

第 1 回 PHP の基本

1 – 1. PHP 基本

1 – 1 – 1. 拡張子

PHP ファイルの拡張子は「.php」です。

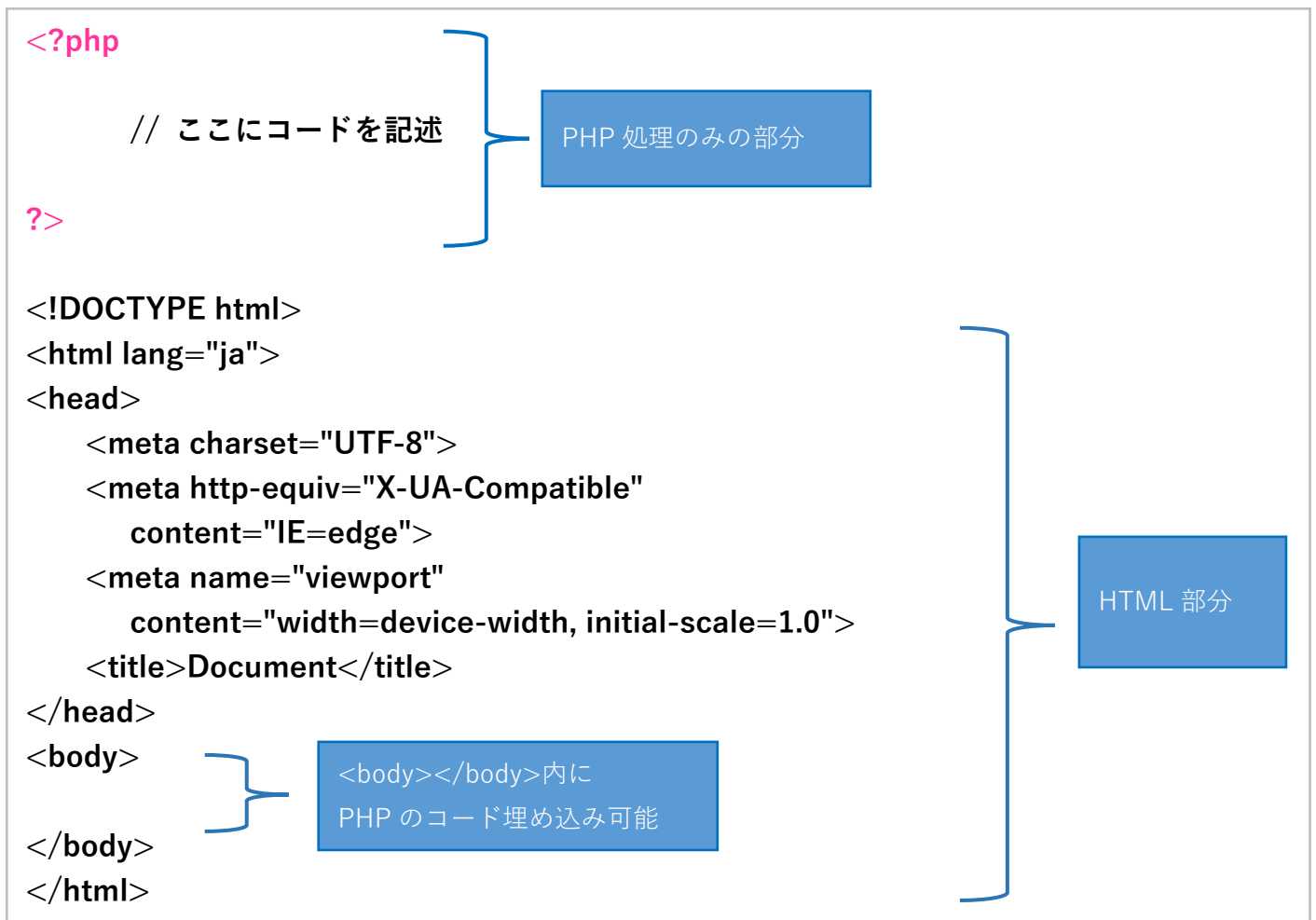
1 – 1 – 2. コードの記述方法

ファイル内のコードは、「<?php ?>」にて記載します。

■sample.php（php のコードのみの時）

```
<?php  
  
    // ここにコードを記述  
  
?>
```

■sample.php（php+HTML の時）



1 - 2. 変数

1 - 2 - 1. 変数の宣言方法

変数は、「\$変数名」として、宣言します。変数名には、英数字、アンダースコア（_）などの文字が使用可能です。

使用【可】 一例	使用【不可】 一例
\$student_name	\$123name ※数字から始まる
\$_name	\$_name ※アンダースコア 2 つは非推奨
\$sum1	\$this ※予約語や特別な変数

