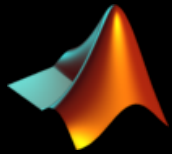


Master program Mind, Brain and Behavior



Programming

Structures

Winter Semester 2024/2025

Dimitris Voudouris, PhD

FB06 – Psychology, F2, Room 342

dimitris.voudouris@psy.jlug.de

www.uni-giessen.de/fbz/fb06/psychologie/abt/allgemeine-psychologie/wh/Team/dvoudouris

Structures

Variables that contain fields of various data types

```
subject.name =      'Dimitris';  
subject.age =       59;  
subject.test =      [55, 60, 61, 74; 82, 72, 83, 71];  
subject.group =     2;  
subject.comments =  {'none'};
```

% The structure “subject” contains 5 fields,
% whose names are mentioned after the dot.
% Each field can be of different data type: string, scalar, matrix, cell, ...

Structure array

Add a second structure

```
subject(2).name =      'Jon';  
subject(2).age =       41;  
subject(2).tests =     [12, 11, 15, 14; 12, 10, 14, 15];  
subject(2).group =     1;  
subject(2).comments =  {'none'};
```

```
% The structure "subject" contains a second element with  
% fields about another student, named "Jon"  
%  
% This is now an array of two structures  
% We call this a structure array
```

Structure array

Add a third structure in your array

```
subject(3).name =      'Marina';  
subject(3).age =       35;  
subject(3).tests =     [84, 94, 85, 84; 83, 72, 95, 75];  
subject(3).group =     1;
```

% The structure “subject” contains a third element with
% fields about another subject, named “Marina”
%

% This structure does **not** contain a field about “comments”, so this field
% will exist in your third structure but will remain empty

Obtain information

Use *fieldnames* to get the names of your fields

```
myfieldnames = fieldnames(subject);
```

% This will return a *cell* array with the names of your fields

Use *orderfields* to order the fields

```
subject = orderfields(subject, [2, 1, 3, 5, 4]);
```

% The numbers in the square brackets show the order that the 'original'
% fields should have in your new arrangement.

Use *rmfields* to remove a field

```
subject = rmfields(subject, 'group');
```

% This will *update* your structure after removing the field 'group'

Hit ***help fieldnames*** _ ***help orderfields*** _ ***help rmfields*** for more details!

Access elements

Use an index after the structure's name to access a structure from the structure array

```
subject(2)
```

% will show the second structure of your array (the one for "Jon")

```
subject(3).tests(end, 1)
```

% will show the value of the (last row, column 1), of the field 'tests' that
% belongs to the third array of your structure (the one for "Marina")

Conversion of structures

Convert a structure to a table

```
tabled = struct2table(subject)
```

% will put the contents of the structure into a table

% each field will become a column with the respective header

```
structured = table2struct(tabled)
```

% will convert the table to a structure

sprintf

Irrespective of whether you work with structures or other variables, you can format text in a variable.

This can be very useful when you want to dynamically change a text variable according to the value of a variable

```
sprintf('this is a demo text')
```

% will create a variable (ans) with the string named within the quotes

```
var1 = sprintf('this is another demo text')
```

% will create the variable 'var1' with the string named within the quotes

sprintf

Change elements of your sprint output

```
trialNr = 1;
```

```
datafile = sprintf('trial_%d', trialNr);
```

% will create variable 'datafile' that will have the text:

% *trial_1* , because at the placeholder **%d** we put the value of *trialNr*

```
data = readmatrix(datafile);
```

% will read the data in the file specified by the variable 'datafile'

Enjoy the structures!