

# Java

配列 多次元配列

## 11 時間目

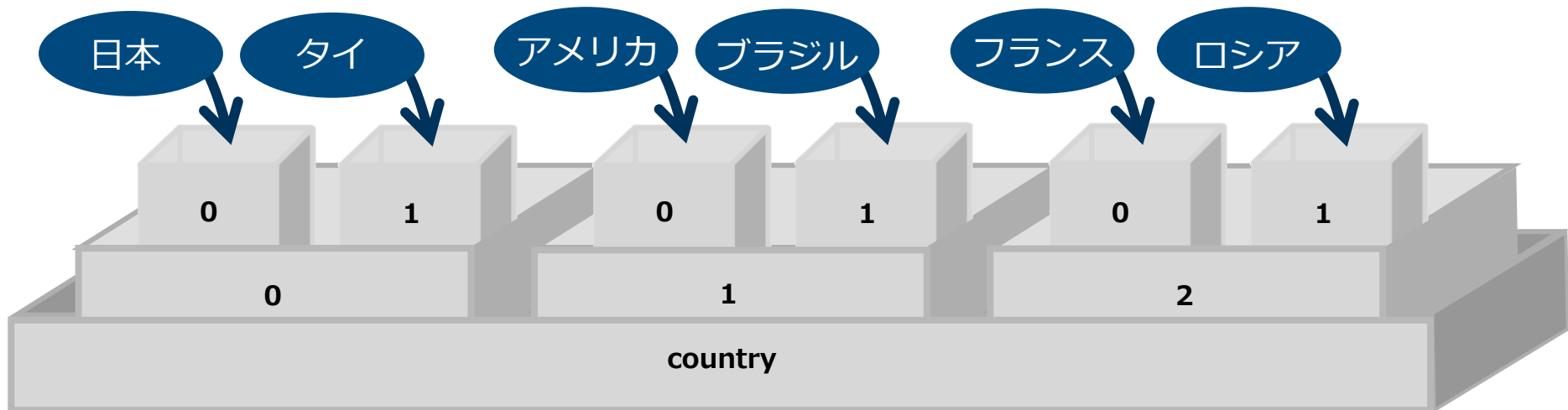
多次元配列とは、

配列の各箱（要素）の中に

さらに小さな箱（要素）を入れて管理する方法

# 多次元配列①

```
String[][] country = {  
    {"日本", "タイ"},  
    {"アメリカ", "ブラジル"},  
    {"フランス", "ロシア"}  
};
```



Javaファイル

```
String[][] country =  
    {  
        {"日本", "タイ"},  
        {"アメリカ", "ブラジル"},  
        {"フランス", "ロシア"}  
    };  
  
System.out.println(country[0][0]);
```

日本

Javaファイル

```
String[][] country =  
    {  
        {"日本", "タイ"},  
        {"アメリカ", "ブラジル"},  
        {"フランス", "ロシア"}  
    };  
  
System.out.println(country[0][1]);
```

タイ

Javaファイル

```
String[][] country =  
    {  
        {"日本", "タイ"},  
        {"アメリカ", "ブラジル"},  
        {"フランス", "ロシア"}  
    };  
  
System.out.println(country[1][0]);
```

**アメリカ**

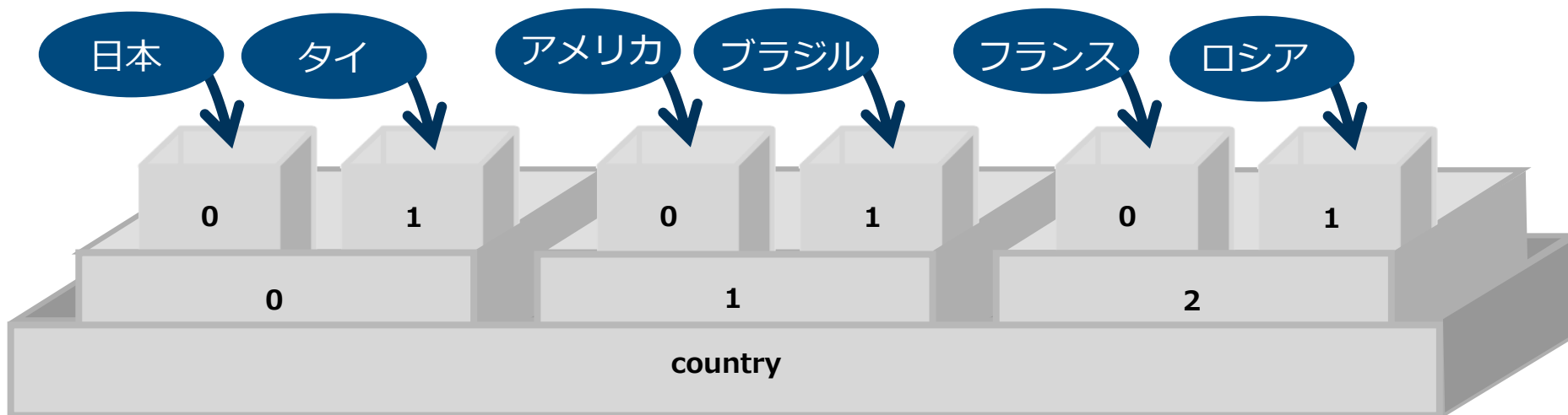
Javaファイル

```
String[][] country =  
    {  
        {"日本", "タイ"},  
        {"アメリカ", "ブラジル"},  
        {"フランス", "ロシア"}  
    };  
  
System.out.println(country[2][1]);
```

ロシア

## 多次元配列②

```
String[][] country = new String[3][2];  
country[0][0] = "日本";  
country[0][1] = "タイ";  
country[1][0] = "アメリカ";  
country[1][1] = "ブラジル";  
country[2][0] = "フランス";  
country[2][1] = "ロシア";
```





Javaファイル

```
String[][] country = new String[3][2];
```

```
country[0][0] = "日本";  
country[0][1] = "タイ";  
country[1][0] = "アメリカ";  
country[1][1] = "ブラジル";  
country[2][0] = "フランス";  
country[2][1] = "ロシア";
```

```
System.out.println(country[0][0]);
```

日本

Javaファイル

```
String[][] country = new String[3][2];
```

```
country[0][0] = "日本";  
country[0][1] = "タイ";  
country[1][0] = "アメリカ";  
country[1][1] = "ブラジル";  
country[2][0] = "フランス";  
country[2][1] = "ロシア";
```

```
System.out.println(country[1][1]);
```

**ブラジル**