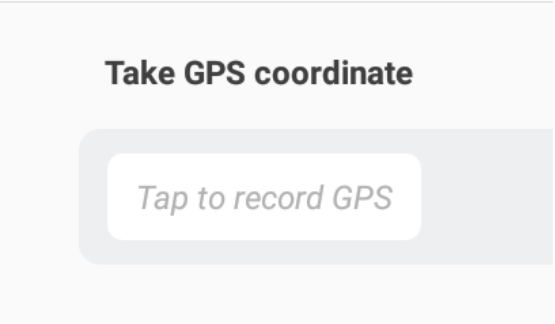
Special question types

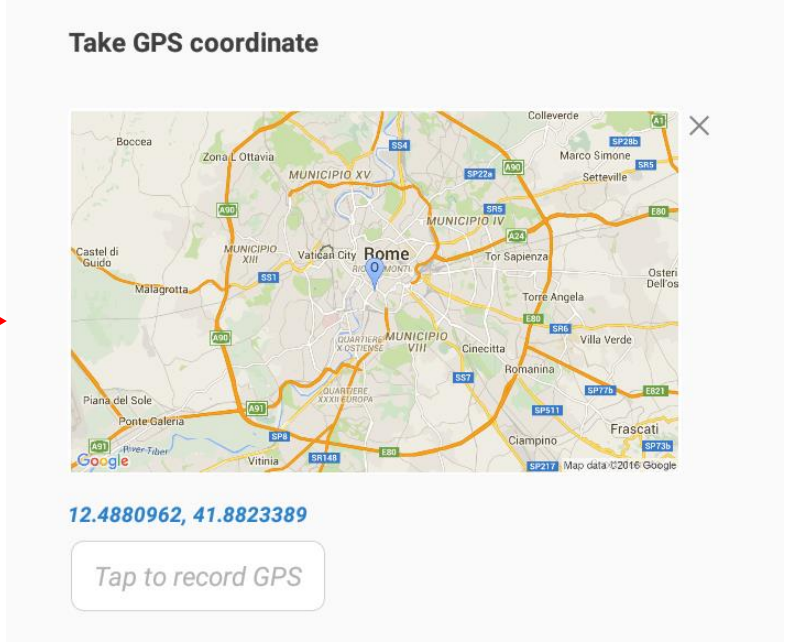
Special question types

- 3 types for now:
  - GPS
  - Barcode
  - Picture

Special question types: GPS

Designer





Interviewer

Special question types: Barcode

Question Type

QR Barcode

Designer

Barcode of the gift given to the household

Scan

Interviewer

Barcode of the gift given to the household

9781464801334

Scan

Special question types: Photo

Designer

Question type

Picture

Interviewer

Photo graph product

Tap to take a photo

Photo graph product

Tap to take a photo

