The 2017 IEEE International Conference on Cybernetics and Computational Intelligence

Perkedel: Spreadsheet-inspired domain-specific programming language for data entry

Authors:
Noprianto
Benfano Soewito, Ph.D
Dr. Ford Lumban Gaol
Prof. Bahtiar Saleh Abbas, Ph.D
Spits Warnars Harco Leslie Hendric, Ph.D
Agung Trisetyarso, Ph.D

Presented by:
Spits Warnars Harco Leslie Hendric, Ph.D

Data Entry Development

- General-Purpose Programming Language

- Domain-Specific Programming Language for Data Entry

- End-user tools
    - No programming needed
    - Less flexible

General-Purpose Programming Language

- Probably overkill for simple data entry applications

- Too complex for end-user programmers

Domain-Specific Programming Language

- Offers expressive power, focused on, usually restricted to a particular problem domain:
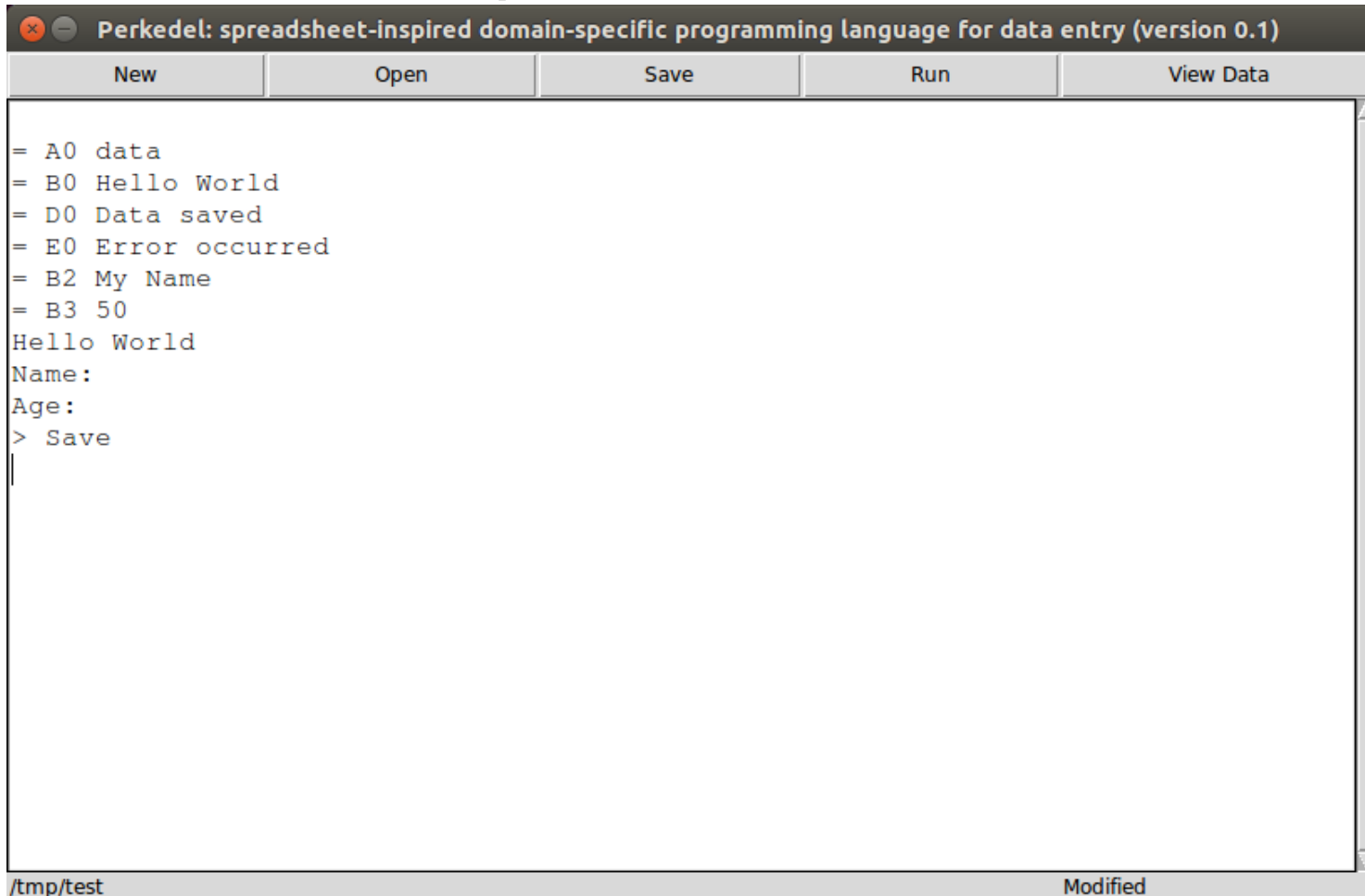
    – Data entry domain

