**Objective**

**LEARN PHP**

**FizzBuzz**

FizzBuzz is one of the most commonly used interview questions for applicants to programming positions. Each interviewer has their own variant, but the prompt is usually something like:

Write code that prints the numbers from 1 to 100 (inclusive), except for these cases:  
- If a number is a multiple of 3, write "Fizz".  
- If a number is a multiple of 5, write "Buzz".  
- If a number is a multiple of both 3 and 5, write "FizzBuzz".

**Tasks**

**0/13 Complete**

Mark the tasks as complete by checking them off

**Use a while loop**

**1.**

There are many ways to solve this problem using PHP. Let’s start by using a while loop to count from 1 to 100.

For now, create a $counter variable to keep track of the number and print it to the screen on each iteration.

Make sure to put a newline before the end of each iteration.

You’ll also need to increment your $counter variable within the code block.

Hint

$counter = 1;  
while ($counter <= 100) {  
  echo $counter;  
  echo "\n";  
  $counter++;  
}

**2.**

Within the loop, establish some if, elseif, else statements to determine what to print at each iteration.

Despite the order that the interviewer introduces the various conditions, you actually need to check for numbers divisible by 15 first. If you check first for divisible by 3, you will print "Fizz" and skip past the elseif check for 15.

Remember to use the modulo operator (%) to check if a number is divisible by another. The modulus is 0 when two numbers are divisible.

Hint

while ($counter <= 100) {  
  if ($counter % 15 === 0) {  
   
  } elseif ($counter % 3 === 0) {  
   
  } elseif ($counter % 5 === 0) {  
   
  } else {  
   
  }  
  echo $counter;  
  echo "\n";  
  $counter++;  
}

**3.**

Move the echo $counter; line inside the correct conditional statement. Remember, we only print the number if none of the other conditions are met.

Hint

} else {  
  echo $counter;  
}

**4.**

Place statements within the other three conditionals for "FizzBuzz", "Fizz", and "Buzz".

Your code should now print the complete output for FizzBuzz!

Hint

# while loop  
$counter = 1;  
while ($counter <= 100) {  
  if ($counter % 15 === 0) {  
    echo "FizzBuzz";  
  } elseif ($counter % 3 === 0) {  
    echo "Fizz";  
  } elseif ($counter % 5 === 0) {  
    echo "Buzz";  
  } else {  
    echo $counter;  
  }  
  echo "\n";  
  $counter++;  
}

**Use a for and foreach loop**

**5.**

Let’s implement the solution again, but this time we will make use of a for and a foreach loop.

Instead of printing the statements at each step, we will queue them up into an array and print them all out at the end.

Begin by creating an empty array $output to store the statements in.

Hint

$output = [];

**6.**

Add a for loop that counts from 1 to 100. Use $i as your loop counter variable.

Hint

There are a few ways to do this. We went with:

for ($i = 1; $i <= 100; $i++) {  
   
}

**7.**

Add conditions within your for loop to determine what to add to the output at each iteration. These should be the same as before, using the modulus (%) operator.

Hint

if ($i % 15 === 0) {  
   
} elseif ($i % 3 === 0) {  
   
} elseif ($i % 5 === 0) {  
   
} else {  
   
}

**8.**

Instead of adding echo statements inside each conditional, push the appropriate statement onto the $output array.

You can use the built in function array\_push for this.

Hint

if ($i % 15 === 0) {  
  array\_push($output, "FizzBuzz");  
} elseif ($i % 3 === 0) {  
  array\_push($output, "Fizz");  
} elseif ($i % 5 === 0) {  
  array\_push($output, "Buzz");  
} else {  
  array\_push($output, $i);  
}

**9.**

You now have an $output array with the correct print statements, but it’s not formatted very nice. Let’s use a foreach loop to iterate through it and print out the statements.

Create a foreach loop that iterates through $output. Use $value for the variable at each position in the array.

Hint

foreach ($output as $value) {  
  echo $value . "\n";  
}

**10.**

Print the $value at each loop iteration followed by a newline.

This should match the output from the while loop implementation.

Hint

foreach ($output as $value) {  
  echo $value . "\n";  
}

**Break and continue**

**11.**

Having the output stored in an array is nice since we can re-print the same output but add new conditions.

Create a copy of the previous foreach loop that iterates over $output.

Hint

foreach ($output as $value) {  
  echo $value . "\n";  
}

**12.**

The interviewer has asked that now you avoid printing anything when a number is divisible by 3 ("Fizz").

Add an if and a continue statement to avoid printing anything when the $value is "Fizz".

When you scroll to the bottom, your last output should only have numbers, "Buzz", and "FizzBuzz".

Hint

foreach ($output as $value) {  
  if ($value === "Fizz") {  
    continue;  
  }  
  echo $value . "\n";  
}

**13.**

As a final step, the interviewer has asked that now you stop printing values after the first "FizzBuzz".

Add an elseif statement to your conditional. Within it, it should print the $value and exit from the loop.

Hint

foreach ($output as $value) {  
  if ($value === "Fizz") {  
    continue;  
  } elseif ($value === "FizzBuzz") {  
    echo $value;  
    break;  
  }  
  echo $value . "\n";  
}

whole:

<?php

$counter = 1;

while ($counter <= 100) {

if ($counter % 15 === 0) {

} elseif ($counter % 3 === 0) {

} elseif ($counter % 5 === 0) {

} else {

}

echo $counter;

echo"\n";

$counter++;

}

output:

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

61

62

63

64

65

66

67

68

69

70

71

72

73

74

75

76

77

78

79

80

81

82

83

84

85

86

87

88

89

90

91

92

93

94

95

96

97

98

99

100

Complete