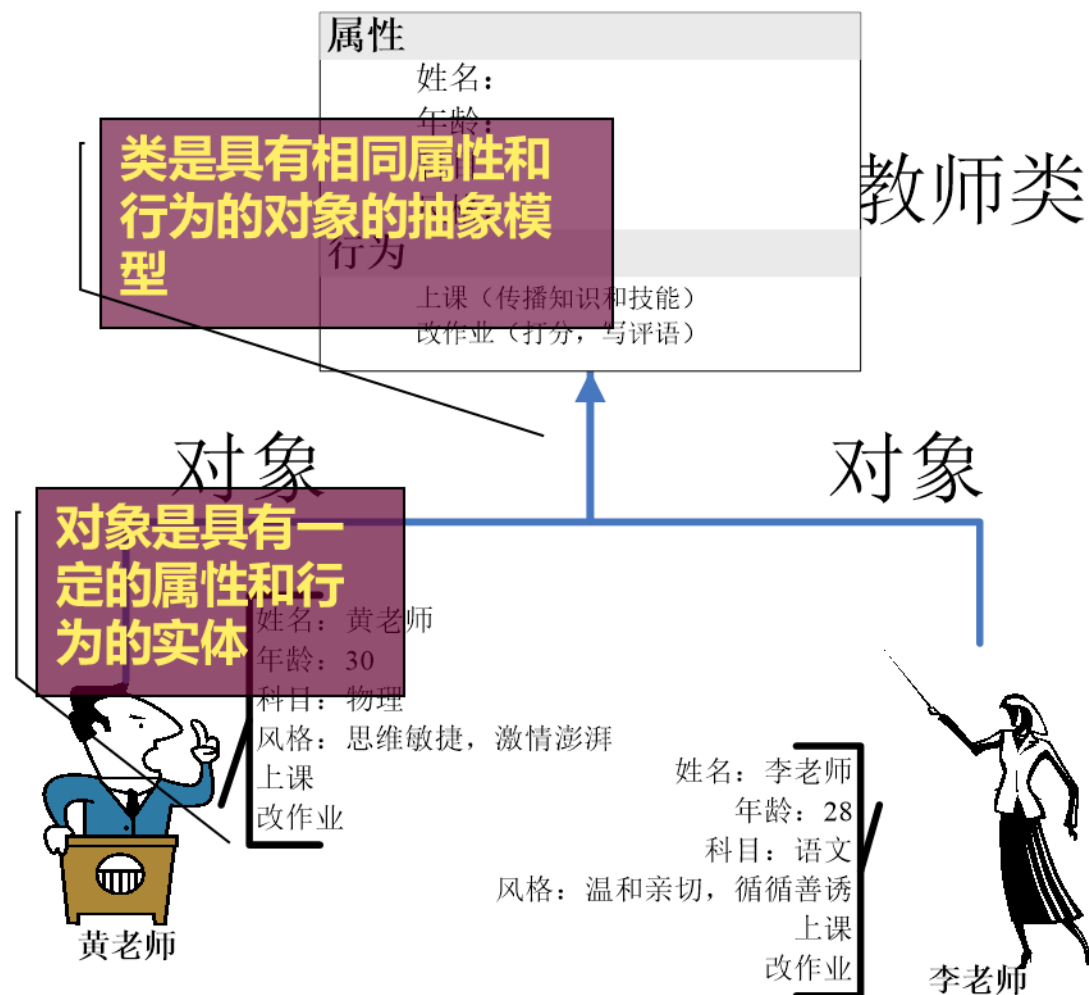


学习任务一技能节报名小程序

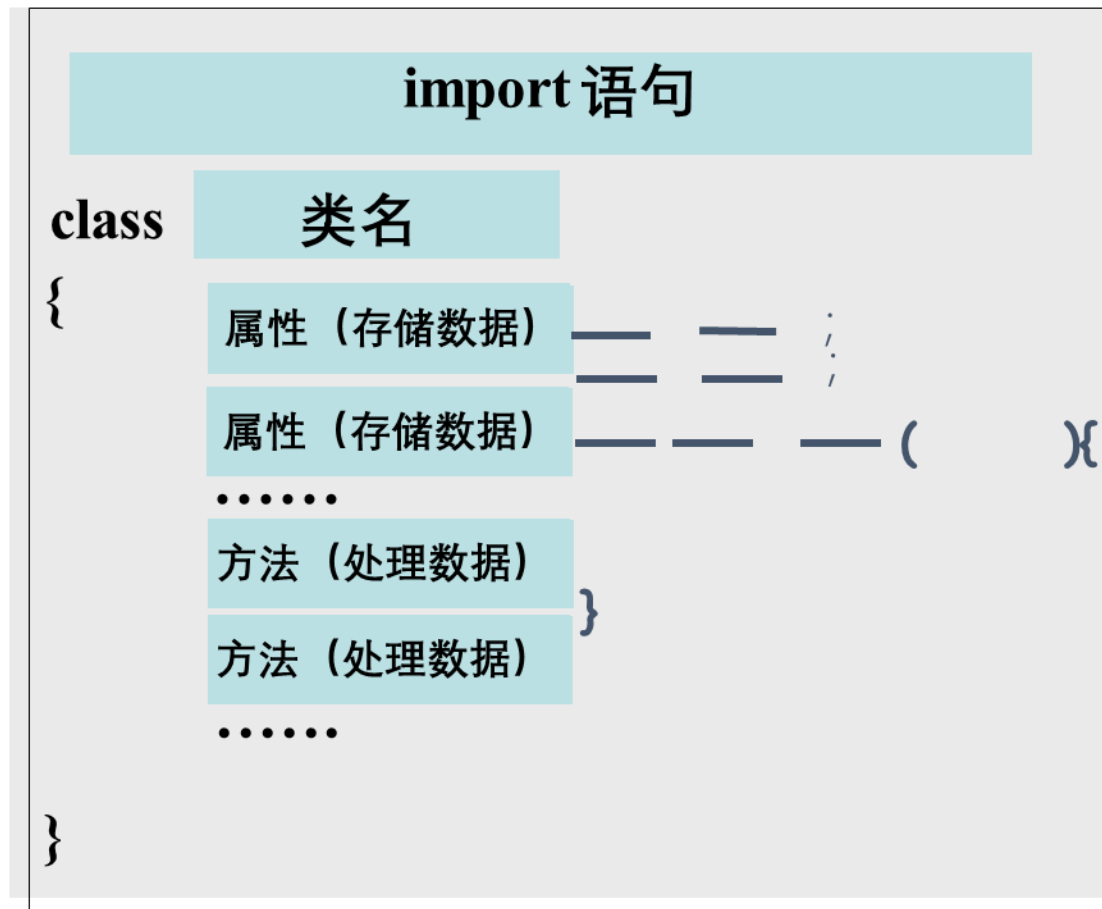
第一次课 设计类



类和对象



类的基本结构是什么



Java类中如何存储简单数据

在需要存储数据的地方，使用如下语法格式：

数据类型 变量名；

或者是

数据类型 变量名=初始值；

例如：

```
int age=10;
