String Manipulations with Arrays

back to Fan's Intro Math for Econon, Matlab Examples, or Dynamic Asset Repositories

String Array

Three title lines, with double quotes:

```
ar_st_titles = ["Title1","Title2","Title3"]';
disp(ar_st_titles);
```

```
    "Title1"
    "Title2"
    "Title3"
```

Three words, joined together, now single quotes, this creates one string, rather than a string array:

```
st_titles = ['Title1','Title2','Title3'];
disp(st_titles);
```

```
Title1Title2Title3
```

String Cell Array

```
ar_st_titles = {'Title1','Title2','Title3'};
disp(ar_st_titles);
```

```
    'Title1'    'Title2'    'Title3'
```

Duplicate String

```
it_duplicate_n = 10;
disp(repmat({'String'}, [1, it_duplicate_n]));
```

```
    'String'    'String'    'String'    'String'    'String'    'String'    'String'    'String'    'String'    'Str
```

String Join to form Single Element

using char() is safe

```
st_var_name = "abc"
```

```
st_var_name = 
"abc"
```

```
st_var_name = [st_var_name ' percentile values']
```

```
st_var_name = 1×2 string array
"abc"        " percentile values"
```

```
strjoin(st_var_name)
```

```
ans = 
"abc  percentile values"
```

```
st_var_name = "abc"
```

```
st_var_name =
"abc"
```

```
st_var_name = [char(st_var_name) ' percentile values']
```

```
st_var_name =
'abc percentile values'
```

```
st_var_name = 'abc'
```

```
st_var_name =
'abc'
```

```
st_var_name = [char(st_var_name) ' percentile values']
```

```
st_var_name =
'abc percentile values'
```

String Join dash (Paste)

This is similar to R's paste function:

```
st_var_name = "abc";
```

```
st_var_name =
"abc"
```

```
st_var_name = [st_var_name, 'efg', 'mqo'];
```

```
st_var_name = 1×3 string array
"abc"        "efg"        "mqo"
```

```
disp(strjoin(st_var_name, "_"));
```

```
ans =
"abc_efg_mqo"
```

```
disp(strjoin(st_var_name, ","));
```

Numeric Array to String without Space

String replace

```
ar_it_test_grp = [3, 8, 9];
strrep(num2str(ar_it_test_grp), '  ', '_')
```

```
ans =
'3_8_9'
```

Substring replace in Cell Array

```
ar_st_cells = {'shock=0.35','shock=0.40','shock=0.46'};
ar_st_updated_cells = strrep(ar_st_cells, 'shock', '$\epsilon$');
```

```
disp(ar_st_updated_cells);
```

```
    '$\epsilon$=0.35'    '$\epsilon$=0.40'    '$\epsilon$=0.46'
```

Find position of String in String Cell

```
ls_st_param_key = {'fl_crra', 'fl_beta', ...
                   'fl_w', 'fl_r_save', ...
                   'fl_a_max', 'it_z_n', 'it_a_n'};
st_param_key = 'fl_a_max';
find(strcmp(ls_st_param_key, st_param_key))
```

```
ans = 5
```

Find the positions of String Cells in Full String Cells

```
ls_st_param_key = {'fl_crra', 'fl_beta', ...
                   'fl_w', 'fl_r_save', ...
                   'fl_a_max', 'it_z_n', 'it_a_n'};

cl_st_param_keys = {'fl_w', 'fl_beta', 'it_z_n'};

cell2mat(cellfun(@(m) find(strcmp(ls_st_param_key, m)), ...
                 cl_st_param_keys, 'UniformOutput', false))
```

```
ans = 1×3
     3     2     6
```

```

find(strcmp(ls_st_param_key, st_param_key))
```

```
ans = 5
```

Cell to string Paste and Replace dash

```
cl_st_param_keys = {'fl_crra', 'fl_beta'};
display(strrep(strjoin(cl_st_param_keys, '-'), '_', '\_'));
```

```
fl\_crra-fl\_beta
```