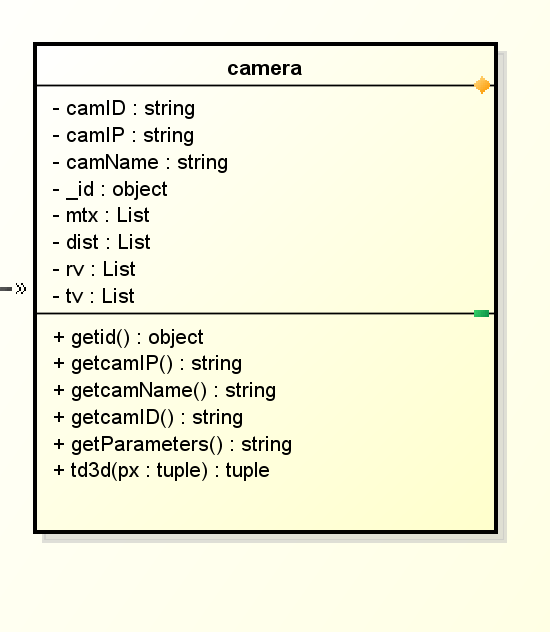
**クラス図**

* クラス図はクラスの情報・クラス間の関係から、システムの静的な構成を表示する
* クラスはオブジェクト情報を抽象化して、オブジェクト型を定義する
* システムの構成の種類：
  + 動的な構成
  + 静的な構成
* クラス図はシステムの骨組みになります

表記法：

* 長方形の枠を三に分ける
  + 上段はクラス名
  + 中段は属性
  + 下段は関数・操作



**可視性：**

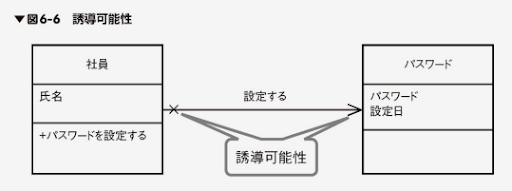
* 属性や操作に対して、システム内におけるアクセス可能範囲を定義する
* 可視性の種類：
  + Public (+)：全てのクラスからアクセス可能
  + Private (-)：自クラスのみアクセス可能
  + Protected (#)：自クラスと継承されているクラスのみアクセス可能
  + Package (~)：同じパッケージ内のクラスのみアクセス可能

関連：

* 二つのクラスが何らかの関係を持つ場合、関連として表示します
  + 他のクラスのデータを属性として保持する場合
  + 他のクラスの操作・関数を呼び出す場合
* 関連をクラス間に線を引いて表示する
* 関連を名称する時、システム全体から見るのが重要