```

```
char c='x';
```

```
float rate=0.3f;
```

```
double price=23.3;
```

```
boolean flag=true;
```

如何存储字符串和日期

Java类中一般方法的定义

修饰符

类型

方法的名字

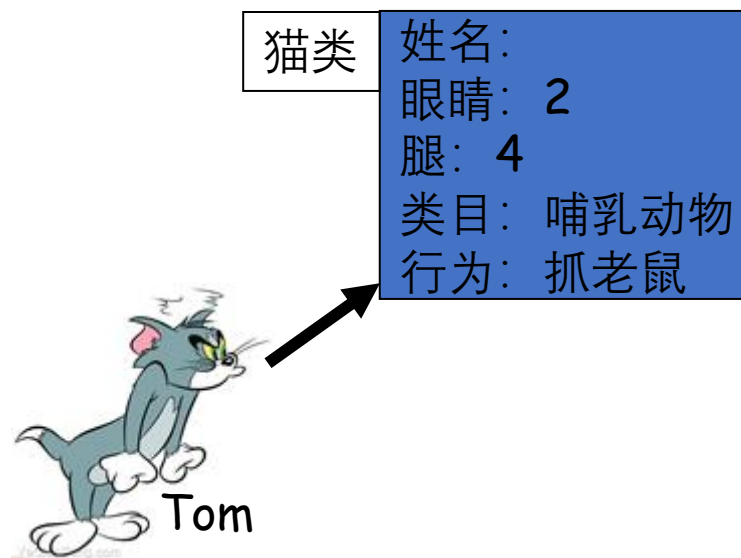
(需要的参数)

{

方法的内容;

