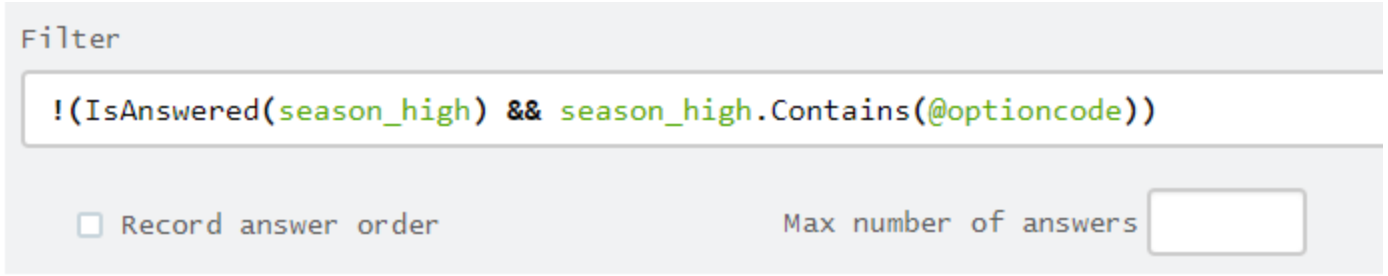
C# usage in Survey Solutions

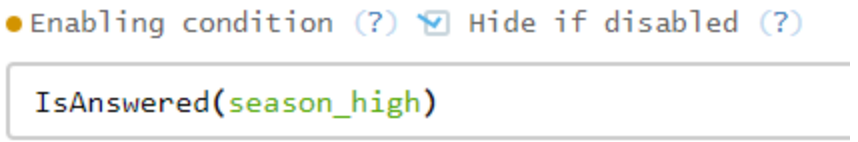
Viktor Gladkikh

Alexandra Komarovskaya

Introduction

- What is the programming language and what it stands for
- What benefits provide usage of programming languages for CAPI
- C# language to power calculation of conditions and validations





Objects

- Questionnaire in Survey Solutions consist of objects
  - Rosters
  - Questions
  - User defined variables, lookups, macroses…
- Each object has attributes and methods for administration
  - Variable name
  - Question text/Section title, question specific properties
- Answers on question has different type from question type

Object types

- Simple types:
  - String
    - “some text”
  - Int
  - Double
    - 3.14159
  - Boolean
    - True/false
- Complex types:
  - DateTime
  - Geo location
  - Audio
  - Yes/No answer
  - Text list question
- Arrays
  - [1, 2, 3, 4]
- Array can be accessed by index
- Index start from zero
  - Household[1]
  - Household[2]

Complex types and rosters

- Complex types like DateTime and GPS answer consist of other objects
- To access properties of complex type answer C# syntax use **dot**

- So if we have GPS question `Gps`
  - Then we access it's properties via dot
  - `Gps.Latitude`
  - `Gps.Longitude`
  - `Gps.Accuracy`

- Roster items are also complex types
- Use dot to access roster item variables
  - `Household[0].age`
  - `Household[0].sex`

Operators

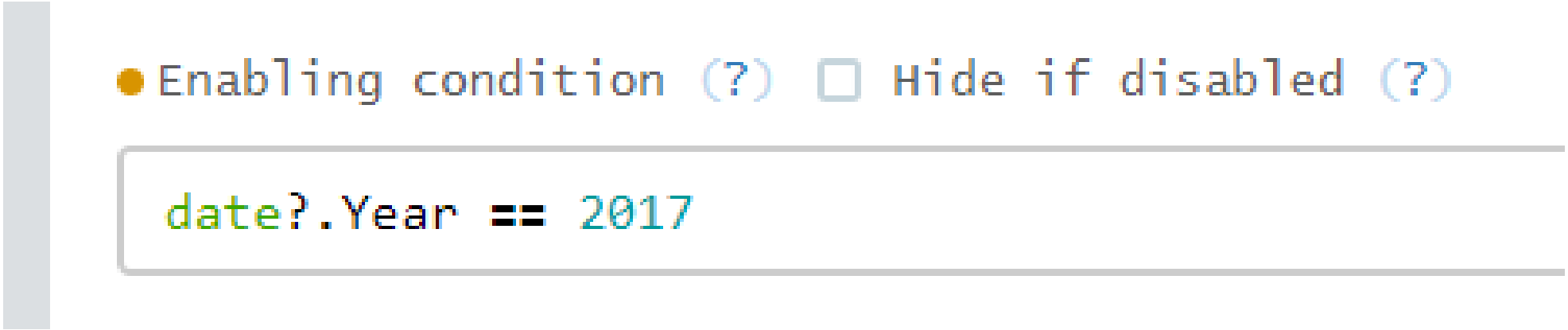
- Relational
  - Equal to: `==`
  - Greater, less than:
    `>` , `>=` , `=<` , `<`
  - Not equal to: !=
- Logical
  - And: &&
  - Or: ||
  - Contrary: !
  - If-then: `exp1 ? exp2 : exp3`

