## From bilibili kuangshen

Lec\_1: Use blog to keep track of the study

Lec\_2: Intro

Hadoop (Big data)

Andriod

JavaSE (Disktop)

JavaME (Mobile phone)

JavaEE (Enterprise)

JDK (Java Development Kit)

JRE (Java Runtime Environment)

JVM (Java Virtual Machine)

Uninstall (Delete install home, Java\_home, Java\_path)

Lec\_3:

Hello word:

Public class Hello{

Public static void main (String[] args) {

System.out.print (“Hello, World!”);

}

}

Note that:

class name = file name

Java is capital sensitive

Gramma:

Sentence is end by ;

// This is the note /\*\*/

Java is both Compile (C language) and Interpretive (Websites)

Lec\_4: Identifier

Can only start with Letters, $, or \_

Java is strong decretive language

Two types of data:

1. Primitive type:

1, Value

1,1Int

Byte

Short (2B)

Int (4B)

Long (8B)

1,2Float

Float (4B)

Double (8B)

1,3Character

2, Boolean (T &F)

1. Reference Type (Object,

Lec\_25: Type transfer

Byte, short, char, 🡪 int 🡪 long 🡪 float 🡪 double

Componsory exchange 🡪 may overflow

Lec\_26: The variables

Type varName [=value] [{, varName[=value]}];

Every var(including name, type and domain) has type, each is a sentence.

Domain



Constant – Use final

Some rules:

Class\_member\_name: monthSalary

Constant: MAX\_VALUE

Method: use

P27~

Computation marks:

+, -, \*, /, %, ++, --,

=

See Demo\_operator

Lec\_33: Scanner

To receive the input of the users

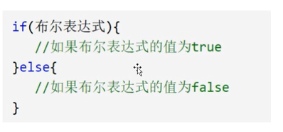
Next(): use tab to return

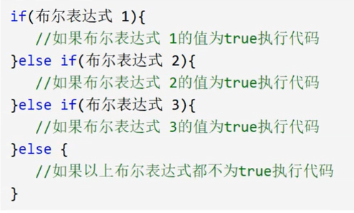
NextLine(): Use enter to return

Lec35: Structure

NO\_1: The sequence order

Mark: P35

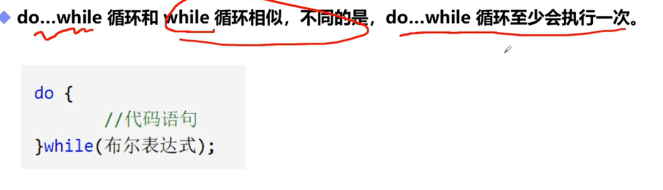


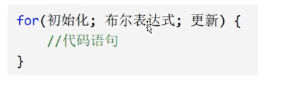


Lec\_37: Switch

How to find the raw code

Lec\_38.39 while:

  
Lec\_40:



Lec\_43: Break, Continue, Goto

Break: could be used to jump out of a loop

Continue: to skip one of the loop

Lec\_46: method2

Lec\_47: Reload



Lec\_50: Recursion

Useable when the number is small

Lec\_51: Truple

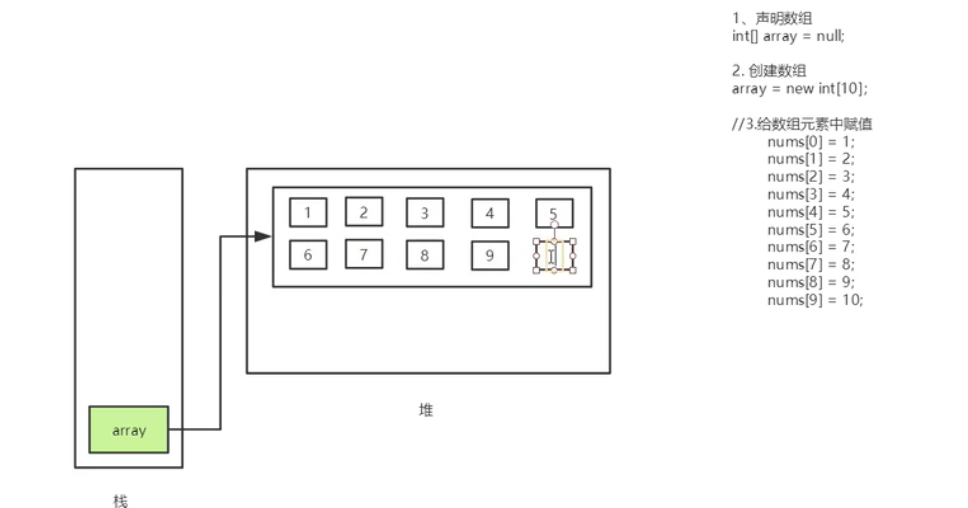
An ordered data with a same type

Can not be changed, could be viewed as an object



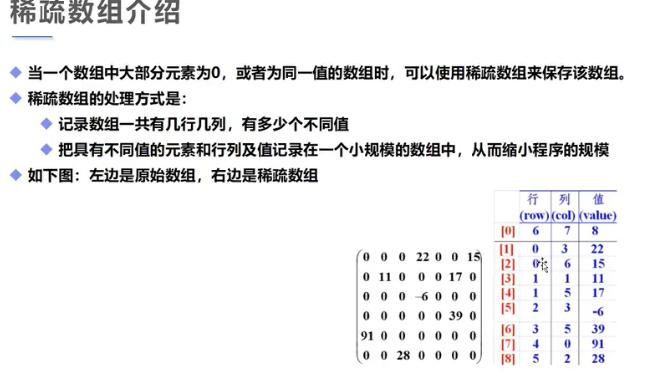
Lec\_52: Memory

Lec\_56

