

内部类和匿名类

内部类

内部类是指将一个类定义在另一类里面，这个类就是内部类。

内部类分为 `static` 内部类和普通内部类

```
1 public class A {  
2  
3     class B{//内部类  
4  
5     }  
6 }
```

这个类编译后会产生两个 `.class` 文件,分别是 `A.class` 、 `A$B.class`

一个类可以有多个内部类，内部类可以实现接口和继承类。内部类可以解决 `java` 类单继承问题，因为内部类和外部类可以相互访问。

数据访问

外部类与内部类可以相互访问类中的成员，包括私有域的

```
1 public class A {  
2     public String name;  
3  
4     private int age;  
5  
6     public class B{//内部类  
7         public String bName;  
8  
9         private int bAge;  
10  
11        private void setAge() {  
12            name = "";  
13            this.bAge = age;  
14        }  
15  
16    }  
17  
18    public class C{  
19        public String cName;
```

```

20
21     private int cAge;
22 }
23
24 public void compareAge() {
25     B b = new B();
26     this.age = b.bAge;
27     b.setAge();
28     C c = new C();
29     c.cAge = 10;
30 }
31
32
33 public static void main(String[] args) {
34
35 }
36 }

```

内部类的实例化，内部类依托于外部类，所以内部类的实例化要先实例化外部类，然后在实例化内部类。

```

1  import com.itlaobing.demo.A.B; //导包
2
3  public class ATest {
4
5      public static void main(String[] args) {
6          A a = new A();
7          B b = a.new B(); //实例化内部类 B b = new A().new B();
8          b.bName = "";
9
10     }
11
12 }

```

静态内部类

```

1  public class AA {
2
3      static class BB{ //`static` 内部类
4
5      }
6
7  }

```

静态内部类不能直接访问外部类的成员，可以直接访问外部内的静态成员。

```
1 import com.itlaobing.demo.AA.BB;//导包
2
3 public class AATest {
4
5     public static void main(String[] args) {
6         AA a = new AA();
7
8         BB b = new BB();//实例化内部类
9     }
10
11 }
```

所有的访问修饰符都在修饰内部类

内部类和外部类区别?

访问修饰符： 外部类只能有两种访问权限： `public` 以及 `package-access`

内部类可以有四种访问权限： `public` / `protected` / `package-access` / `private`

`static` 修饰： 外部类不可以使用 `static` 修饰类，内部类可以

使用： 外部类可以直接被使用，内部类的使用则依赖于外部类

匿名类是没有明确名称的类。

```
1 public static void main(String[] args) {
2     Pet pet = new Dog();
3     pet.cry();
4
5     //匿名类
6     Pet p = new Pet() {
7         @Override
8         public void cry() {
9             System.out.println("喵喵喵");
10        }
11    };
12    p.cry();
13 }
14 }
```

匿名类不能使用修饰符，在方法的内部，对外隐藏。不能是 `static`，隐式 `final` 类。匿名类没有类名，没有构造方法。

~~匿名类不能定义接口，不能定义静态块，静态方法，静态变量~~

匿名类可以访问类中的成员，不能访问方法中的变量，除非是 `final` 的。

匿名类一般是实现接口、继承抽象类。相当于实现、继承了这个接口/类。

```
1 public abstract class Pet {
2
3     public abstract void cry();
4 }
5
6 //匿名类
7 Pet p = new Pet() {
8     @Override
9     public void cry() {
10         System.out.println("喵喵喵");
11     }
12 };
13 //等同于
14 public class 类名 extends Pet{
15     @Override
16     public void cry() {
17         System.out.println("喵喵喵");
18     }
19 }
20 }
```