- Mathematical
  - Addition: `x + y`
  - Subtraction: `x - y`
  - Division: `x / y`
  - Multiplication: `x * y`
  - Many other operators: C# Math library

- Object instantiation:
  - new DateTime(2018, 05, 30)

Nullability

- Answer on question that is not answered will be **NULL**
- Integer, double and DateTime type cannot be null
- In Survey Solutions answers of those types wrapped in nullable form
- Accessing for value is happen through `.Value` notation
- "Shorthand" is `?. notation`



Functions

- Function is something that take an input and produce an output
- Function intakes one or more arguments and returns single result

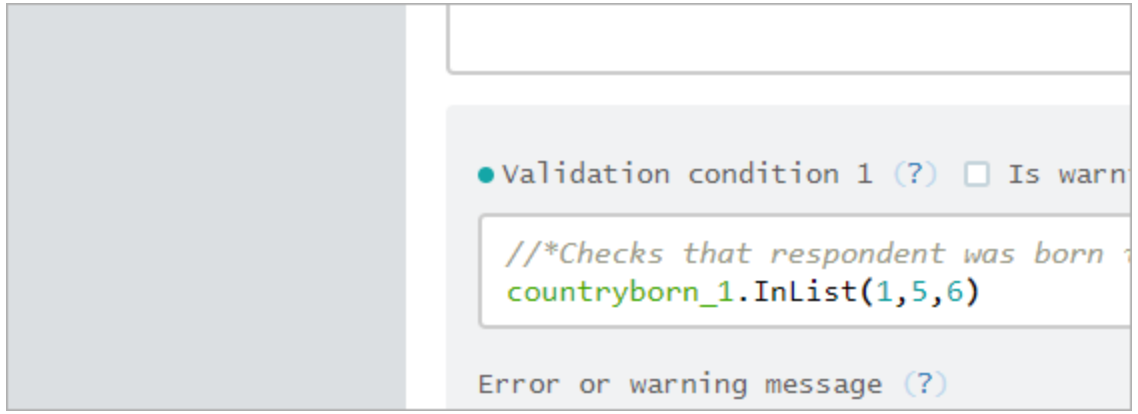
```
function(x){ x + 1 }
```

```
function(2) => 3
function(0) => 1
```

Functions

- Functions can have several arguments
- In C# notation function syntax consists of function name, function return type, list of arguments with specified type and a function body

```
int Max(int a, int b)
{
    return a > b ? a : b;
}
```



Type conversion

- In C# one type can be converted into another
- What You need is parenthesis and proper type name
  - `(decimal?) DayOfVisit?.Year`
- Why You needed:
  - Nowadays it's quite rare operation
  - Most of the type conversions happen automatically
  - But sometimes we need explicit type conversion

```
// date of interview must come after, and thus be greater than, DOB
date_q1 > (new DateTime( (int) date_q2_year, (int) date_q2_month, (int) date_q2_day ))
```