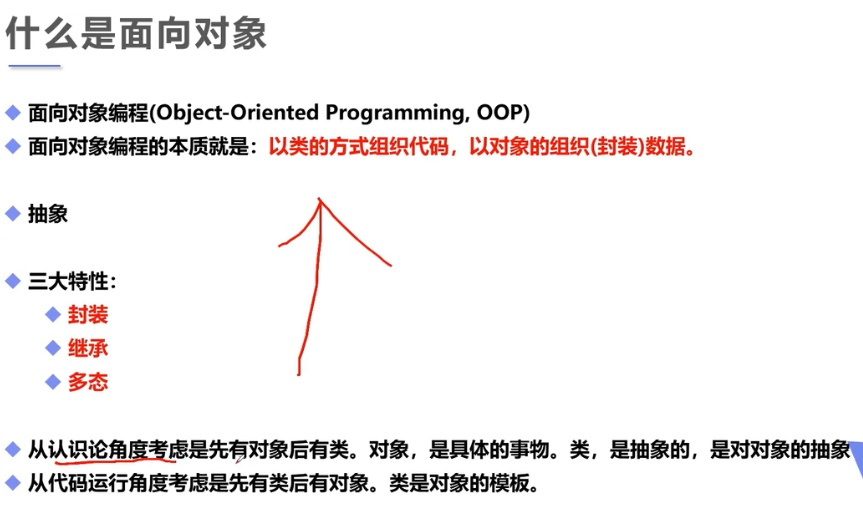


Lec\_57: 2-dim array

Lec\_59: Sparse table

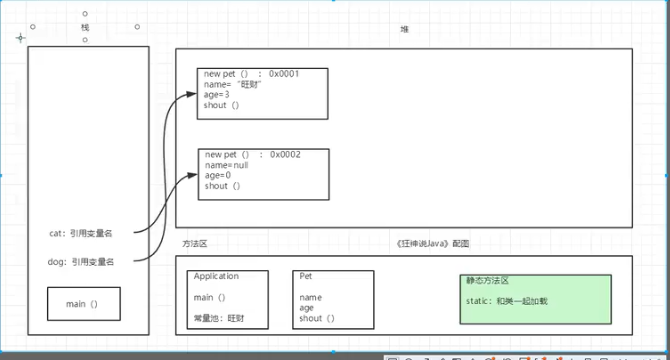


Lec\_60: OOP



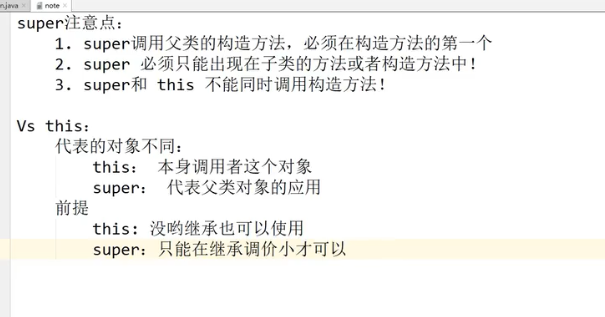
Lec\_64;: Constructor





Lec\_67: Encapsture

Lec\_69: Super

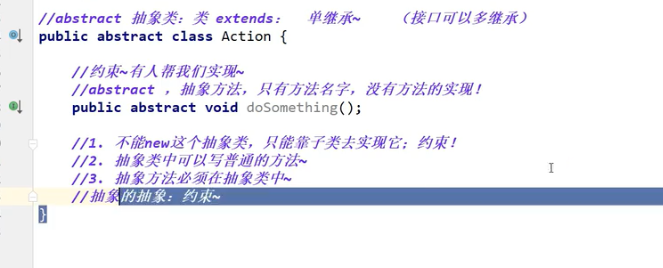


Lec\_70: Override

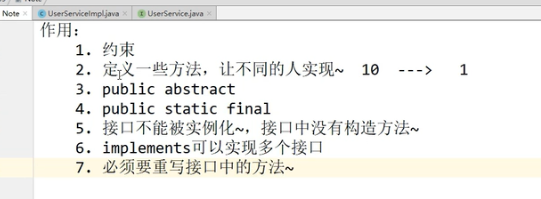


Lec\_71: Ploymorphic

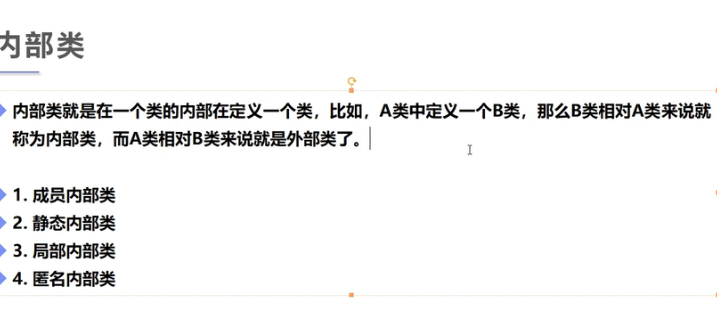
Lec\_74: Abstract



Lec\_75: API?