Data validation

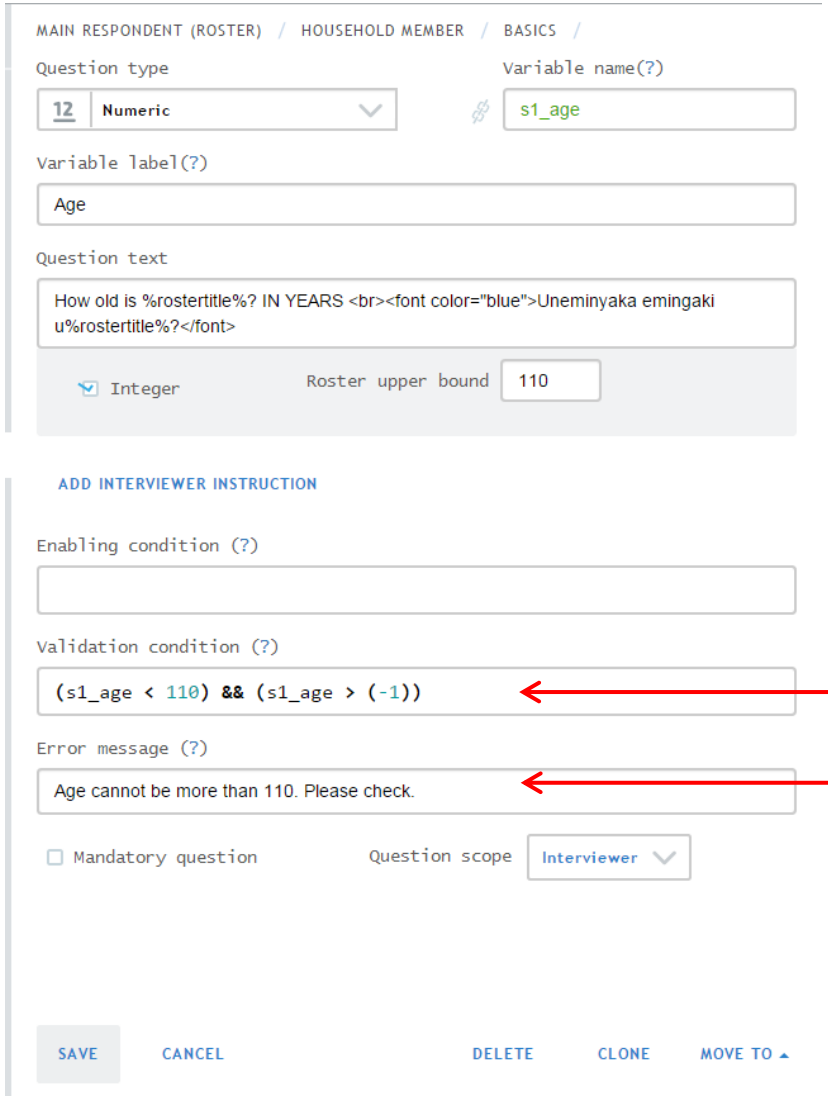
Data validation: description

- Data validation expressions are to:
  - Confirm responses are consistent, and/or responses meet realistic expectations
  - Examples
    - A man cannot be pregnant. **Consistent w/ gender, enabled when household member is male.*
    - No one can be more than 110 years old, and younger than 0. **Realistic expectations.*
- These expressions are written in C# in the 'Validation condition field.

Data validation: description

- What happens when validation is violated?
- A warning message is shown to enumerator

