

SOEN287 – PHP In-Class Problem Solving Session

Thursday November 6/2014

Questions 2 and 3 are based on games found at

<http://www.ibm.com/developerworks/opensource/library/os-php-gamescripts1/index.html>

Question 1: Simple form with PHP

Here are a few screen captures to illustrate how you code should work

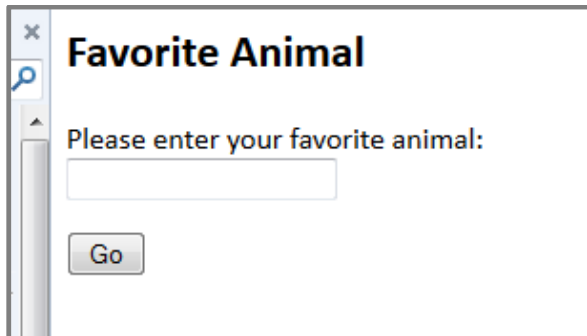


Fig. 1: Starting screen

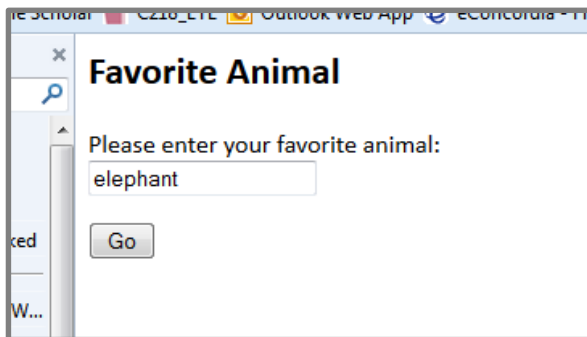


Fig. 2: User enters animal and clicks on go

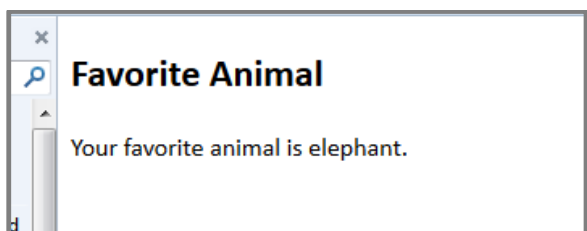
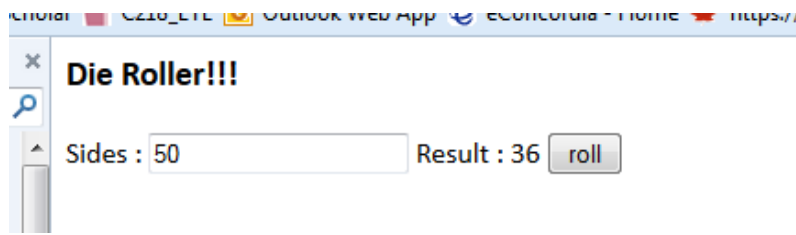


Fig. 3: Response screen, where user's input is echoed

Write the necessary code to produce the above code: one set of instructions for the form, one for the response.

Question 2: Die Roller



Write the PHP code to generate the above form, in which the user enters the sides of his/her die. Each time the user clicks on the roll button a random number between 1 and the number of sides is generated.

Note: `rand(n1, n2)` will generate a number between n1 and n2 inclusive.

What to do:

Write a PHP file which contains:

- a function `roll()` which given the number of sides will generate a number between 1 and the number of sides.
- the necessary code to produce the form in the illustration above, accept the user input and when the `roll` button is 'pressed' will display the result of the roll in the `Result` field.

Question 3: Mad Lib

"Mad Libs is a word game where the player takes a short story and replace key types of words with different words of the same type to create a new, sillier version of the same story."

Given the starting code

```
<?php

$text = "I was _VERB_ing in the _PLACE_ when I found a _NOUN_. I
_VERB_ed in, and _VERB_ed too much _NOUN_. I had to go to the
_PLACE_.";

echo "Original String: ", "<br />",$text, "<br />", "<br />";

$verbs = explode("\n", file_get_contents("words.verbs.txt"));
$places = explode("\n", file_get_contents("words.places.txt"));
$nouns = explode("\n", file_get_contents("words.nouns.txt"));
```

Where `explode` takes the content of the file in between the `()` and breaks it into an array. The `"\n"` specifies where to break the string.

Write a while loop, that will find every occurrence of `_VERB_`, `_PLACE_` and `_NOUN_` in the original string and replace it by a word from the appropriate array. To pick a word, you can either generate a random number from 0 to the size of the array, or use the `shuffle()` method for arrays which shuffles the content of an array, or ...

Here is a sample run:

Original String:

I was VERBing in the PLACE when I found a NOUN. I VERBed in, and VERBed too much NOUN. I had to go to the PLACE.

I was bumping in the PLACE when I found a NOUN. I VERBed in, and VERBed too much NOUN. I had to go to the PLACE.

I was bumping in the bank when I found a NOUN. I VERBed in, and VERBed too much NOUN. I had to go to the PLACE.

I was bumping in the bank when I found a airfield. I VERBed in, and VERBed too much NOUN. I had to go to the PLACE.

I was bumping in the bank when I found a airfield. I rolled in, and VERBed too much NOUN. I had to go to the PLACE.

I was bumping in the bank when I found a airfield. I rolled in, and hideed too much NOUN. I had to go to the PLACE.

I was bumping in the bank when I found a airfield. I rolled in, and hideed too much dares. I had to go to the PLACE.

I was bumping in the bank when I found a airfield. I rolled in, and hideed too much dares. I had to go to the bank.

Final String: I was bumping in the bank when I found a airfield. I rolled in, and hideed too much dares. I had to go to the bank.