Data Entry

- There are data

- That need to entered by someone (or from some device)

- And need to be saved (or send) to somewhere

- If human is required: user interface should be provided
  - Along with comfortable keyboard shortcuts and understandable messages

- What probably different: data model

Perkedel Programming Language

- Perkedel:
  - Combination of Perk (we hope to add a little perk to developer's life) and del (abbreviation of Data Entry Language)
  - A popular food

- Offers User Interface Definitions that resemble the data entry form in its textual representation

- Using spreadsheet-inspired codes

- Download: http://noprianto.com
  - Implemented in Python

Perkedel Programming Language

- Comes with a simple editor

```
Perkedel: spreadsheet-inspired domain-specific programming language for data entry (version 0.1)

  New          Open          Save          Run          View Data

= A0 data
= B0 Hello World
= D0 Data saved
= E0 Error occurred
= B2 My Name
= B3 50
Hello World
Name:
Age:
> Save


/tmp/test                                          Modified
```

Perkedel Programming Language

- Also comes with a simple database viewer
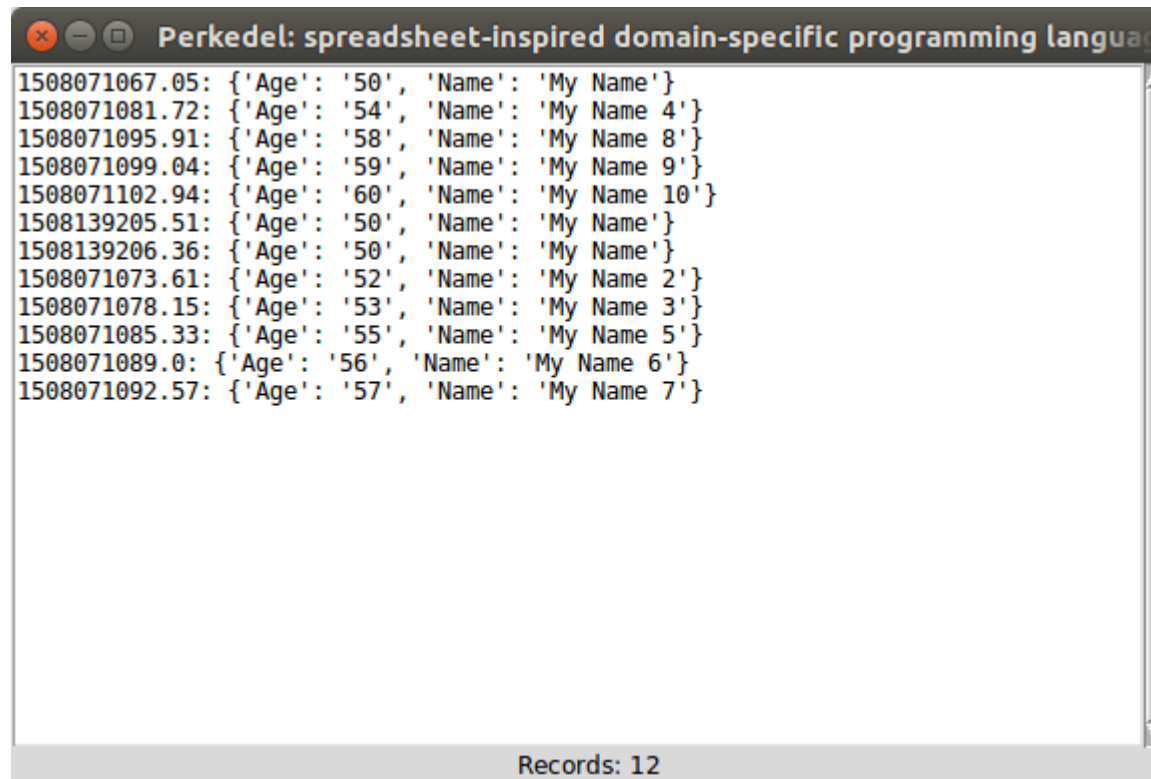
User Interface Definitions (1)

