

도전! PHP

01. data라는 변수가 있습니다. 이 변수에 저장된 값이 90점 이상은 '수', 80점 이상은 '우', 70점 이상은 '미', 60점 이상은 '양', 60점 이하는 '가'라는 프로그램을 제작해 보시오.

정답

```
<!DOCTYPE html>
<html lang="ko">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>제어문</title>
</head>
<body>
    <?php
        $data = 70;

        if($data >= 90) {
            print "당신의 점수는 90점 이상 수입니다.";
        } else if($data >= 80) {
            print "당신의 점수는 80점 이상 우입니다.";
        } else if($data >= 70) {
            print "당신의 점수는 70점 이상 미입니다.";
        } else if($data >= 60) {
            print "당신의 점수는 60점 이상 양입니다.";
        } else {
            print "당신의 점수는 60점 이하 가입니다.";
        }
    ?>
</body>
```

</html>

결과 화면

당신의 점수는 70점 이상 미입니다.

해설

if 조건문을 사용해 \$data 변수에 저장된 값이 90점 이상이면 수, 80점 이상이면 우, 70점 이상이면 미, 60점 이상이면 양, 60점 이하이면 가라는 결과를 보여주는 코드입니다. 이처럼 조건 분기가 많은 경우에는 if문 보다는 switch문을 사용하는 게 더 유용합니다.

02. 다음은 for 반복문을 사용해 7단을 표시하는 샘플입니다. 결과를 유추해 보세요.

정답

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width,
initial-scale=1.0">
  <title>제어문</title>
</head>
<body>
```

```

    <?php
        for($i = 1; $i <= 9; $i++) {
            $data = $i * 7;
            print "7 * $i = $data";
            print "<br>";
        }
    ?>
</body>
</html>

```

결과 화면

```

7 * 1 = 7
7 * 2 = 14
7 * 3 = 21
7 * 4 = 28
7 * 5 = 35
7 * 6 = 42
7 * 7 = 49
7 * 8 = 56
7 * 9 = 63

```

해설

for 반복문을 사용해 i 값이 1부터 9까지 순서대로 반복되도록 설정(\$i<=9) 했습니다. 반복할 때마다 i 값을 7로 곱한 결과를 출력했기 때문에 구구단의 7단이 출력되게 됩니다. 8단을 출력하고자 한다면 7을 8로 변경하면 되고 5단을 출력하고자 하면 7을 5로 변경하면 됩니다.

03. 중첩된 for문의 프로그램입니다. 구구단을 표시하기 위해 () 안에 들어갈 숫자를 입력해 보세요.

```

<!DOCTYPE html>
<html lang="ko">
<head>
    <meta charset="UTF-8">

```

```

        <meta name="viewport" content="width=device-width,
initial-scale=1.0">
        <title>제어문</title>
</head>
<body>
    <?php
        for($i = ( ); $i <= 9; $i++) {
            for($j = 1; $j <= ( ); $j++) {
                $data = $i * $j;
                print "$i * $j = ( )";
                print "<br>";
            }
        }
    ?>
</body>
</html>

```

정답

2, 9, \$data

결과 화면

2 * 1 = 2	5 * 7 = 35	
2 * 2 = 4	5 * 8 = 40	
2 * 3 = 6	5 * 9 = 45	
2 * 4 = 8	6 * 1 = 6	
2 * 5 = 10	6 * 2 = 12	
2 * 6 = 12	6 * 3 = 18	
2 * 7 = 14	6 * 4 = 24	
2 * 8 = 16	6 * 5 = 30	
2 * 9 = 18	6 * 6 = 36	
3 * 1 = 3	6 * 7 = 42	
3 * 2 = 6	6 * 8 = 48	
3 * 3 = 9	6 * 9 = 54	
3 * 4 = 12	7 * 1 = 7	
3 * 5 = 15	7 * 2 = 14	
3 * 6 = 18	7 * 3 = 21	
3 * 7 = 21	7 * 4 = 28	
3 * 8 = 24	7 * 5 = 35	
3 * 9 = 27	7 * 6 = 42	
4 * 1 = 4	7 * 7 = 49	
4 * 2 = 8	7 * 8 = 56	
4 * 3 = 12	7 * 9 = 63	
4 * 4 = 16	8 * 1 = 8	
4 * 5 = 20	8 * 2 = 16	
4 * 6 = 24	8 * 3 = 24	
4 * 7 = 28	8 * 4 = 32	
4 * 8 = 32	8 * 5 = 40	
4 * 9 = 36	8 * 6 = 48	
5 * 1 = 5	8 * 7 = 56	
5 * 2 = 10	8 * 8 = 64	
5 * 3 = 15	8 * 9 = 72	
5 * 4 = 20	9 * 1 = 9	
5 * 5 = 25	9 * 2 = 18	
5 * 6 = 30	9 * 3 = 27	

9 * 4 = 36
9 * 5 = 45
9 * 6 = 54
9 * 7 = 63
9 * 8 = 72
9 * 9 = 81

해설

for 반복문 안에 for 반복문이 있습니다. 너무 복잡하게 생각하지 마세요. 밖의 for 문이 한 번 반복할 때 안쪽 반복문은 설정된 만큼 반복을 합니다. 즉 안쪽 for 문을 보면 j 변수는 1부터 9까지 반복을 하기 때문에 밖의 for문의 변수 값 i가 1이면 j값은 1부터 9까지 반복하고 2일 때 역시 1부터 9까지 반복합니다. 즉 이 설정에 의해 구구단을 표시하는 코드를 만들 수 있습니다.

04. while 문과 do~while 문에 대한 차이점을 설명해 보시오.

정답

```
<!DOCTYPE html>
<html lang="ko">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>제어문</title>
</head>
<body>
    <?php
        $data = 101;
        while($data <= 100) {
            $hap += $data++;
        }
        print "hap = $hap";

        $data = 101;
        do {
            $hap += $data++;
        }
        while($data <= 100);
```

```
        print "hap = $hap";
    ?>
</body>
</html>
```

다음 코드를 보면 쉽게 이해가 될 것입니다. while문의 조건식을 보면 data에 저장된 값이 100과 같거나 작다면 반복문 내부의 명령문이 실행됩니다. 하지만 처음부터 조건이 맞지 않기 때문에 아무런 실행도 되지 않습니다. 하지만 do~while문은 조건식이 뒤에 있기 때문에 무조건 반복문 내부의 명령은 단 한 번 실행됩니다.

05. 다음 switch 문을 보고 문제점이 무엇인지 설명하고 문제점을 해결한 다음 결과를 설명하시오.

정답

```
<!DOCTYPE html>
<html lang="ko">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>제어문</title>
</head>
<body>
    <?php
        $data = 10;
        switch($data) {
            case 100:
                print "data 변수에 저장된 값은 $data
입니다.";
            case 200:
                print "data 변수에 저장된 값은 $data
입니다.";
```

```
        case 300:
            print "data 변수에 저장된 값은 $data
입니다.";

        default:
            print "data 변수에 저장된 값은 100, 200,
300이 아닙니다.";
    }

?>
</body>
</html>
```

결과 화면

data 변수에 저장된 값은 100, 200, 300이 아닙니다.

해설

switch문 역시 if문과 같은 조건문입니다. 즉 조건에 맞는 경우에만 실행되도록 명령을 설정할 수 있습니다. switch문의 \$data에 저장된 값이 100, 200, 300인 경우 실행 명령을 달리 할 수 있고 이 세 개가 아닌 경우 다른 명령을 실행하도록 설정할 수 있습니다.