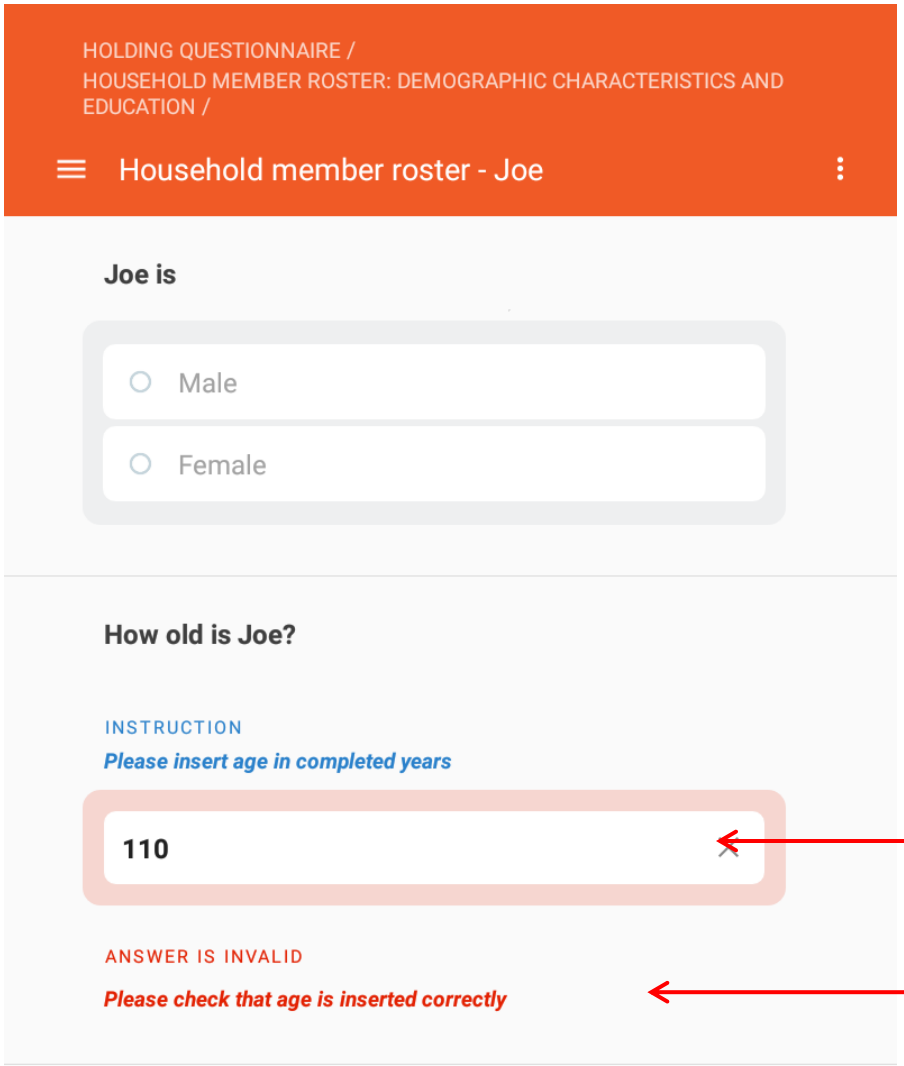
Data validation: description



C#

Message enumerator sees if condition violated

Data validation: description



Value that violates condition

Message enumerator sees if condition violated

Basic C# syntax for creating validation and enablement conditions

Data validation: Building expressions

- How?
  - Connect a variable name with values or another variable using logical operators.
  - Connect expression creating multiple conditions using “||” or “&&”
- Example