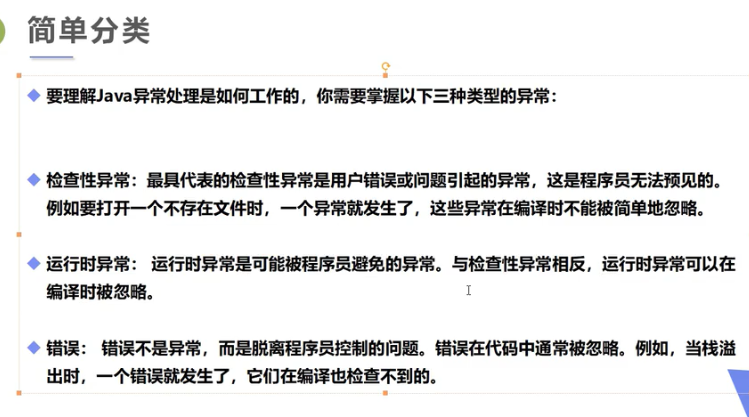


Lec\_76: Internal class

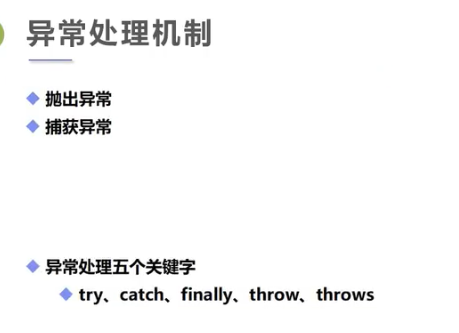


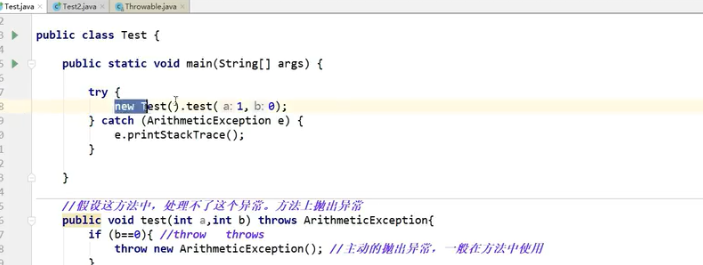
Lec\_77: Exception





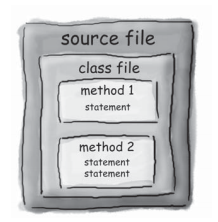
Lec\_78:





## From the book First Head Java

The structure:



**CH\_1: Concepts**

**CH\_2: Class and object**

**CH\_3: Primitive variables**

**CH\_4: Instance (Knows) and method(does)**

**CH\_5: Build a small game using method**

**CH\_6: Using Java functions**

**CH\_7: Inherit P107**

**CH\_8: Polymorphic**

**CH\_9: Object**

CH\_1: Concepts

CH\_2: Class and object

CH\_3: Primitive variables

CH\_4: Instance (Knows) and method(does)

CH\_5: Build a small game using method

CH\_6: Using Java functions

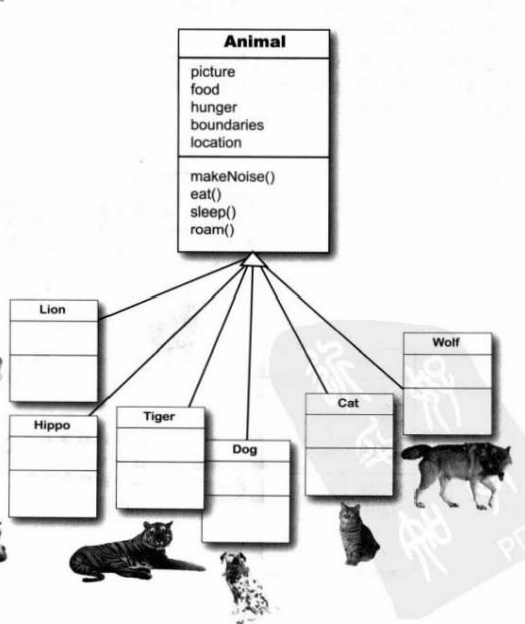
CH\_7: Inherit P107

Is-a

Has-a (extends)

