Énoncé : Conversion Hex2RGB

### 1ere partie de dojo :

Écrire une fonction pour convertir une couleur en hexadécimal vers une couleur en RGB

Si la couleur est #FF44BB, il faut ignorer le “#’, les 2 premiers caractères correspondent au rouge, les deux suivant au vert et les deux derniers au bleu

Pour convertir de l’hexa vers le decimal, on peut utiliser la fonction php hexdec()

Vérifier que le paramètre rentré correspond bien à une couleur en hexa (c’est à dire 6 caractères entre 0-9 et A-F, précédé par un #)

### 2nde partie de dojo (S’il y a le temps) :

Écrire la fonction permettant de convertir le hexa vers le décimal, sans utiliser hexdec()

<https://fr.wikipedia.org/wiki/Syst%C3%A8me_hexad%C3%A9cimal>

Un entier est écrit comme laconcaténation de ces chiffres, et sa lecture s'effectue de droite à gauche. Sa valeur vaut la somme des chiffres affectés de poids correspondant aux puissances successives du nombre 16. Par exemple, 4D5 vaut

5 \* 160 + 13 \* 161 + 4 \* 16 2 = 1237.

## Correction

**Class** Color

{

**public function** convert($hexa)

{

**if** (*preg\_match*(**'/^#[0-9A-Fa-f]{6}$/'**,$hexa)) {

$hexa = *str\_replace*(**'#'**,**''**,$hexa);

$tab = *str\_split*($hexa, 2);

$r = $this->hexToDec($tab[0]); // ou utiliser directement la fonction php hexdec()

$g = $this->hexToDec($tab[1]);

$b = $this->hexToDec($tab[2]);

**return array**($r, $g, $b);

}

}

**public function** hexToDec($hexa)

{

$values = **array** (0,1,2,3,4,5,6,7,8,9, **'A'**,**'B'**,**'C'**,**'D'**,**'E'**,**'F'**);

$values = *array\_flip*($values);

$hexa = *strrev*($hexa);

$hexaTab = *str\_split*($hexa);

$result=$i=0;

**foreach** ($hexaTab **as** $key=>$item) {

$result += $values[$item]\**pow*(16,$key);

}

**return** $result;

}

}

### Tests

**public function** testColor() {

$rgb = **new** Color;

$this->assertEquals([255,255,255], $rgb->convert(**'#FFFFFF'**));

$this->assertEquals([18, 84, 105], $rgb->convert(**'#125469'**));

$this->assertEquals(**null**, $rgb->convert(**'#ZZZZZZ'**));

$this->assertEquals(**null**, $rgb->convert(**'#######’**));

$this->assertEquals(**null**, $rgb->convert(**'FFFFFF'**));

$this->assertEquals(**null**, $rgb->convert(**'#FFFF'**));

$this->assertEquals(**null**, $rgb->convert(**'#FFFFFFF'**));

}

**public function** testHexToDec()

{

$hexColor = **new** Color;

$this->assertEquals(15, $hexColor->hexToDec(**'F'**));

$this->assertEquals(255, $hexColor->hexToDec(**'FF'**));

$this->assertEquals(4090, $hexColor->hexToDec(**'FFA'**));

}