

Code review

Here some suggestions for the **signup.py** file (this list does not pretend to be exhaustive):

1. Let's follow the SoC (**Separation of Concerns**) principle, therefore, move each class to its own file (*pages.py*, *processor.py*, *main.py*) to enhance modularity, testability and maintainability.
2. Let's follow the SRP (**Single Responsibility Principle**) principle, therefore, each class should have only one responsibility. For example:
 - *Processor#process_form* should only focus on processing the form data and not on printing/interacting via messages with the users - so remove the usage of *pages* (line 39) in the *process_form* function.
 - Also, be carefully with circular dependencies, in this case it's not critical but still somehow present: *Pages#signupform_page* creates an instance of *Processor* and calls *Processor#process_form*, in turn *process_form* creates an instance of *Pages* (*Pages* -> *Processor* -> *Pages*).
3. We should search for high cohesive and low coupled code, for this, we should avoid creating instances of other classes and instead follow the IoC (**Inversion of Control**) principle. For this we can pass the dependencies (**Dependency Injection**) via the classes constructors. Particularly, an instance of *Processor* should be provided to *Pages*, so it can be called to handle the user inputs.
4. Move the *users* table name to a constant, configuration variable or environment variable if no other way. We should avoid hardcoded values.
5. Validation on inputs must be implemented:

- Validate that all the user inputs have *length* > 0.
 - Check is a valid email address, a first approach can be implemented using a RegEx.
 - Age should be an integer.
 - User type must be in the set (*user, staff, admin*).
6. After implementing the validations, the *process_form* function can be refactored to avoid duplicated code. At this point *formvars['usertype']* should only hold a valid user type - (user inputs were validated), so no need for those *if/else* statements. Now, it can simply call the *insert* function and add the arguments. More processing on the data can be made in this function if required.
7. Add error handling for DB *insert* function call in *Processor#process_form*.
8. Remove returns from *Pages#thankyou_page* and from *Processor#process_form* as they are useless at this point (**YAGNI, KISS**).
9. Follow the snake_case naming convention. For example, *signupform_page* should be *signup_form_page*.

Here some suggestions for the **test.py** file (this list does not pretend to be exhaustive):

-- TODO --