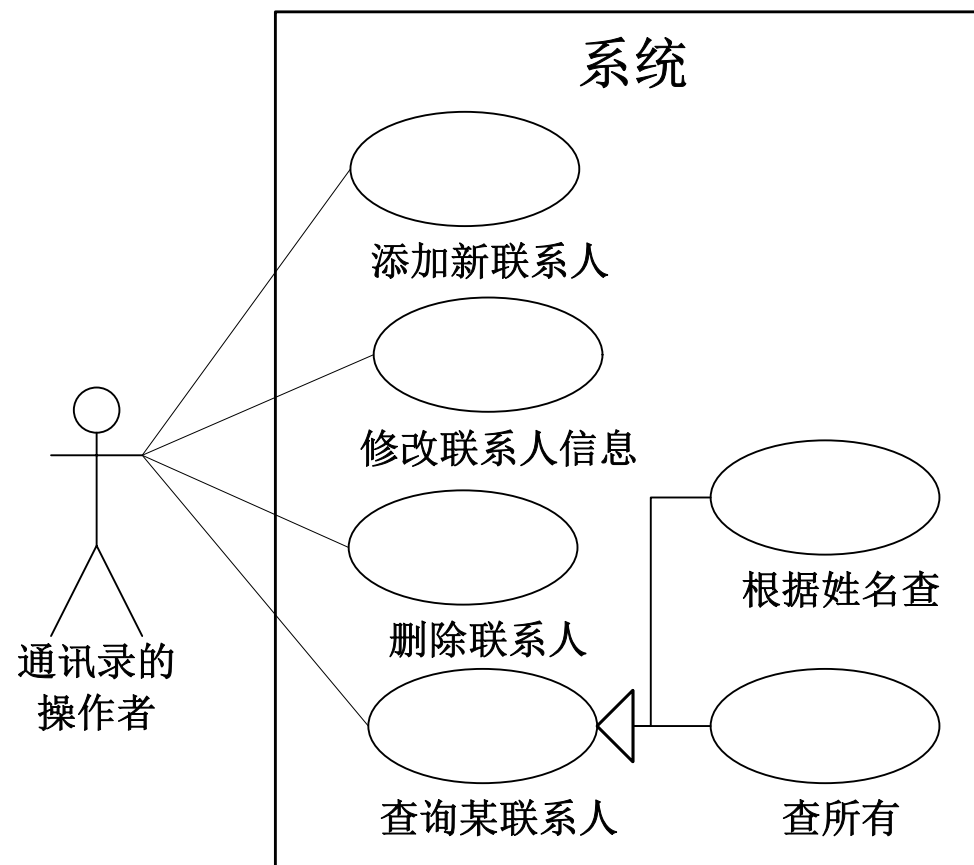
}

new的含义——create,newInstance



用Java解决实际问题的关键

提炼类



操作业务的
类

联系人操作类

属性

无

方法

- 方法一：向数据存储设备添加一个新的联系人信息
- 方法二：修改指定联系人的联系信息
- 方法三：删除指定联系人
- 方法四：查询所有联系人信息
- 方法五：根据姓名查询联系人的联系方式

用Java解决实际问题的关键

存储数据的
类：实体类

联系人

属性

姓名——字符串
手机——字符串
家庭电话——字符串
办公电话——字符串
QQ号码——字符串
EMAIL——字符串
其他——字符串

联系人对象举例

“王五”的联系方式

“李四”

手机：123456
办公电话：12
家庭电话：15212
QQ：212
EMAIL：aa@aa.com
其他：asdf

“张三”

手机

办公

家庭

QQ

EMAIL

其他

手机

办公

家庭

QQ

EMAIL

其他

手机

办公

家庭

QQ

EMAIL

其他

手机

办公

家庭

QQ

EMAIL

其他

Java中的特殊方法

构造函数

方法的重载

```
Class Boy{  
    int age;  
    public Boy(){age=0;}  
    public Boy(int age){this.age=age;}  
}
```

Java中的特殊方法

构造函数

方法的重载

get/set方法

```
private int age;  
public void setAge(int age){this.age=age;}  
public int getAge(){return this.age;}
```

Java中的特殊方法

构造函数

方法的重载

get/set方法

toString方法

- 什么可以被打印
 - `int a=100;System.out.println(a);`
 - `Timer s=new Timer(); System.out.println(s);` ?
 - 结论：对象如要被打印，必须提供明确的toString方法，告诉系统如何打印①

Java中的特殊方法

构造函数

方法的重载

get/set方法

toString方法

静态方法

对比

静态

非静态

public **static** void add()

public void add()

类名.add()

对象名.add()

不可以使用this引用

可以使用this

方法里面不可以调用
同一个类里面的非静态的成员

可以调用静态
的成员方法

声明的形式上

调用的形式上

This引用的使用权

兄弟方法之间的
关系

Java中的特殊方法

构造函数

方法的重载

get/set方法

toString方法

静态方法

方法的重写

```
abstract class Employee {  
    int basic = 2000;  
    abstract void salary();//抽象方法  
}  
class Manager extends Employee{  
    void salary() {  
        System.out.println("薪资等于 "+basic*5);  
    }  
}  
class Worker extends Employee {  
    void salary() {  
        System.out.println("薪资等于 "+basic*2);  
    }  
}
```

Java中的特殊方法

构造函数

方法的重载

get/set方法

toString方法

静态方法

方法的重写

```
interface A {  
    double g=9.8;  
    void show( );  
}
```

```
class B implements A {  
    public void show( ) {  
        System.out.println("g="+g);  
    }  
}
```

用一个综合小例子回顾java基础语法

今天是：2021-10-1

国庆快乐!!!

- 1、这个程序一打开，会出现当前的系统时间
- 2、如果今天是几个常用的节日，会出现相应的提示
如果今天是你的生日，也会出现相应的提示

- 1、要求：两个人一个小组，在www.processon.com绘制类图
- 2、提示：这个小任务，需要几个类，每个类里有什么属性和方法