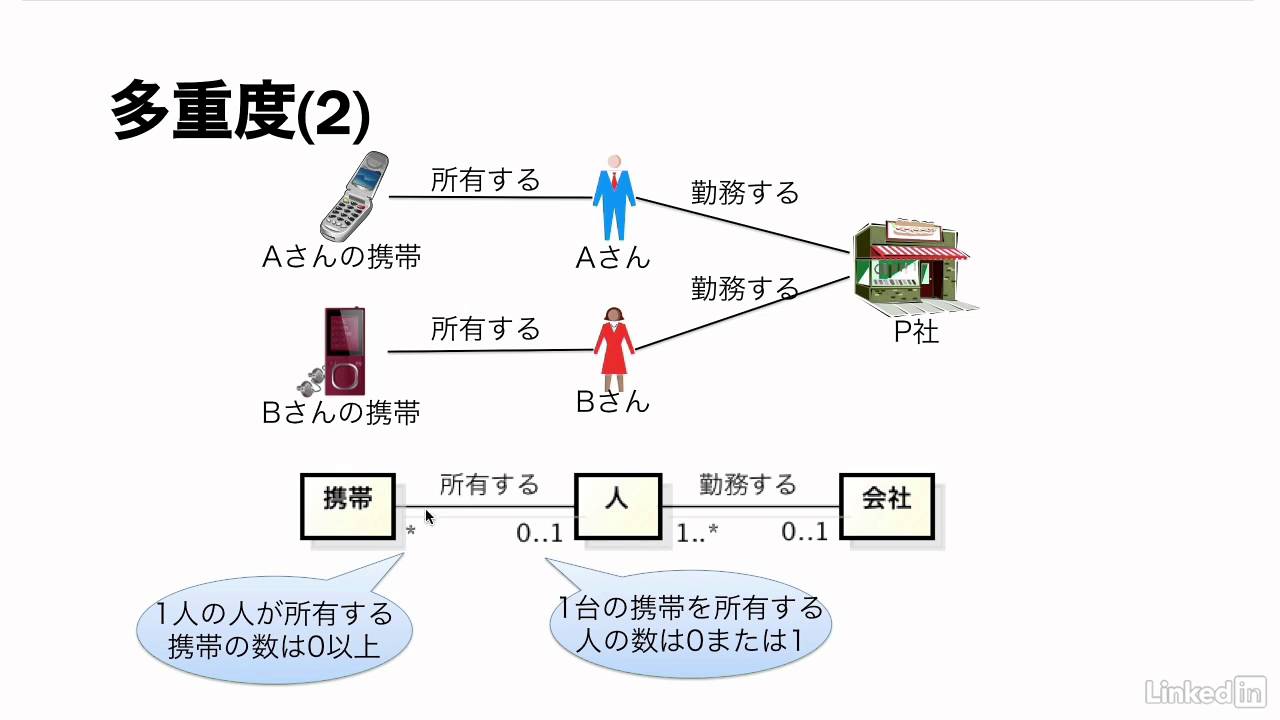
**誘導可能性：**

一方のオブジェクトからもう一方のオブジェクトにしかメッセージを送らず、逆の方向は行われない場合矢印にxを付ける



**多重度：**

* 記号の種類：
* \* : 0以上
* 1 : 1のみ
* 0..\* : 0以上
* 1..\* : 1以上
* 0..1 : 0や1
* 1..3 : 1~3
* 1,3,5 : 1や3や5

