

質問1 次のクラスを下のように宣言しました。コンパイルしたら“メソッド本体がないか、**abstract** として宣言されています。”とエラーメッセージが出力されました。どういう意味ですか？

フィールド：

```
int x;                // 変数 x
```

メソッド：

```
void setX(int px);    // 値 px を変数 x に設定するメソッド
```

```
int getX();           // 変数 x の値を取得するメソッド
```

```
class MyValue
{
    /// フィールド
    int x;

    /// メソッド
    // 値を設定するメソッド
    void setX(int px);
    {
        x=px;
    }
    // 値を取得するメソッド
    int getX();
    {
        return x;
    }
}
```

回答

クラスがもつメソッドを記載するとき、

```
void setX(int px);    // 値 px を変数 x に設定するメソッド
```

のように、メソッドの名前と引数、戻り値、処理の説明のみを記載し、本体コード部は省略されることが多いです。引数の後ろのセミicolon“;”は本体コード部の省略を意味します。

このように記載する理由は、利用者がメソッドを使用する場合、その名前と引数、戻り値、処理内容が分かれば十分であり、本体コード部を知る必要がないためです。

この例の場合、名前は setX、引数は int 型 1 つ、戻り値は持たないメソッドを表しています。実際にこのメソッドの本体コード部をプログラムする場合は、

```
void setX(int px)
{
    本体コード部
}
```

のように記述します。

質問2 次のコードをコンパイルしたら“構文解析中にファイルの終わりに移りました”とエラーメッセージが出力されました。どういう意味ですか？

```
class Car
{
    int num;
    double gas;
}

class Sample
{
    public static void main(String[] args)
    {
        Car mycar = new Car();
        mycar.num=1234;
        mycar.gas=10.5;
    }
}
```

回答

このエラーはブロックの括弧のペアが組になっていないことを示しています。上の例では、main()メソッドのブロックの閉括弧が抜けていますね。

ブロックを書くときに開括弧と閉括弧を一緒に書いてしまいましょう。そうするとこのミスを上手に防ぐことができます。