

Фрагменты Perl 6 в Perl 5.10

Фрагменты Perl 6 в Perl 5.10 и различия

```
use feature qw(  
    say  
    switch  
    state  
);
```

```
use feature qw(  
    say  
    switch  
    state  
);
```

```
use feature "5.10";
```

```
use feature qw(  
    say  
    switch  
    state  
);
```

```
use feature "5.10";  
use 5.10.0;
```

```
use feature qw(  
    say  
    switch  
    state  
);
```

```
use feature "5.10";  
use v5.10.0;
```

say

```
my $x = 'UPW2008';
```

```
say $x;
```

5.10

```
my $x = 'UPW2008';  
say $x;
```

UPW2008

```
my $x = 'UPW2008';  
say $x;
```

UPW2008

```
my $x = 'UPW2008';  
say ($x);
```

```
my $x = 'UPW2008';  
say ($x);
```

UPW2008

```
my $x = 'UPW2008';  
say ($x);
```

UPW2008

```
my $x = 'UPW2008';  
say($x);
```

UPW2008

```
my $x = 'UPW2008';  
$x.say;
```

UPW2008

```
my $x = 'UPW2008';  
$x.say;
```

```
my $x = 'UPW2008';  
$x.say();
```

UPW2008

```
my $x = 'UPW2008';  
$x.say();
```

Undefined subroutine

&main::say called at -e

line 1

```
for (1..3) {  
    say;  
}
```

5.10

```
for (1..3) {
```

```
  say;
```

```
}
```

1

2

3

```
for (1..3) {  
  say;  
}
```

```
\n
```

```
\n
```

```
\n
```

```
for (1..3) {  
  say $_;  
}
```

1

2

3

```
for (1..3) {  
    $_.say;  
}
```

1

2

3

```
for (1..3) {  
  .say;  
}
```

1

2

3

Switch

5.10, 6

```
my $str = "YAPC::Asia";  
given ($str) {  
  when (/Asia/) {  
    say "Asia"  
  }  
}
```

5.10, 6

```
my $str = "YAPC::Asia";  
given ($str) {  
  when (/Asia/) {  
    say "Asia"  
  }  
}
```

```
my $str = "YAPC::Asia";  
given ($str) {  
    when (/Asia/) {  
        say "Asia"  
    }  
}
```

```
my $str = "YAPC::Asia";  
given $str {  
  when /Asia/ {  
    say "Asia"  
  }  
}
```

```
my $str = "YAPC::Asia";  
given $str {  
    say "Asia" when /Asia/  
}
```

statae

```
sub f {  
    state $c;  
    say ++$c;  
}
```

```
sub f {  
    state $c;  
    say ++$c;  
}  
f(); f(); f();
```

5.10

```
sub f {  
    state $c;  
    say ++$c;  
}  
  
f(); f(); f();  
1 2 3
```

```
sub f {  
    state $c;  
    say ++$c;  
}  
  
f(); f(); f();  
1 2 3
```

```
sub f {  
    state $c = 0;  
    say ++$c;  
}  
f(); f(); f();
```

5.10

```
sub f {  
    state $c = 0;  
    say ++$c;  
}
```

```
f(); f(); f();
```

1 2 3

pugs

```
sub f {  
    state $c = 0;  
    say ++$c;  
}
```

```
f(); f(); f();
```

1 1 1



//

defined-or

```
my $c = 0;
```

```
my $d = $c // 3;
```

```
say $d; 0
```

```
my $e = 0;
```

```
my $f = $e || 4;
```

```
say $f; 4
```