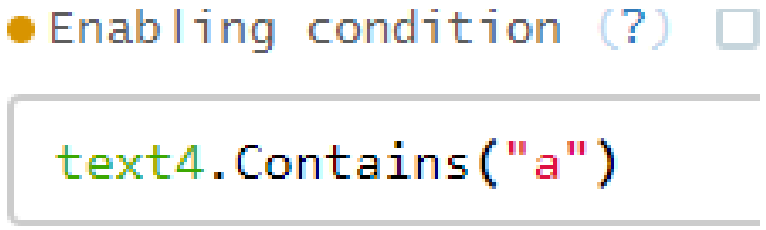
Comments

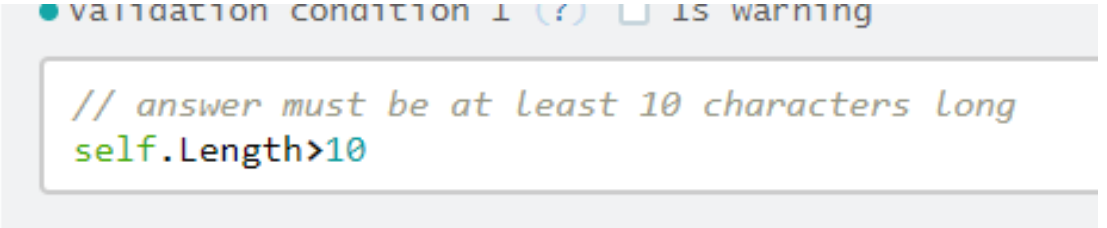
- It's helpful to comment what do particular number code means
- Two types of comments:
  - `//` - will comment whole line. Text after `//` will be ignored
  - `/* some text */` inline block of comments

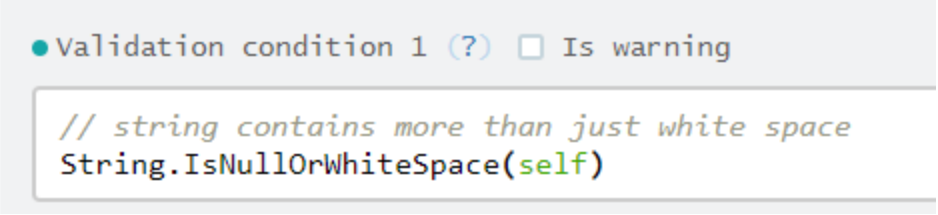
```
// either "none" is selected and no other option...
self.ContainsOnly(1)

||

// ...or there is one or more option selected but it's not "none"
( self.Length>=1 && !self.Contains(1 /* none */) )
```

Text functions

- Answer on text question has type String
- Properties
  - Q.Length
- Functions:
  - Q.Contains(string value)
  - Q.ConsistsOf(string chars)
  - Q.IsAlphaLatin()
  - Q.StartsWith/EndsWith(string val)
  - Q.IsNumber()
- There is also helper functions inside String object
  - String.IsNullOrWhiteSpace(Q)
- Barcodes are strings







Numeric functions

- Comparison to a range or values
  - `Q.InRange(a, b)`
    - Return true if `a <= Q <= b`
  - `Q.InList(a,b,c)`
    - Return true if Q equal to a, b or c
- `Math` object contains a lot of useful mathematic functions
  - Abs, Floor, Max, Min, Pow, etc…
  - You can refer to Survey Solutions Support portal or C# Math class description

Categorical questions

- For single select questions, answer is numeric value
- For Multi-Select question answer is an array of selected codes
- Common functions are:
  - `Q.Contains(value)`
  - `Q.ContainsAll(v1, v2, …)`
  - `Q.ContainsOnly(v1, v2, …)`
  - `Q.ContainsAny(v1, v2, …)`

- Categorical Yes/No question
- Answer on this question is complex type
- It consist of four main parts
  - `Q.Yes`. Contains the values with “yes” answer
  - `Q.No`. Contains the values with “no” answer
  - `Q.Missing`. Contains values without answer
  - `Q.All`. Contains all values answered or not
- All those properties are arrays of codes
- You can use any common functions for single select question

Yes no sample

Question text

FOR A PARTICULAR ITEM
Does your household own any ... ?

501	Mortar/pestle (mtondo)
502	Bed
503	Table
504	Chair
505	Fan

Validation condition 1 (?) Is warning

```
/* however poor, owns a bed */

// are all items answered (aka does the array Missing have any item values)?
self.Missing.Any() ?

// if yes, return true--that is no validation error--for now
true :

// if no, determine whether item 502 (bed) has a "yes" answer
self.Yes.Contains(502)
```

Error or warning message (?)

Unlikely answer. Most households have a bed

DateTime

- Methods
  - `FullYearsSince(DateTime? Date)`
  - `InRange(DateTime? dateA, DateTime? dateB)`
- Properties
  - `Q?.Year, Q?.Month, Q?.Day,`
  - `Q?.Hour, Q?.Minute, Q?.Second`
  - `Q?.DayOfYear`

```
date?.Year == 2017
```

```
date.Value.Year > 2017
```

GPS Location

- Properties:
  - `Latitude`, `Longitude`, `Accuracy` and `Altitude`
- Functions:
  - `Q.InRectangle(north, west, south, east)`
  - `Q.GpsDistance(gps)` distance from `Q` to `gps` in meters
  - `Q.GpsDistanceKm(gps)` distance from `Q` to `gps` in kilometers