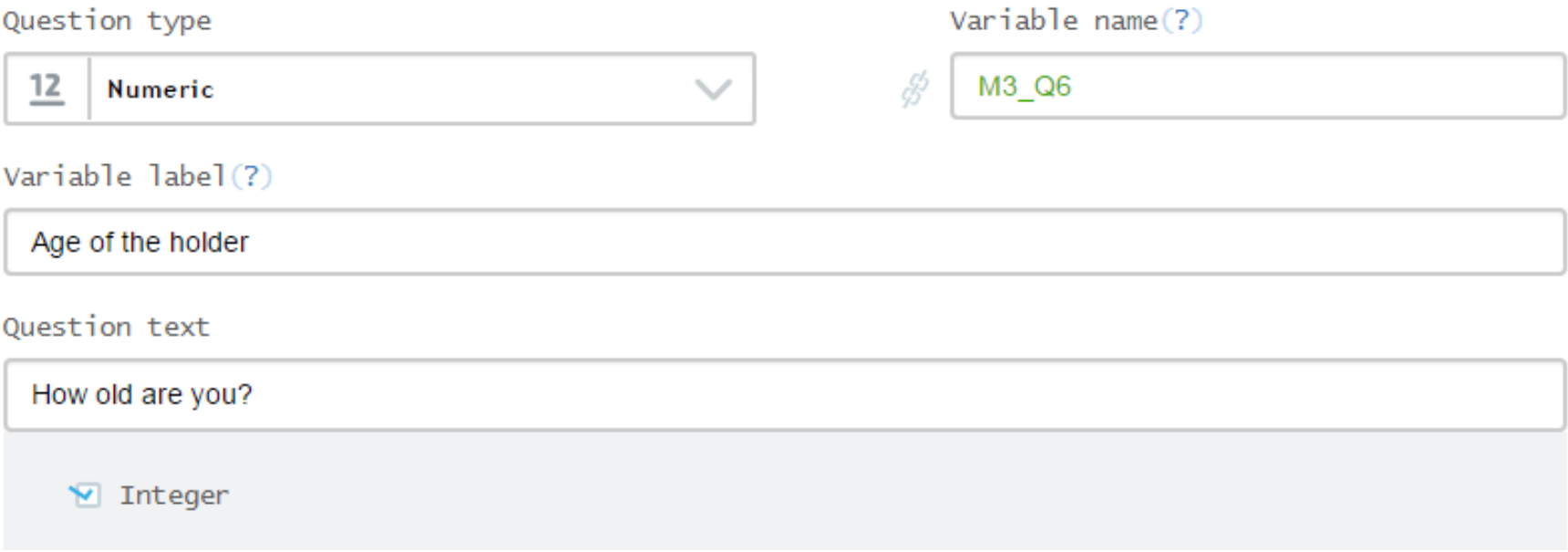
  age > 18 || age < 65

Data validation: Operators

Operator	Description
>	Greater than
<	Less than
==	Equal too
!=	Not equal to
<=	Less than or equal to
>=	Greater than or equal to
&&	and
||	or
variableName==1 || variableName==2	(**single select**) contains in
new decimal?[]{1,2}.Contains(variableName)	(**single select**) contains in
variableName.Contains(1)	(**multiselect**) single values
variableName.Intersect(new decimal[] {1, 3}).Any()	(**multiselect**) single values, multi values

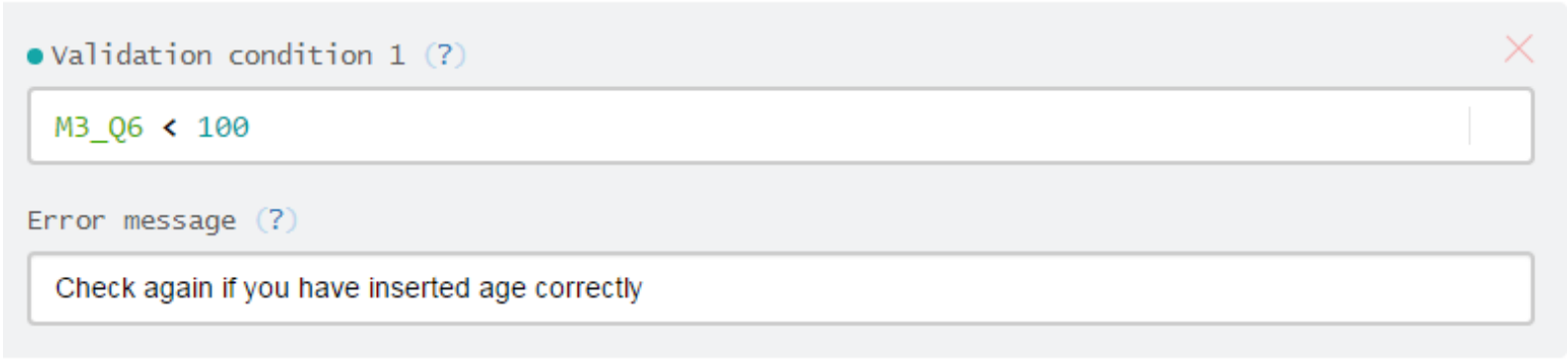
Operators depend on type of UNLEASH QUESTION!

