

Testing BEAR6 Excel User Interface in the sandbox

Cloning the BEAR6 repository [NB: This step will be replaced by a proper Matlab App installation process]

- In a folder of your choice, run the `git clone` command to obtain a local copy of the repository files... One of following commands will work for you, depending which authentication process you use (personal access token or SSH):

```
git clone https://PERSONAL_ACCESS_TOKEN@github.com/european-central-bank/BEAR-toolbox-6.git
```

```
git@github.com:european-central-bank/BEAR-toolbox-6.git
```

where you replace `PERSONAL_ACCESS_TOKEN` with your own GitHub personal access token with the appropriate privileges.

- The `git clone` command creates a `BEAR-toolbox-6` subfolder, and this subfolder will be referred to as the root folder from now on.

Preparation for testing in the sandbox [NB: This step will be not needed since users will run BEAR from their own working folders]

- Open Matlab
- Switch to the `BEAR-toolbox-6/tbx/sandbox`
- Make sure the BEAR6 toolbox is on the Matlab path

```
>> bear6.ping
```

Create a local copy of the Excel UX file

- The `BEAR-toolbox-6/tbx/sandbox` folder contains a `BEAR6_UX.xlsx` file.
- For the current testing purposes, some specific meta information (e.g. variable names) is filled in for convenience but will be removed when deployed.
- Make a local copy of this Excel UX file under a different name, still within the sandbox folder, e.g. `BEAR-toolbox-6/tbx/sandbox/BEAR6_UX_test.xlsx`

Fill in meta information

- Open the newly created copy of the Excel UX file.
- Fill in all information for estimating the reduced-form model on sheet `Reduced-form meta information`
- Fill in all information for indentifying the structural model on sheet `Structural meta information`
- Save the Excel UX file, and close it. Closing the file is (unfortunately) critical for Matlab/BEAR6 to be able to finalize the file.

The meta information sheets will now be used to generate the meta-dependent templates on some of the estimation and identification sheets.

Automatically generate meta-dependent templates

- After making sure the Excel UX file is closed, run the following command to

```
bear5.finalizeExcelUX("BEAR6_UX_1.xlsx")
```

where `BEAR6_UX_1.xlsx` stands for the name of your local copy of the Excel UX file created previously.

Fill in the remaining information

- Open the Excel UX file again, and fill in the remaining information on the estimation and identification sheets.

Run the model

- Run the model by running the following command

```
bear6.runFromExcelUX("BEAR6_UX_1.xlsx")
```

where `BEAR6_UX_1.xlsx` stands for the name of your local copy of the Excel UX file.

- The command will run the reduced-form estimation, the structural identification, and all the tasks specified on the "Tasks" sheet in the Excel UX file.
- The results will be saved in the files specified on the "Tasks" sheet in the Excel UX file.