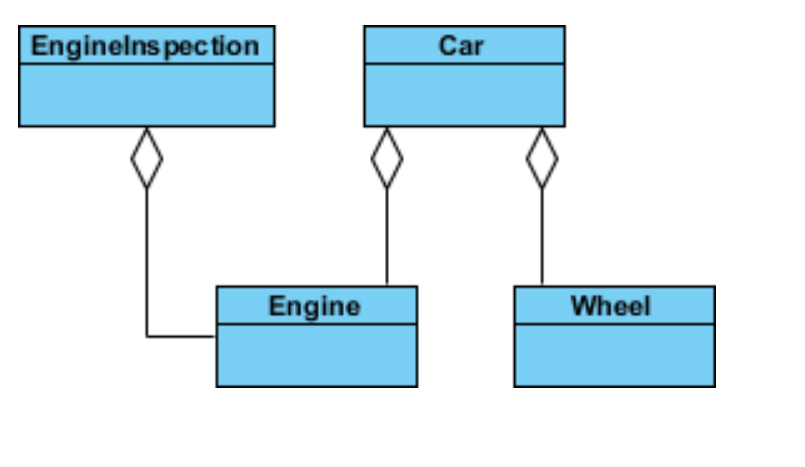


**限定子（qualifier）：**

* 一対多の関係にあるクラス間に、特定のために用いる属性を表示するものとなります
* 限定子を使うことで、「1..\*」の多重度の関係が「1」になります

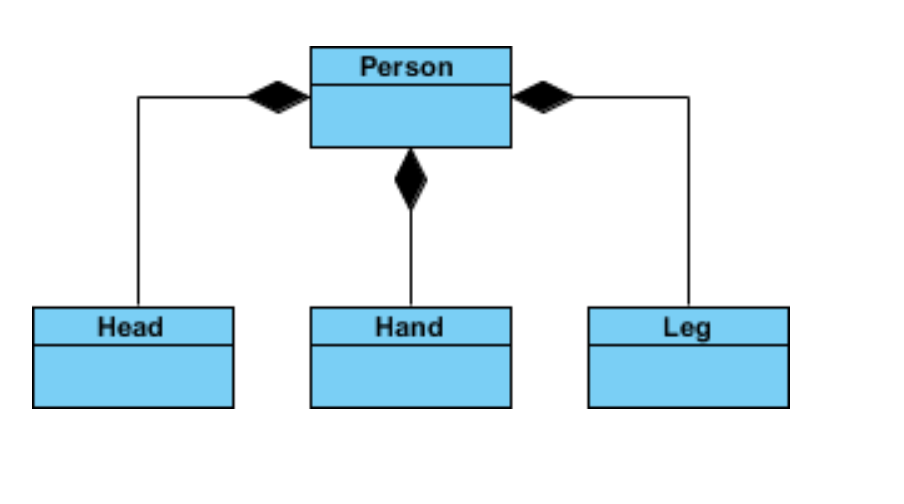
**集約（Aggregation）：**

* クラス間に「全体・一部分」の関係がある時を表す
* 全体に当たるクラスが部分に当たるクラスを包含する



コンポジション (composition)：

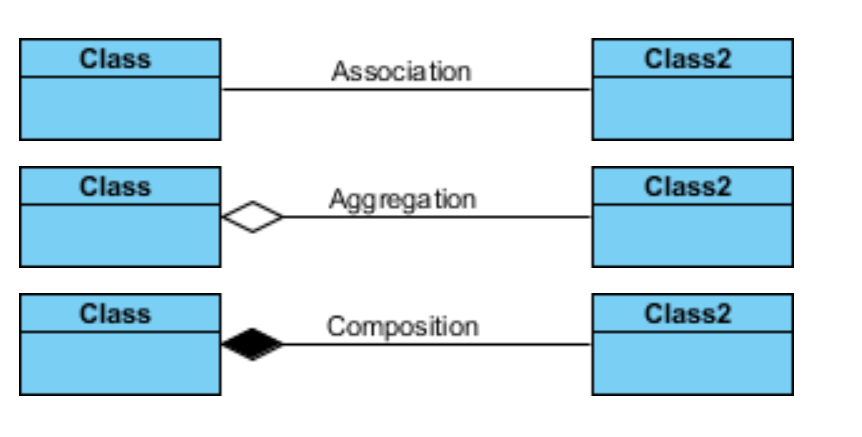
* ある全体インスタンスがある部分インスタンスを所有する時、部分インスタンスの生成・削除は全体インスタンスが制御する場合に使います
* 集約 (aggregation)よりも強い関係



**依存（Dependence）：**

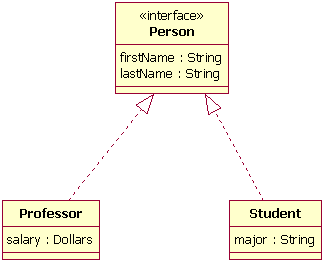
* オブジェクト間に利用される可能性を表す
* 一方のクラスの操作において影響を受ける関係
* あるモデル要素が他のモデル要素を参照としている関係