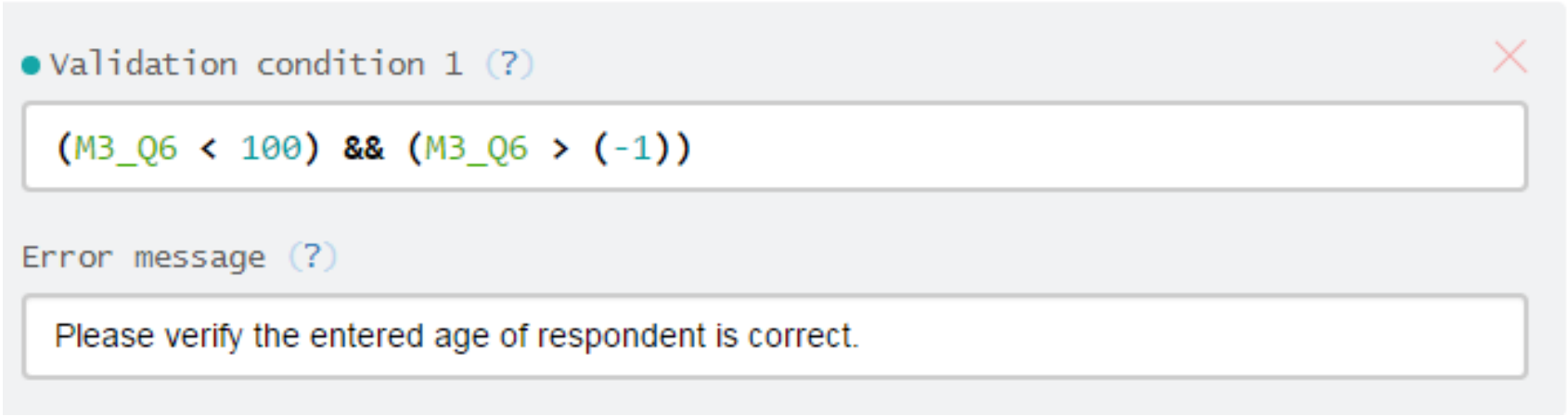
Data Validation examples



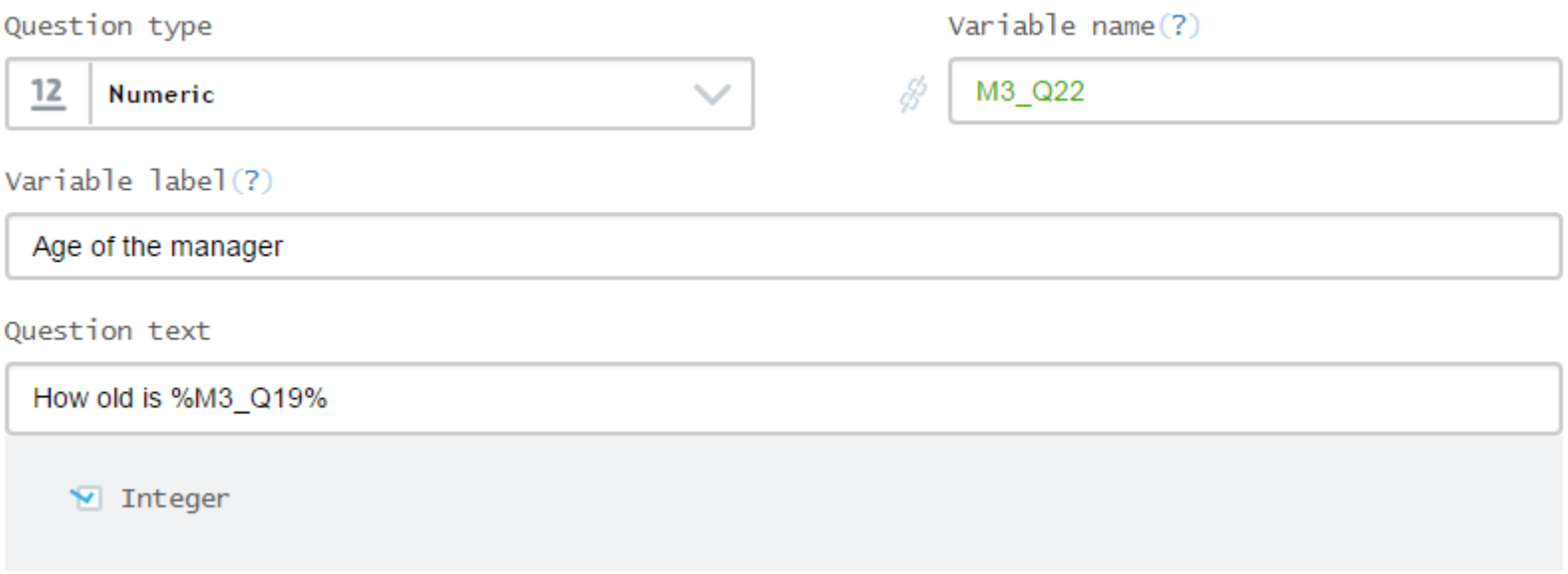
Data Validation examples

- Validation condition, we want the response to be less than 100 and positive.