Private 🡪 default 🡪 protected 🡪 public

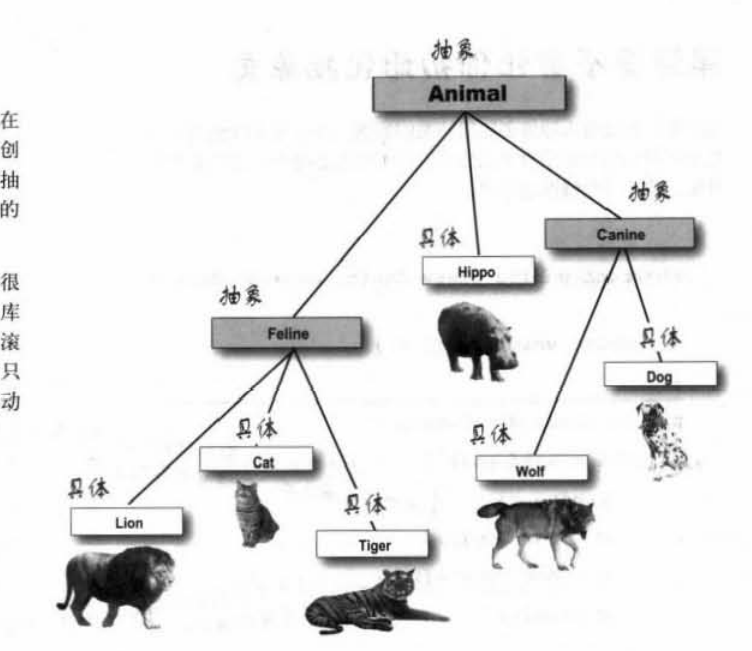
CH\_8: Polymorphic

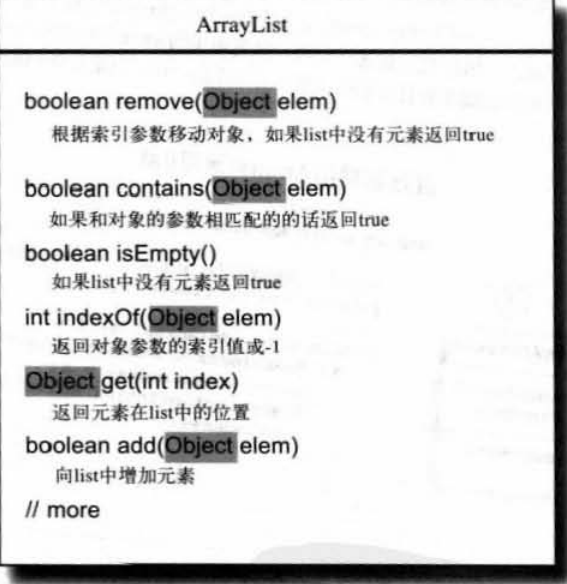