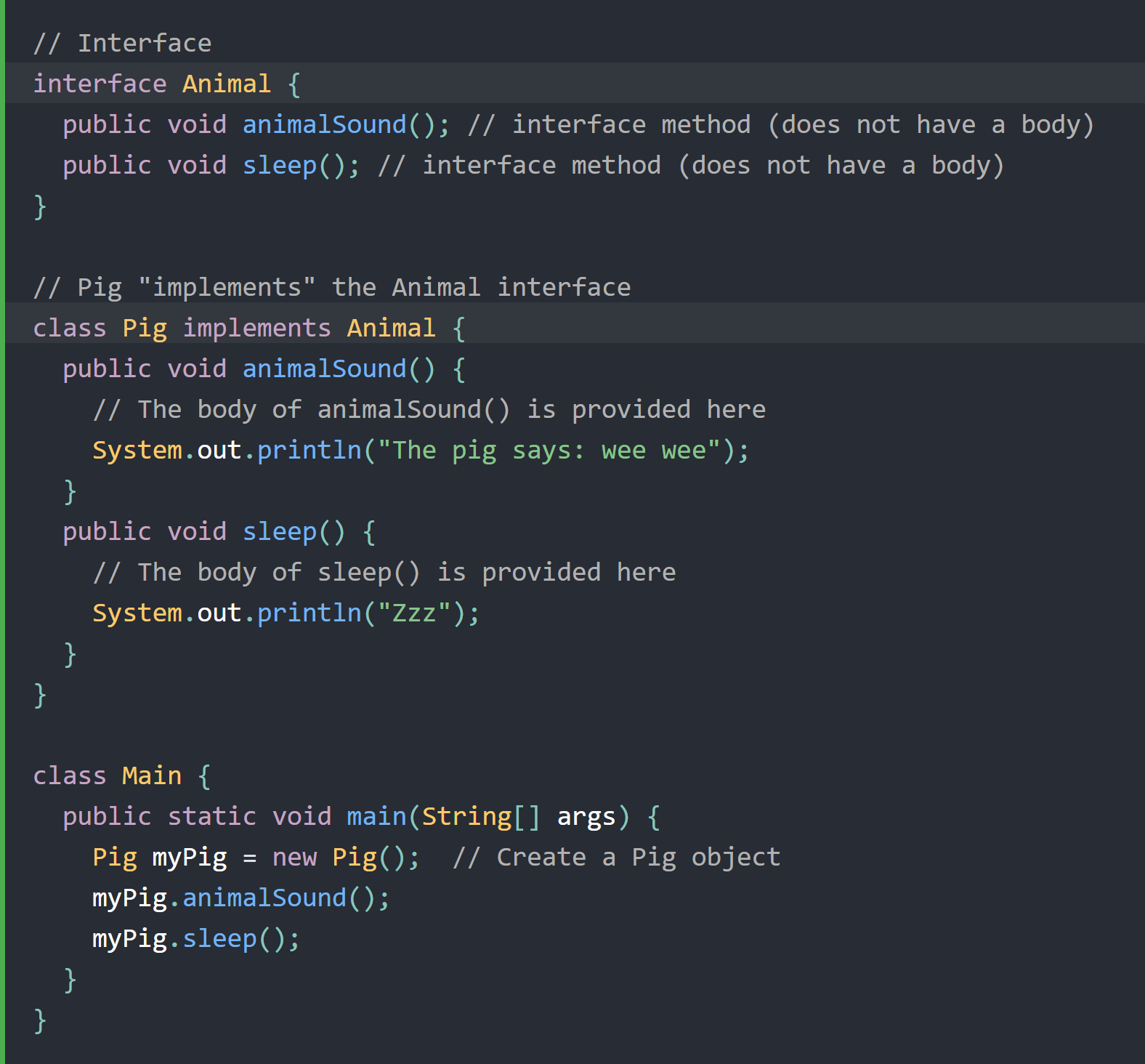
**――――――**



**インタフェース：**

* クラスが実現すべき操作・関数のみが定義された特別なクラス
* 具体的な定義はしない





**クラス図の書き方：**

* 分析工程において、ユースケース図を基にクラスを抽出して、クラス間の関連・操作を「分析クラス図」で表示します
  + ユースケース図やオブジェクト図からクラスを抽出する
  + クラス間の関連を作成する
  + 属性、多重度、関連名、関連端名を作成する
    - 明瞭しない場合、無理して名前を付ける必要はありません
  + クラスの操作、汎化関係、サブクラスを抽出する
* 設計クラス図は、分析クラス図より詳細を表現したクラス図となります
  + 実装しながらクラス図を修正するのも問題ありません
  + 実装とクラス図の整合性が取れているのが重要
  + シーケンス図やアクティビティ図を基に作成します
  + データ型・引数等を表示する