

## ▶ キーワード

抽象クラス、抽象メソッド、abstract、

ポリモフィズム、多態性



## ▶ 板書

1

▶ 2種類の継承

① 機能追加

```

graph BT
    Person --> Student
    Person --> Teacher
  
```

② 共通点まとめ

```

graph BT
    Club --> TandF
    Club --> Football
  
```

```

class TandF {
    String name;
    TandF(String name) {
        this.name = name;
    }
    void display() {
        System.out.println("部活:" + name);
    }
    void practice() {
        System.out.println("ウォームアップ");
    }
}

class Football {
    String name;
    Football(String name) {
        this.name = name;
    }
    void display() {
        System.out.println("部活:" + name);
    }
    void practice() {
        System.out.println("ドリブル練習");
    }
}
  
```

2

▶ 抽象クラスとポリモフィズム

- 実装内容を持たないメソッドを \_\_\_\_\_ とよび \_\_\_\_\_ をつける
- 抽象メソッドをもつクラスを \_\_\_\_\_ とよび \_\_\_\_\_ をつける。インスタンス化は \_\_\_\_\_

```

abstract class Club {
    String name;
    Club(String name) {
        this.name = name;
    }
    void display() {
        System.out.println("部活:" + name);
    }
    abstract void practice();
}

class Student {
    Club club;
    Student(Club club) {
        this.club = club;
    }
    void display() {
        club.display();
    }
    void practice() {
        club.practice();
    }
}
  
```

```

graph BT
    Club --> TandF
    Club --> Football
  
```

- Studentクラスのメソッドは clubオブジェクトによって異なる処理 = \_\_\_\_\_

## ▶ プログラム

Club.java

```
1 | public abstract class Club{
2 |     private String name;    // 部活名
3 |
4 |     public Club(String name){
5 |         this.name = name;
6 |     }
7 |     public void display(){
8 |         System.out.println("部活動：" + name);
9 |     }
10 |     public abstract void practice();
11 | }
```

TandF.java

```
1 | public class TandF extends Club{
2 |     public TandF(String name){
3 |         super(name);
4 |     }
5 |     public void practice(){
6 |         System.out.println("ウォームアップ");
7 |         System.out.println("インターバル");
8 |         System.out.println("筋肉トレーニング");
9 |     }
10 | }
```

### Football.java

```
1 | public class Football extends Club{
2 |     public Football(String name){
3 |         super(name);
4 |     }
5 |     public void practice(){
6 |         System.out.println("ドリブル練習");
7 |         System.out.println("シュート練習");
8 |         System.out.println("ミニゲーム");
9 |     }
10 | }
```

### Student6.java

```
1 | public class Student6{
2 |     private String name;    // 氏名
3 |     private Club club;      // Club オブジェクト
4 |
5 |     public Student6(String name, Club club){
6 |         this.name = name;
7 |         this.club = club;
8 |     }
9 |     public void display(){
10 |         System.out.println("名前：" + name);
11 |         club.display();
12 |     }
13 |     public void practice(){
14 |         club.practice();
15 |     }
16 | }
```

## StuSample6.java

```
1 public class StuSample6{
2     public static void main(String[] args){
3         TandF taf = new TandF("陸上競技部");
4         Football fb = new Football("サッカー部");
5
6         Student6 stu1 = new Student6("菅原",taf);
7         stu1.display();
8         stu1.practice();
9
10        Student6 stu2 = new Student6("桜井",fb);
11        stu2.display();
12        stu2.practice();
13    }
14 }
```



＼フリーラーニング（無料で学べる場）をもっと広げたい！／  
チャンネル登録や拡散よろしくお願いします！



NEXT DOOR

