

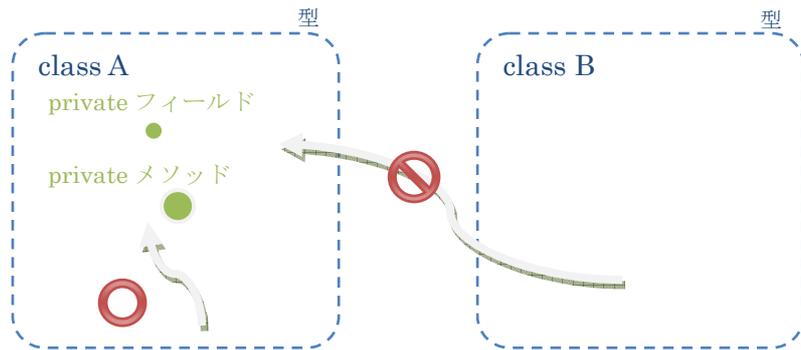
## JavaプログラミングII

### 3回目 クラスの機能（1）アクセス制限、オーバーロード

#### アクセス制限

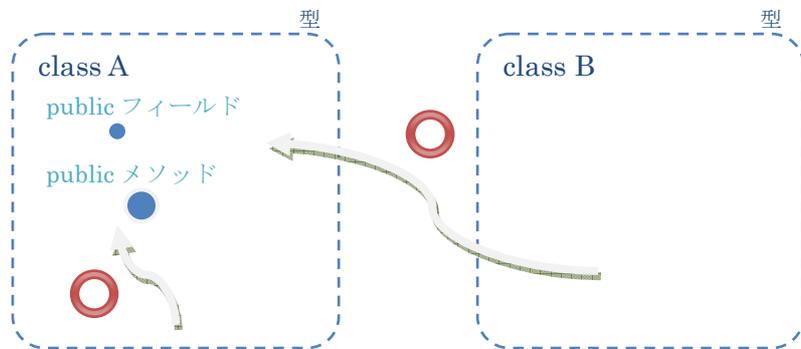
**private** メンバ 同じクラスからのみアクセスできるメンバ

宣言 メンバの宣言に **private** 修飾子を付ける



**public** メンバ どこからでもアクセスできるメンバ

宣言 メンバの宣言に **public** 修飾子を付ける



#### 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	×	×	×	○

## カプセル化

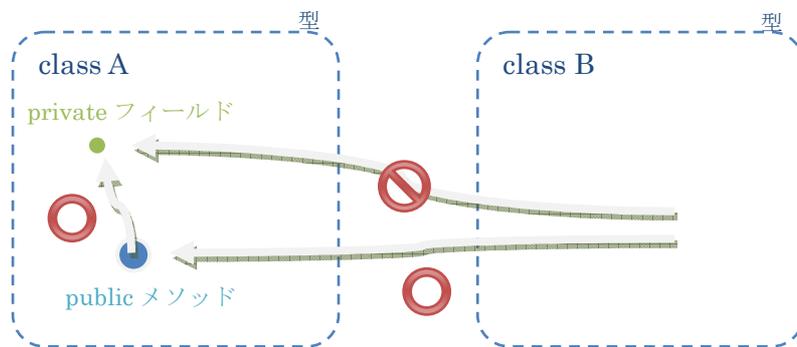
抽象データ型

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  
抽象データ型を実現する方法の1つがクラス

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

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する。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



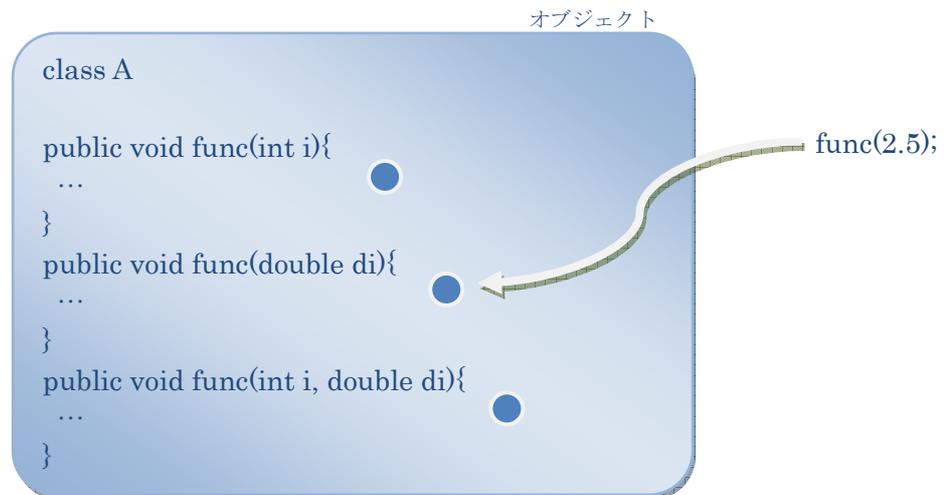
## オーバーロード

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

宣言

同じ名前のメソッドを同じクラス内に複数個定義する  
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)  
※Java はメソッドの呼出し時にこれを利用してどのメソッドか判別する



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  
ポリモーフィズムを実現する方法の1つがメソッドのオーバーロード