

# PHP – ciklai



**Code Academy**

Programuok savo ateitį!

- » Kai rašome kodą, norime, kad tas pats bloko kodas būtų kartojamas. Užuoť pridęję lygių kodo eilučių, mes galime naudoti kilpas(*angl.* Loops), kad atliktume tokią užduotį.

- » **for** – pereina per kodo bloką nurodytą kartų kiekį. Turi sąlygą bei skaitiklį, kuris užima papildomą kiekį atminties, todėl yra lėtesnis būdas.

```
for ($i=0; $i < 10 ; $i++) {  
    echo "Labas $i<br>";  
}
```

## For ciklas

- » `for (skaitiklio inicijavimas; patikrinimo skaitiklis; skaitiklio prieaugis){`  
    *Kodas, kuris turi būti įvykdytas;*  
}
- » skaitiklio inicijavimas - Inicijuojame ciklo skaitiklio vertę.
- » patikrinimo skaitiklis – kiekvieno ciklo iteracijos įvertinimas. Kol sąlyga teisinga(*true*), tol ciklas tęsiasi, kai sąlyga yra neteisinga(*false*), ciklas yra sustabdomas.
- » Skaitiklio prieaugis – nusako kaip skaitiklis turi būti pakeistas su kiekviena iteracija. Skaitiklio kitimas turi vesti į programos baigtį.

## Foreach ciklas

- » **Foreach** – Pereina per masyvo kiekvieną elementą (geriausiai suderinamas ciklas su masyvais bei masyvų sarašais).

```
//          array          value
foreach ($masyvas as $reikšmė) {
|   echo "<span>$value</span>";
}
```

```
//          array          key          value
foreach ($masyvas as $raktas => $reikšmė) {
|   echo "<div>[$key] => $value</div>";
}
```

## While, do...while

- » **while** – kodas yra vyddomas, kol sąlyga yra teisinga (*true*).
- » **do...while** – Kartoja kodo bloką kartą, bet kartos tą kodą iki kol sąlyga yra teisinga, bet ivykdydys kodą kartą, jeigu sąlyga yra ir neteisinga.

```
$skaitiklis = 1;
while($skaitiklis <= 5) {
    echo "<div>Skaitiklis: $skaitiklis</div>";
    $skaitiklis++;
}

do {
    --$skaitiklis;
    echo "<div>Skaitiklis: $skaitiklis</div>";
} while($skaitiklis != 1);
```

## Paskirstymo operatoriai

`$a += $b`

`$a -= $b`

`$a *= $b`

`$a /= $b`

`$a %= $b`

`$a = $a + $b`

`$a = $a - $b`

`$a = $a * $b`

`$a = $a / $b`

`$a = $a % $b`

Sudėtis

Atimtis

Daugyba

Dalyba

Modulis

`$a++`

`$b--`

`++$a`

`--$b`

Padidina `$a` vienetu po komandos įvykdymo

Sumažina `$b` vienetu po komandos įvykdymo

Padidina `$a` vienetu prieš vykdant komandą

Sumažina `$b` vienetu prieš vykdant komandą

## Užduotys (for)

- » Išvesti į ekraną 10 kartų žodį 'Labas';
- » Išvesti į ekraną skaičius nuo 1 iki 10;
- » Išvesti į ekraną skaičius nuo 10 iki 1;
- » Išvesti į ekraną visus lyginius skaičius nuo 1 iki 100;
- » Išvesti į ekraną visus masyvo elementus (skaičiu masyvas);
- » Išvesti į ekraną visų masyvo elementų sumą;
- » Išvesti į ekraną visų masyvo elementų sandaugą;
- » Išvesti į ekraną kas antrą masyvo elementą;
- » Funkciją, kuri apsuktų masyvo elementus (iš kito galo);



## Užduotys (foreach)

- » Išvesti visus masyvo elementus;
- » Išvesti visų masyvo elementų vidurkį;
- » Padidinti kiekviena masyvo elementą vienetu;
- » Išvesti automobilis masyvą į ekraną; (dvilypis masyvas);
- » Sukurti Fibonacci seką (tai tokia seka, kuomet sekantis skaičius yra sudedamas su prieš tai 2 einančiu.)

## Užduotys (while)

- » Parašykite funkciją, kuri mestų kauliuką iki kol iškris 5 arba 6. Visus savo metimus išvestu į ekrana. (||, while) **Bonus**
- » Metame 2 kauliukus (arba vieną kauliuką 2 kartus).

### taisyklės:

Jeigu kauliukų suma yra mažiau už 5 reiškia - jūs iškarto pralaimėjote.

Jeigu kauliukų suma daugiau už 5 ir:

- išmesta buvo 6 ir 6, reiškia - laimėjote dviratį.
- abiejų kauliukų išmestas skaičius yra vienodas, bet ne 6 ir 6 - reiškia jūs laimėjote bilietą į kiną.
- kitais atvejais – kauliukas metamas dar kartą.

- » Žaidimas turi vykti iki kol žaidėjas laimės / pralaimės(while)
- » Visa žaidimo eiga turėtų būti išvesta į ekraną.