Polymorphism

Cheat Sheet - Informatik 1

Source

Merker E. (2004) Objektorientierung mit Java. In: Grundkurs Java-Technologien. Vieweg+Teubner Verlag.
https://doi.org/10.1007/978-3-322-80261-3_2

Polymorph Variables

The type of a variable (eg. "Game") is set when the variable is declared:

```
Game game;
```

The variable "game" can be assigned any object that is of type "Game", it can also be assigned an object that is of a class inheriting from "Game" (eg. "Soccer"):

```
game = new Soccer();
```

An array of type Game could hold objects of any class that inherits from Game.

Dynamic Binding

A class "Game" might implement a method "play()". The classes "Basketball" and "Soccer" inherit from "Game" and override the available "play()"-method.

Dynamic Binding means that it is only decided during runtime, which of the "play()"-methods will be called - depending on which type the object is, on which the method is called.

```
private Game decideOnGame() {
  if(this.lovesSoccer) {
    return new Soccer();
  }
  else {
    return new Basketball();
  }
}
// ...
Game game = decideOnGame();
game.play();
```

Method Overriding

A class (eg. "Game") implements a public, non-static method (eg. "play()"). Several classes (eg. "Soccer", "Basketball") that inherit from class "Game" can override the parent's method "play()" (it needs to have the same signature).

If the parent's method is overridden in the child class, the child's method will be called whenever "play()" is called on an object of the child class.

```
public class Soccer extends Game {
  public void play() {
    // overrides parent's method
    runToBall(); // and so on...
    super.play(); // call parent's
                  method
  }
}
```

Method Overloading

A class may have multiple versions of a method. They have an identical name and return type but differ in their parameters.

Depending on the parameters passed in a method call, the according method is chosen.

```
public void kickBall(){
  System.out.println("a player kicks
  the ball");
}

public void kickBall(Direction dir){
  System out.println("a player kicks
  the ball in direction " +
  dir.toString());
}
```