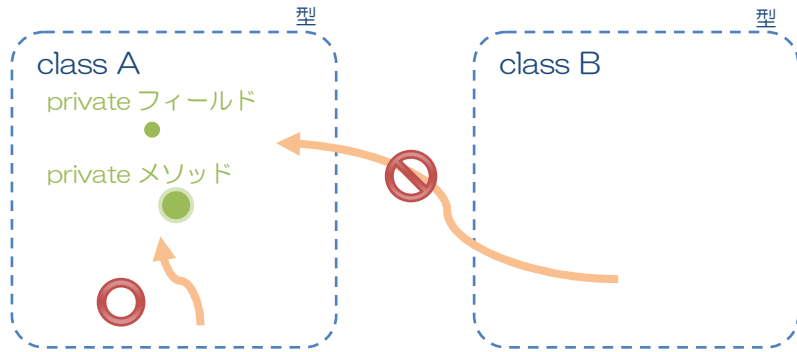


**アクセス制限**

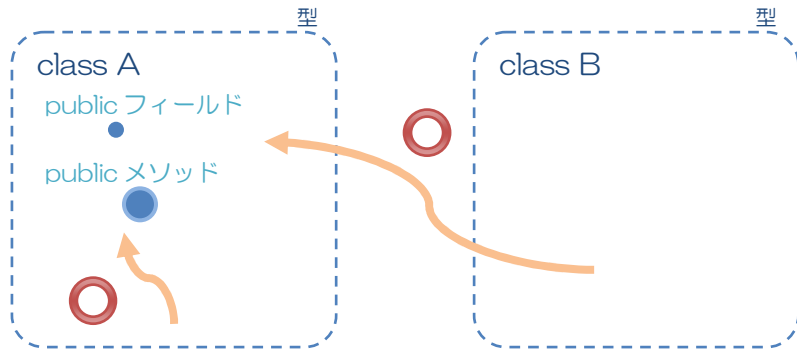
private メンバ 同じクラスからのみアクセスできるメンバです

宣 言 メンバの宣言に private 修飾子を付けます



public メンバ どこからでもアクセスできるメンバです

宣 言 メンバの宣言に public 修飾子を付けます



その他の修飾子

場所	private	指定ない	protected	public
同じクラス	○	○	○	○
同じパッケージ内のサブクラス	×	○	○	○
同じパッケージ内の一般クラス	×	○	○	○
他のパッケージ内のサブクラス	×	×	○	○
他のパッケージ内の一般クラス	×	×	×	○

※パッケージについては13章に説明があります

## カプセル化

### 抽象データ型

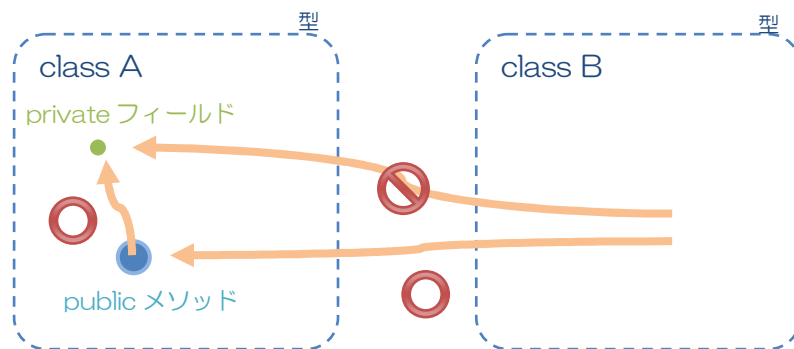
データとそれにアクセスする手続きを一つにまとめたデータ型です  
抽象データ型を実現する方法の1つがクラスです

### カプセル化

抽象データ型を用いることにより内部のデータへのアクセスを与えられた  
手続きを用いてのみ可能にして、内部の細かなデータやその構成を外部  
から隠蔽することです

クラスでは、フィールドを private メンバにメソッドを public メンバとし、フィールドにア  
クセスする時は public メソッドを用いることでカプセル化を実現します

メソッドに値のチェック機能を付けてフィールドを保護するデータの保護の他、データの保  
守や独立性に貢献します



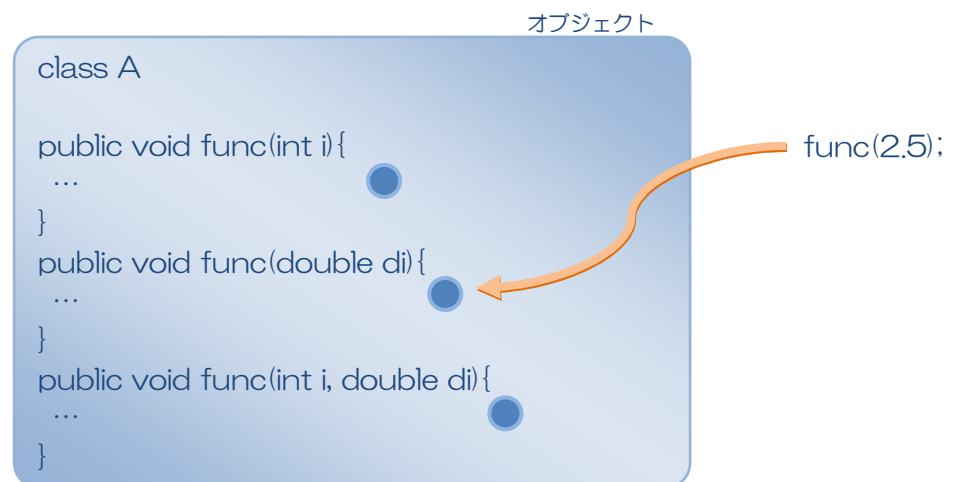
## オーバーロード

### オーバーロード

1つのメソッドに複数の機能を持たせることです

### 宣言

同じ名前のメソッドを同じクラス内に複数個定義します  
但し、各メソッドの引数の型・個数は異なること(戻り値は同じでもよい)  
※メソッドの呼出し時にこれを利用してどのメソッドが判別します



## ポリモーフィズム (多態性/多様性)

一つの対象が状況に応じて別々の働きをすることです  
ポリモーフィズムの実現の1つがメソッドのオーバーロードです