CH\_8: Polymorphic

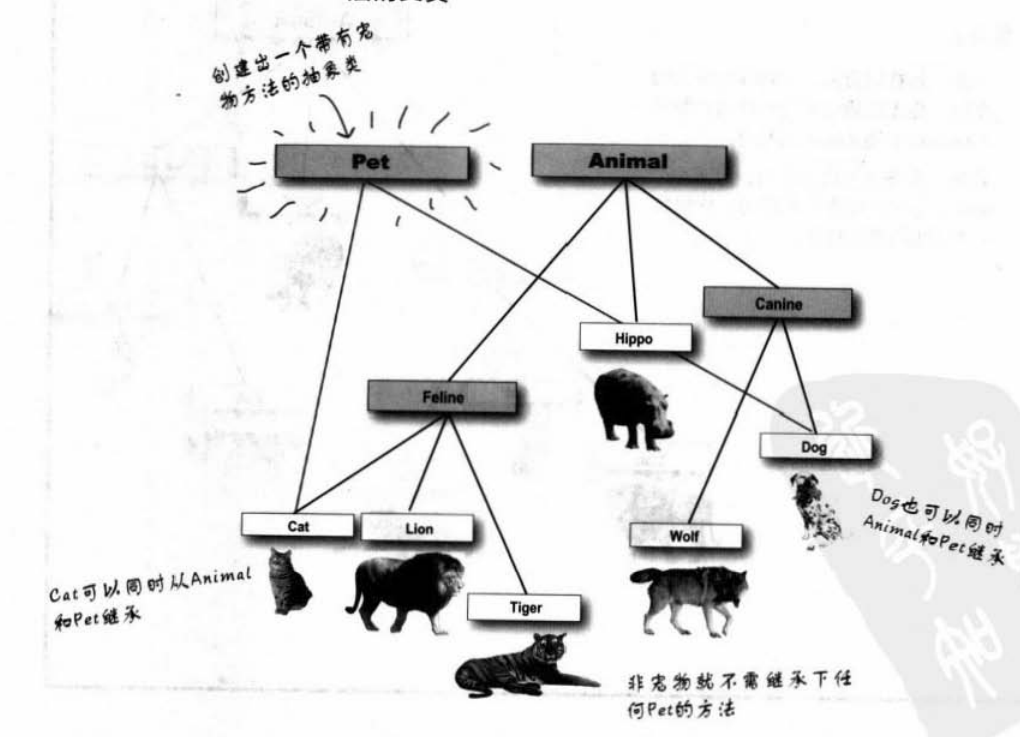


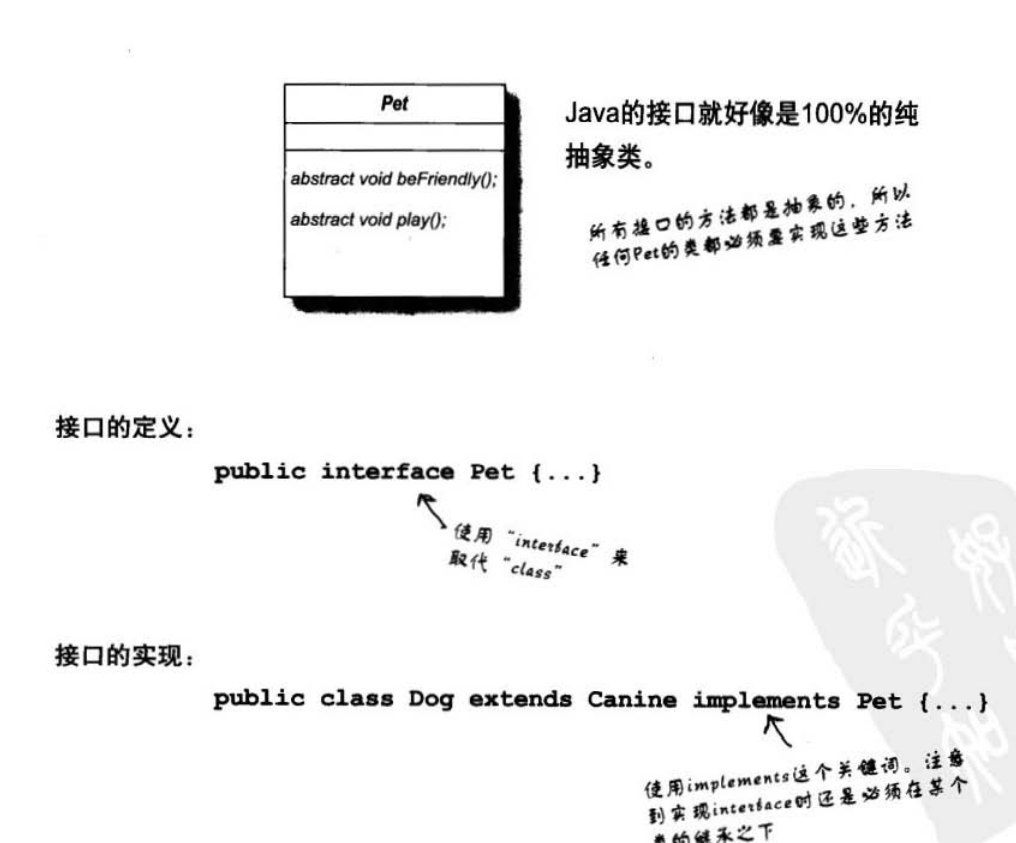
Abstract













**P237**

**Lec\_9: Object**