Enablement and validation

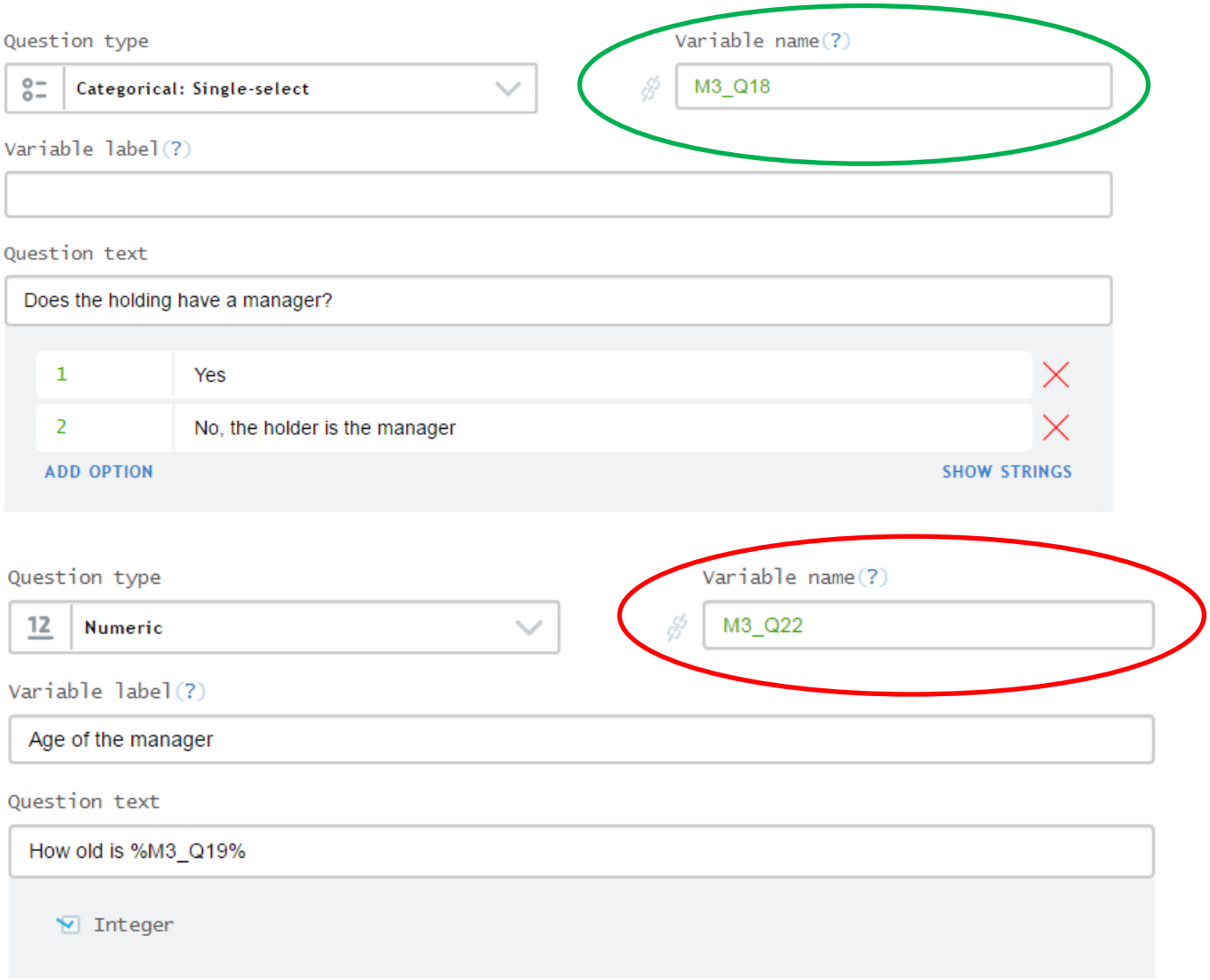


KEY CONCEPT AHEAD

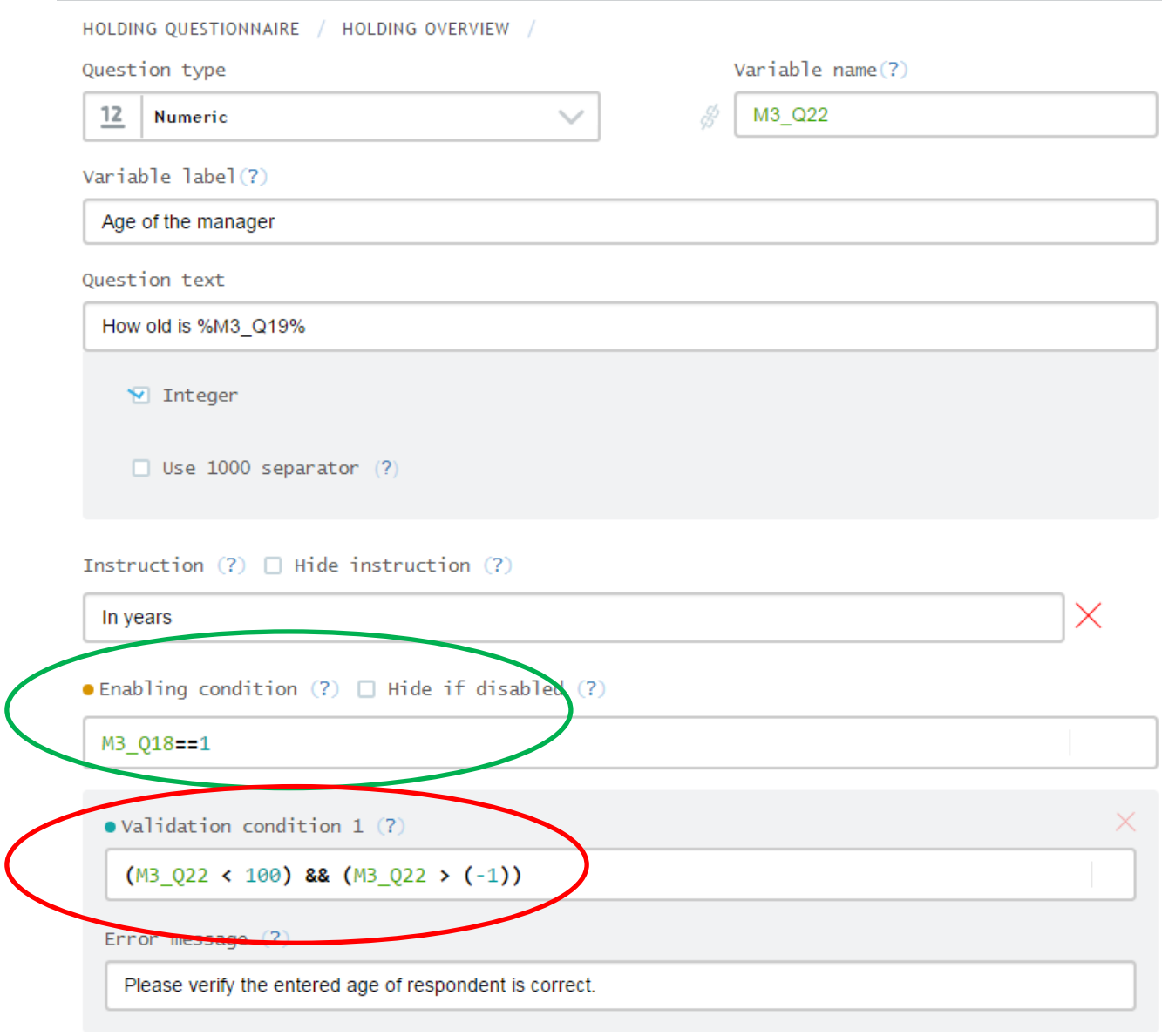
Enablement and validation

- **Enable** only if there is a manager
- **Validate** that value is less one 100 and positive.