Validation condition 1 (?) Is warning

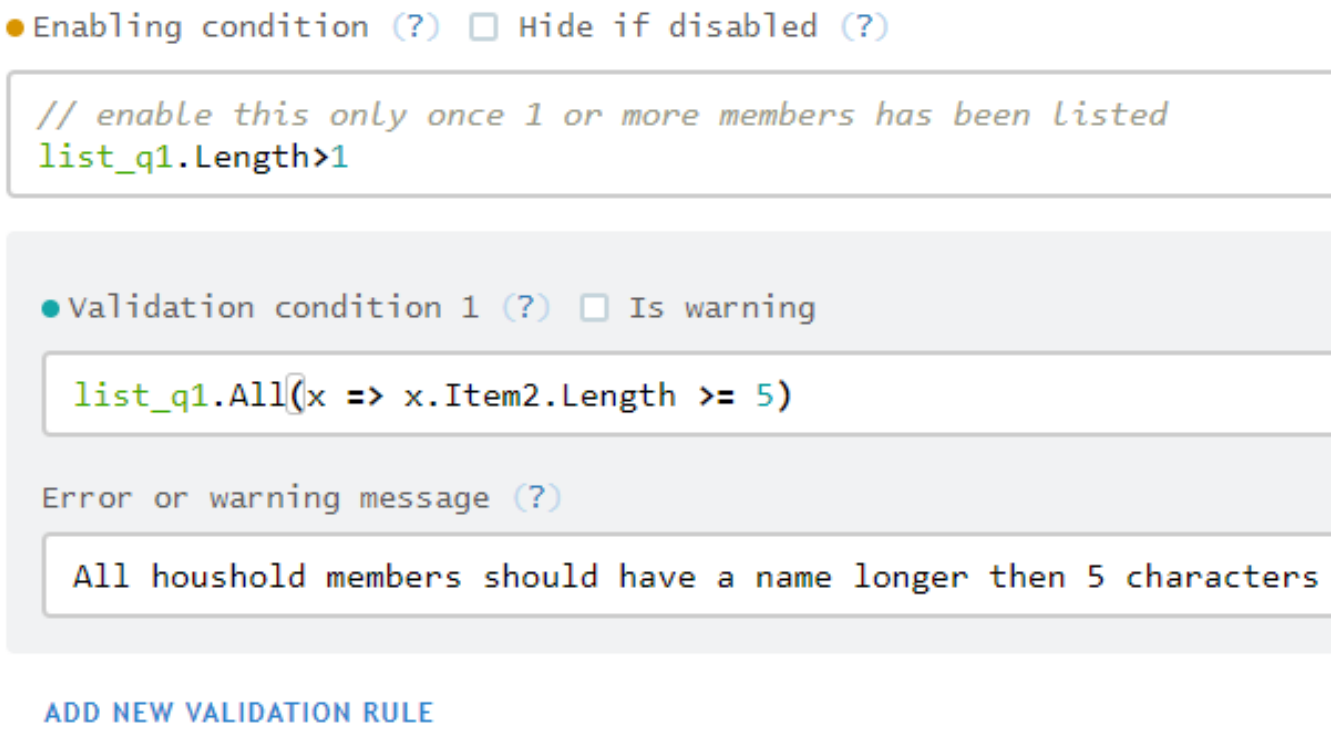
```
// the second coordinates must be at least 10 meters away from the first coordinates
gps_q1.GpsDistance(gps_q2)<10
```

Validation condition 1 (?) Is warning

```
// only accept GPS measures with accuracy of plus or minus 5 meter
self.Accuracy<=5
```

Text list

- Answer on text list questions is an array of complex type with properties
  - Item1 – is a number
  - Item2 – is a value of text list item
- For example, household_members = [
- { 1 , “Mary”},
- { 2, “Paul”}
-]



- Useful properties are Q.Length for number of items
- For particular item – Q.Item2

Lambda and LINQ

- Useful functions
  - `Q.Count(x => x)`
  - `Q.Where(x => x)`
  - `Q.Select(x => x)`
  - …
- For example
  - `Members.Count(person => person.sex == 1)`
  - `Members.Where(person => person.age > 15).Min()`
  - `Foods.Select(item => item.weight * item.calories).Sum()`

Exercise

1. In dwelling section ask the household owner - do they own a car, a garage, a microwave, a fridge or a washer
2. Ask the owner of the household to answer where they park a car for the night if there is no garage in the answer
3. Ask where household members wash their clothes if there is no washer