

3-2 データの容れもの(変数/定数と配列)

変数/定数

<code>age = 14</code>	<code># => age という名前の変数に 14 という値を代入する</code>
<code>puts age</code>	<code># => age という名前の変数の中身を表示する</code>
<code>address = "東京"</code>	<code># => address という名前の変数に "東京" という値を代入する</code>
<code>puts "住所: #{address}"</code>	<code># => address という名前の変数の中身を表示する</code>
<code>SIZE = 100</code>	<code># => SIZE という定数に 100 という値を代入する</code>

- ・変数(定数)はデータ保存用の器。
- ・変数は値の変更ができるが、定数は値の変更ができない。
- ・1つの変数(定数)には1つのデータしか入れることができない。



配列

<code>friends = ["池田", "田中", "佐藤"]</code>	<code># => friends という名前の配列に</code> <code># "池田", "田中", "佐藤" という 3 つの値を代入する</code>
<code>puts friends[0]</code>	<code># => 1 番目の要素の値を表示する (池田)</code>
<code>puts friends[1]</code>	<code># => 2 番目の要素の値を表示する (田中)</code>
<code>puts friends[2]</code>	<code># => 3 番目の要素の値を表示する (佐藤)</code>
<code>friends.each do x </code>	<code># => friends という名前の配列に収録されている値の個数分だけ繰り返す</code>
<code>puts x</code>	<code># => friends という名前の配列に収録されている値を表示する</code>
<code>end</code>	

- ・配列はデータ保存用の器。
- ・1つの配列にはいくつでもデータを入れることができ、
配列の中に入れたそれぞれのデータは 0 から始まる番号で区別される。
- ・似た内容のデータをリストのようにまとめて入れておくと効果的。
(each do～end を使うと、収録されているデータの数だけ繰り返し処理をおこなうことができる)



※変数、配列ともに、できるだけ簡潔でわかりやすい名前を付けるのがプログラミングのコツ。