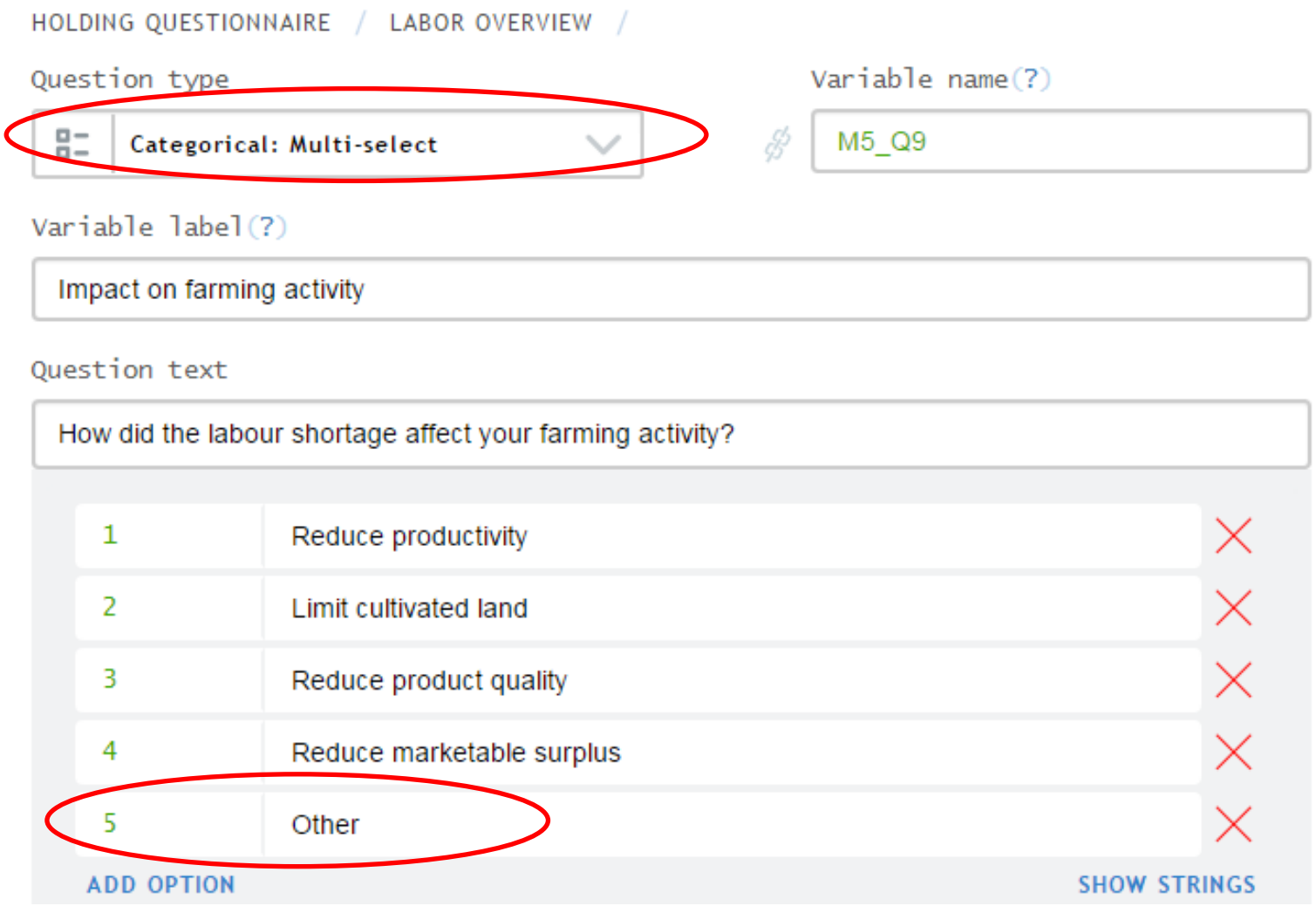
Enablement and validation



Enablement and validation



Enablement: Another Example



Data validation: Operators

Operator	Description
>	Greater than
<	Less than
==	Equal too
!=	Not equal to
<=	Less than or equal to
>=	Greater than or equal to
&&	and
||	or
variableName==1 || **variableName**==2	(**single select**) contains in
new decimal?[]{1,2}.Contains(**variableName**)	(**single select**) contains in
variableName.**Contains(1)**	(**multiselect**) single values
variableName.**Intersect(new decimal[] {1, 3}).Any()**	(**multiselect**) single values, multi values

Enablement and validation

Questions?