Name:

Age:

That's all :)

User Interface Definitions (2)

This is the Title

Name:

Age:

> Save

With a title and a save button

User Interface Definitions (3)

This is the Title

This is another label
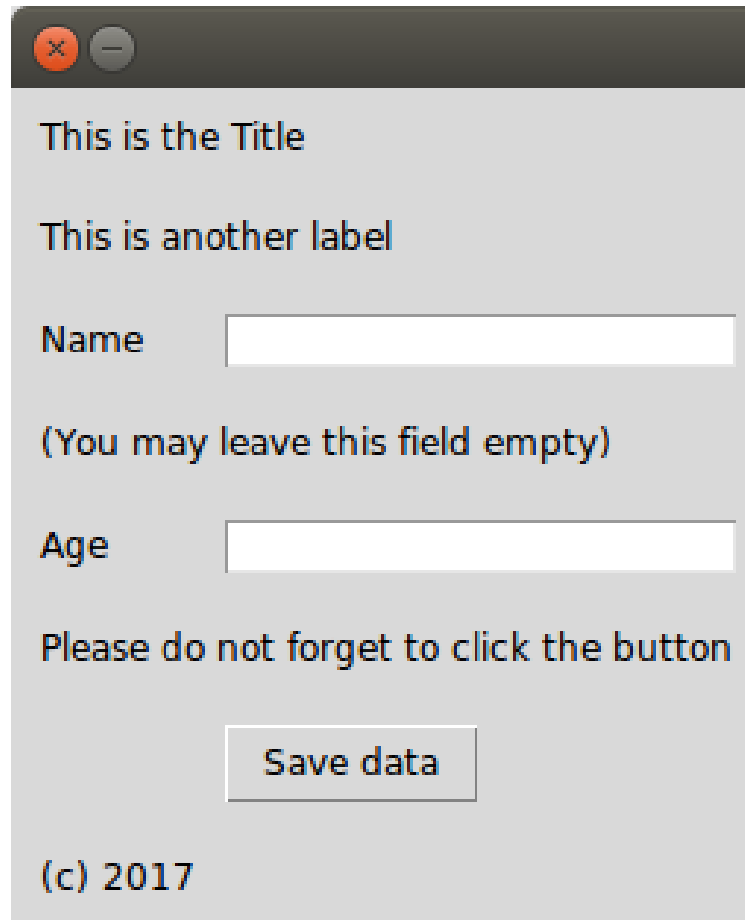
Name:

(You may leave this field empty)

Age:

Please do not forget to click the button

> Save data

(c) 2017

With some titles, labels, and a button

User Interface Definitions (4)

This is the Title

This is another label

Name [   ]

(You may leave this field empty)

Age [   ]

Please do not forget to click the button

[Save data]

(c) 2017

Spreadsheet-inspired codes (1)

- For example, set default value

This is the Title
This is another label
Name:
(You may leave this field empty)
= B5 50
Age:
Please do not forget to click the button
> Save data
(c) 2017

Spreadsheet-inspired codes (2)

- Data entry form: two columns

- Columns:
  - A: Labels
  - B: Input

- Valid identifier: Alphanumeric starting with at least one a-z <u>and</u> at least one 0-9 (case-insensitive)

Spreadsheet-inspired codes (3)

