**PHP Form Validation**

# Goal(s)

* PHP form Validation

# What You Should Already Know

Before starting this assignment, you should have the basic knowledge of the followings:

* XAMPP
* Apache
* PHP basics

# Academic Honesty

All the work that you do toward the fulfilment of this assignment must be your own unless collaboration is explicitly allowed (e.g., by some problem set or the final project). Viewing or copying another individual's work (even if left by a printer, stored in an executable directory, or accidentally shared in the course's virtual classroom) or lifting material from a book, magazine, website or from other sources, even partially and presenting it as your own, constitutes academic dishonesty, as does showing or giving your work, even in part, to another student.

Similarly, dual submission academic dishonesty: you may not submit the same or similar work to this course that you have submitted or will submit to another. You should not provide or make available your assignments to individuals who have taken this course or may take this course in the future.

You are welcome to discuss the course's material with others in order to better understand it. You may even discuss problem sets with classmates, but you should not share code. You may also turn to the Web for instruction beyond the course's lectures, for references, and for solutions to technical difficulties, but not for outright solutions to problems on projects. However, failure to cite (as with comments) the origin of any code or technique that you do discover outside of the course's lectures and (even while respecting these constraints) and then integrate into your own work may be considered academic dishonesty.

All forms of academic dishonesty will be dealt very harshly.

# Grades

Your work will be evaluated along the following axes.

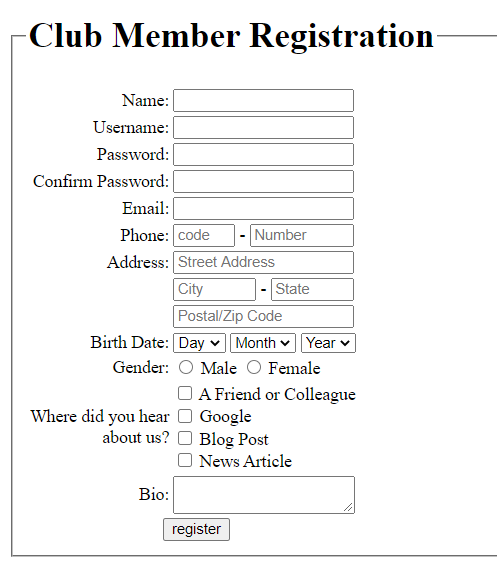
**Correctness**: Consistency with the given specifications and free of bugs.

**Design:** Well written code (i.e., clearly, efficiently, elegantly, and/or logically)

**Style:** Readable and modifiable code (commented, aptly named variables etc.)

# Problem 1

Design and validate the following form.



# Functional Requirements

1. All empty values should be checked.
2. Check for html characters.
3. Username must contain at least 6 characters and space is not allowed.
4. Email should be validated with php. Conditions must contain @ and at least one .(dot) after @
5. Date fields’ options should be generated by loop.
6. Phone number fields will only accept numeric value.
7. Password must contain at least 8 character, 1 special character(only check # or ?),1 number and combination of uppercase and lowercase alphabet.
8. After occurrence of error proper error message should be shown.
9. **Usage of regular expression or input type date, email, number, required is strictly prohibited for validation. You must validate with your own logic using server side language php.**

# Technical Requirements

* Your PHP should be well-formed.
* You should not use an intelligent text editor to generate HTML.
* Use these php functions
  + Htmlspecialchars($string)🡪filtering html tags
  + strlen($string)🡪finding the length of a string
  + is\_numeric($string)🡪checking a string is numeric or not
  + strpos($string,$char)🡪finding character/word position In a string
  + isset($var)🡪checking a variable is set or not
  + empty($var)🡪checking a string is empty or not
  + ctype\_upper($string)🡪checking of uppercase existence
  + ctype\_lower($string)🡪checking of lowercase existence

# Submission

github

**Submission Deadline: This Lab**