

PHPerKaigi 2023 のトークン問題で ボツにした問題を供養する

nsfisis (いまむら)

第 149 回 PHP 勉強会@東京

自己紹介

nsfisis (いまむら)



@ デジタルサーカス株式会社

PHPerKaigi とは

2018 年から開催されている PHP のカンファレンス

今年 2023 年も、3 月 23 日から 25 日に開催

PHPer チャレンジとは

PHPerKaigi の企画の 1 つ

#から始まる文字列 (トークン) を、
公式サイトやスポンサーブログから探す

例: #PHP

トークン問題とは

PHPer チャレンジのトークンを、PHP のソースコードに隠す
実行したり解読したりするとトークンが入手できる

デジタルサーカスのトークン問題

今年も問題を作成しました (全 5 問)

ボツ問になった問題のうちの 1 問を供養

Catchline

```
<?php
try {
    f(3);
} catch (Throwable $e) {
    while ($e = $e->getPrevious()) {
        printf('%c', $e->getLine() + 23);
    }
    echo "\n";
}
```

Catchline

```
function f(int $i) {  
  if ($i < 0) f();  
  try {  
    match ($i) {  
      0 => X, // 12行目  
      // (間の空行は省略)  
      2 => X, // 49行目  
      // (間の空行は省略)  
      1, 3 => X, // 57行目  
    };  
  } finally {  
    f($i - 1);  
  }  
}
```


Catchline

```
<?php
try {
    f(3);
} catch (Throwable $e) {
    while ($e = $e->getPrevious()) {
        printf('%c', $e->getLine() + 23);
    }
    echo "\n";
}
```

f() を呼び出して Throwable を捕まえ、何かする

Catchline

```
function f(int $i) {  
  if ($i < 0) f(); // エラー！引数が足りない！  
  try {  
    match ($i) {  
      0 => X, // 12行目: エラー！未定義の定数！  
      // (間の空行は省略)  
      2 => X, // 49行目: エラー！未定義の定数！  
      // (間の空行は省略)  
      1, 3 => X, // 57行目: エラー！未定義の定数！  
    };  
  } finally {  
    f($i - 1);  
  }  
}
```

Catchline

```
<?php
try {
    f(3);
} catch (Throwable $e) {
    while ($e = $e->getPrevious()) {
        printf('%c', $e->getLine() + 23);
    }
    echo "\n";
}
```

f(3) からスタート

Catchline

```
function f(int $i) {  
    if ($i < 0) f();  
    try {  
        // ...  
    } finally {  
        f($i - 1);  
    }  
}
```

1 ずつ減らして、`$i` が負なら引数なしで呼ぶ

Catchline

- $f(3)$
- $f(2)$
- $f(1)$
- $f(0)$
- $f(-1)$
- $f()$ ここで終わり

Catchline

```
<?php
try {
    f(3);
} catch (Throwable $e) {
    while ($e = $e->getPrevious()) {
        printf('%c', $e->getLine() + 23);
    }
    echo "\n";
}
```

`\Throwable::getPrevious()` を順に辿り、(エラーの発生した行数 +23) を

ASCII コード [1] と見做して出力

[1]: VAS Syndrome

Catchline

`\Throwable::getPrevious()`

このエラーの1つ前のエラー (= 大本の原因)

- エラー処理中に別のエラーが起きたとき、元々のエラーを保存する
- 内部利用しているライブラリが投げた例外クラスを、自分で定義した例外クラスでラップする

Catchline

```
<?php
try {
    try {
        throw new \Exception("a");
    } finally {
        throw new \Exception("b");
    }
} catch (\Exception $e) {
    echo $e->getMessage(), PHP_EOL; // => b
    echo $e->getPrevious()->getMessage(), PHP_EOL; // => a
}
```

finally節の中で例外を発生させると、PHP の処理系が勝手に \$previous を設定する

Catchline

```
function f(int $i) {  
  if ($i < 0) f(); // エラー！引数が足りない！  
  try {  
    match ($i) {  
      0 => X, // 12行目: エラー！未定義の定数！  
      // (間の空行は省略)  
      2 => X, // 49行目: エラー！未定義の定数！  
      // (間の空行は省略)  
      1, 3 => X, // 57行目: エラー！未定義の定数！  
    };  
  } finally {  
    f($i - 1);  
  }  
}
```

Catchline

```
<?php
try {
    f(3);
} catch (Throwable $e) {
    while ($e = $e->getPrevious()) {
        printf('%c', $e->getLine() + 23);
    }
    echo "\n";
}
```

(エラーの発生した行数 +23) を ASCII コードと見做して出力

Catchline

```
function f(int $i) {  
  if ($i < 0) f(); // エラー！引数が足りない！  
  try {  
    match ($i) {  
      0 => X, // 12行目: エラー！未定義の定数！  
      // (間の空行は省略)  
      2 => X, // 49行目: エラー！未定義の定数！  
      // (間の空行は省略)  
      1, 3 => X, // 57行目: エラー！未定義の定数！  
    };  
  } finally {  
    f($i - 1);  
  }  
}
```

Catchline

- $12 + 23 = 35$ (#)
- $49 + 23 = 72$ (H)
- $57 + 23 = 80$ (P)

組み合わせで #PHP に

おわりに

来たる 3 月の PHPerKaigi 2023 で
これより数段凝った問題を 5 つ出題します！
みなさまの挑戦をお待ちしております！