- Assignments:
  - Use equal sign: =
  - Prefix notation
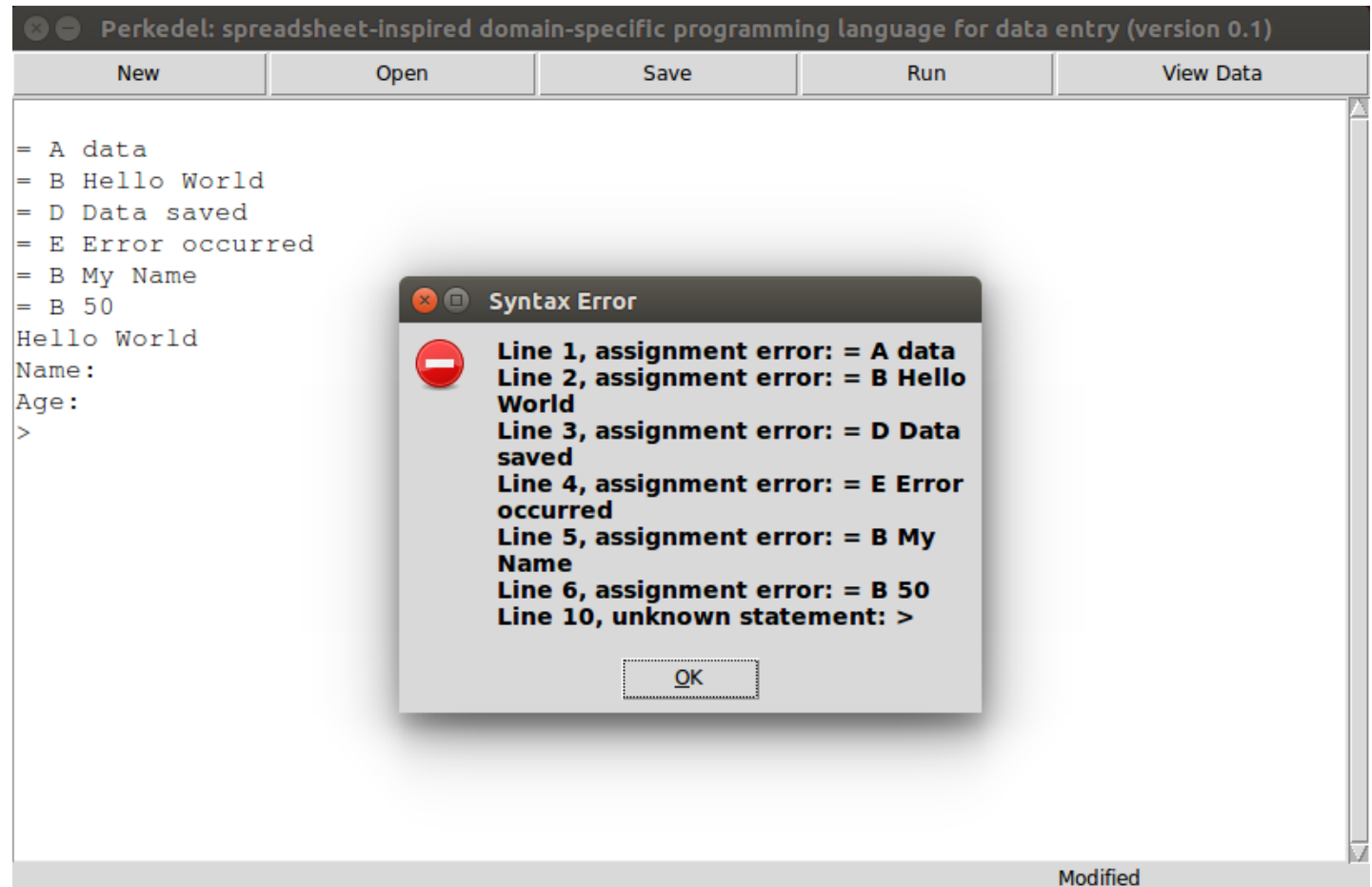
    = B5 50

    - Assign 50 to Input number 5 (B5)

Command

- \> followed by a label

- \> Label

- Example:

  - \> Save

- Currently: only save functionality is provided, more to be implemented

Errors

- Assignment errors, command error, unknown statement

Generated Codes

- Python codes, with Tkinter user interface toolkit

- Spreadsheet-inspired identifiers are used as (variable) names

- Can be run independently of Perkedel Editor/Runtime

Experiments and Result

Respondents	Drag and Drop GUI Builder (minutes)	JSON-based Data Entry Form (minutes)	Perkedel (minutes)
1	43	17	10
2	39	**15**	9
3	50	20	13
4	**32**	18	7
5	33	**15**	7
6	35	16	8
7	40	17	**5 → Fastest Time**
8	41	18	6

All respondents are end-user programmers, familiar with data entry, and – at some levels – also familiar with drag-and-drop GUI builder

Conclusion

- Fast data entry application development
- Source codes that look nothing like ordinary source code
- Further developments:
  - Entering and validating number/boolean
  - User selectable value
  - Multi-line text input
  - More functionalities
  - Required field