

JavaプログラミングII

8回目 抽象クラスとインタフェース

抽象クラス

抽象メソッドとは 処理内容が定義されない空のメソッド

宣言 メソッドの修飾子に *abstract* を記述
このとき、メソッドの処理内容は書かない

抽象クラスとは 0 または 1 個以上の抽象メソッドをメンバーにもつクラス

宣言 クラスの修飾子に *abstract* を記述
1 個以上の抽象メソッドをメンバーにもつ場合
→ 必ず *abstract* を記述
※クラスが抽象メソッドを含むことを明示
0 個の抽象メソッドをメンバーにもつ場合
→ 修飾子 *abstract* により抽象メソッドをもたない抽象クラスとなる

特徴 抽象クラスのオブジェクトを直接は作成できない
クラス型変数としての利用は可能

利用 継承してすべての抽象メソッドをオーバーライドする
オーバーライドされない場合 → サブクラスは抽象クラス

instanceof 演算子

参照 *instanceof* クラス名

演算結果

true ← 左辺のオブジェクトのクラスが右辺のクラス、またはそのサブクラス
false ← それ以外

インタフェース

インタフェースとは 抽象メソッド、定数をメンバーにもつ型（インタフェース型）
※その他、クラスやインタフェースをもつことができる

宣 言 `interface インタフェース名{
 メンバー
}`

ただし、インタフェースとそのメンバーは暗黙的に次の修飾子となる

`interface` → `abstract`
メソッド → `public abstract`
変 数 → `public static final`

特 徴 インタフェース型のオブジェクトを直接は作成できない
インタフェース型変数（クラス型変数のように）としての利用は可能

利 用 実装してすべての抽象メソッドをオーバーライドする
オーバーライドされない場合 → 実装されたクラスは抽象クラス

定数は `インタフェース名.変数名` でアクセス
※クラス変数と同じアクセス方法

実 装 クラスと組み合わせること
※継承と似ているが異なることに注意

`class クラス名 implements インタフェース1 インタフェース2 . . . {
 クラスメンバー
}`

インタフェースは多重継承の仕組みを実現する
※同じ抽象メソッド名がある場合は1つとみなされる

インタフェースの拡張

インタフェースはクラスと同じように継承による拡張が可能

宣 言 `interface サブインタフェース extends スーパーインタフェース1 スーパーインタフェース2 . . . {
 メンバー
}`

インタフェースは多重継承の仕組みを実現する
※同じ抽象メソッド名がある場合は1つとみなされる

抽象メソッドのオーバーロード/オーバーライドは同様になされる