■例：

```
$total;          // 変数の宣言のみ
$price = 1200;   // 変数の宣言と初期化
```



PHP では、Java のように変数の宣言時に、型を指定する必要はありません。

変数は格納されているデータの型を自動的に採用します。

※メソッドの引数や戻り値に明示的に記述することは可能です。

1 - 2 - 2. 型

代表的な型は、下記です。

分類	データ型	説明	格納できる値
スカラー型	論理型	真偽値 (boolean)	true または false
	整数型	小数点が付かない正負の数値 (int)	1, -10 などの整数値
	浮動小数点型	小数点を含む実数値 (float, double)	3.14 などの小数
	文字列型	文字の集合 (string)	'abc', 'あいうえお' など
複合型	配列	複数の値をまとめて扱う	
特殊型	null	未定義の値	

1 - 2 - 2 - 1. 文字列型

文字列の値は、「'」（シングルコーテーション）または「"」（ダブルコーテーションで囲みます。

■ 「'」（シングルコーテーション）

変数は展開されません。

■ 「"」（ダブルコーテーション）

変数は展開されます。エスケープシーケンスなどの特殊な文字を扱えます。

■ SampleCode

```
$word = 'ECC 太郎';
```

echo は画面出力文

```
echo 'こんにちは。 {$word} さん!';
```

```
echo '<br>'; //比較の分かり易さため、改行は分けて記述
```

```
echo "こんにちは。 {$word} さん!";
```

```
echo '<br>';
```

```
$quote1 = 'シングルコーテーション¥t で囲んだ文字列'; // ¥t はタブ
```

```
$quote2 = "ダブルコーテーション¥t で囲んだ文字列";
```

```
echo $quote1;
```

```
echo '<br>';
```

```
echo $quote2;
```

■実行結果

```
こんにちは。 {$word} さん!
```

```
こんにちは。 ECC 太郎 さん!
```

```
シングルコーテーション¥t で囲んだ文字列の表示
```

```
ダブルコーテーション で囲んだ文字列の表示
```

変数\$word が展開されず、文字列として表示されている

変数\$word が展開され、内容が表示されている



文字列内で変数を展開する場合は、"こんにちは。 {\$word} さん!" のように、ダブルコーテーションで文字列を囲み、かつ変数を「{ }」で囲みます。

エスケープシーケンスの改行については、ブラウザ上では改行されません。(ソース表示では改行されています。) ブラウザ表示での改行は
を使用しましょう。

1 - 2 - 2 - 2. null

「存在しないこと」を表す特殊な型です。文字列型でも数値型でもありません。数値の「0」や文字列型の「空文字」と似ていますが、コーディングにて区別することができます。



null と、数値の「0」や文字列型の「空文字」を区別しないことも可能ですが、想定外のバグの発生の原因にもなり得ますので、区別するようにしましょう。

1 - 3. 演算子

1 - 3 - 1. 四則演算

演算子	意味
+	加算
-	減算
*	乗算
/	除算
%	剰余
**	累乗

1 - 3 - 2. 論理演算子

演算子	意味
&&	論理積 (AND)
	論理和 (OR)
!	否定 (NOT)

1 - 3 - 3. 比較演算子

下記赤字は要注意箇所。

演算子	説明
==	2つの値が等しいとき true (ゆるやかな比較)
===	2つの値とデータ型が等しいとき true (厳格な比較)
!=	2つの値が等しくないとき true (ゆるやかな比較)
!==	2つの値とデータ型が等しくないとき true (厳格な比較)
<	左辺が右辺より小さいとき true
>	左辺が右辺より大きいとき true
<=	左辺が右辺以下のとき true

>=	左辺が右辺以上のとき true
<>	!=と同じ

1 - 3 - 3 - 1. echo での文字や変数連結について



echo での画面出力時に、文字列や変数を連結したい場合は、「,」（カンマ）もしくは「.」（ドット）を使用します。

「,」と「.」では内部の挙動が異なります。

■ 「,」（カンマ）を使用した場合

```
echo $num,$word;
```

- ① \$num を echo する
- ② \$word を echo する

■ 「.」（ドット）を使用した場合

```
echo $num.$word;
```

- ① \$num と \$word を連結した文字列を作成する
- ② 作成した文字列を echo する

新しく文字列を作成している分、文字列が長くなるほど「.」の方がメモリの消費量が増えることを念頭に入れておきましょう。