

1. java 特性.

封装性. 继承性 抽象性 多态性
Simple 面向对象的. 分布式的. 解释性的. 强类型的.
可移植性强 多线程.

2. Java. development environment.

normally go through edit
compile load verify execute.
校验.

3. Java SDK.

micro edition
J2SE J2ME J2EE.
standard edition portable devices enterprise edition

4. java. Command Line. javac 先编译.class再执行.

java虚拟机实例执行
main

5. main方法. java虚拟机. 类加载器加载类 → 成员变量 → static → main函数 → 卸载类

6. Scanner System.out.println, print

Scanner input = Scanner(System.in) input.nextInt

7. 基本数据类型. boolean byte char short int long float double.

标识符. 字母, 数字, 下划线, \$, 数字不能开头

8. Command line 用 standard stream Console Input Stream Reader System Console

9. 引用数据类型. 类 数组. 枚举类.

基本类型分配在栈上, 引用类型分配在堆上

先分配在栈中一块内存, 对象类型指向真正内容的指针.

String a, b. == equals.
↓
比较变量地址.

10. 垃圾回收. finalize方法 System.gc()



member access modifiers 成员修饰符 instance variable 实例变量 类的实例 可以
 public 可用在包和子类同包里。 局部变量。 人为初始化。 自动初始化
 private. data hiding private 实例变量。 Superclass methods 只能在类中定义的函数访问。
 protected. 同一个包内可使用 可以从superclass的方法获取(可以通过类定义的) 子类可用。
 default. 只可以在包中和同包的地方访问。 同一个package中用
 (什么都不写)

4. if else. ? :

5. for switch

switch () JDK 12
 case : switch () {
 case a, b, c → system.out.println ()

JDK 13
 System.out.println ()
 switch (k) {
 case 1 → "one"
 }
)

& 与 按位或 或 非 异或。
 按位

6. final finally finalize
 ↓ 异常处理。 垃圾回收时。
 属性不可变 语言的一部分
 方法不可覆盖 永不执行。
 类不可继承。

7. static. { 方法 类方法。 只能调用其他静态方法。
 类。 内部类 静态内部类, 只能访问静态成员变量和方法。
 变量。 被一个类中对象共享。

8. 参数提升。



扫描全能王 创建

1. composition has-a.
在类中成员有另一个类的对象.

2. enum type

3. enum static + final

使用 public enum Book

```
{ C( " ", 1 )
```

```
private final String;
```

```
private final int;
```

EnumSet, range ()

4. public static 变量.

private static 只能被类中的方法使用.

5. superclass 已经有子类

subclass 准备 new 的类

java 单继承 extends

用 super() 调用父类构造函数. super.

6. 构造函数在. 继承中. 用 super ()

7. Object Class

equals 方法 ==

== 表示变量地址. equals() 包含比较内容.
(字符串)

8. 静态及如何使用. 动态绑定. 在执行阶段. 实际对象决定. 重载的函数
(到底用哪个)
1个基类 其他类继承. abstract class 虚基类

9. interface 接口 implements. 不相关的类 继承 行为的抽象.

concrete class \longleftrightarrow 能被实例化的类

10. instanceof 决定一个对象的类型.



1. throws 在函数参数后面. try catch finally

unchecked exception runtime 及其子继承. \therefore index Out of Bounds.

checked exception ^{Exception 类}非 runtime exception 不解决无法通过编译.
(\therefore Exception)

2. ~~java~~ event handlers

~~throw~~

catch. Catch 异常. 并对异常处理异常. finally 不管有无异常都执行

try. 写可能会报错的代码.

throw 和 throws. 方法可能抛出异常的声明. void int A throws Exception

↓
语句抛出异常 throw e;

2. String immutable String s1 = "hello"
里面的 character 不能改变 String s2 = "hello"
if (s1 == s2) true.

String.length.

.charAt(i)

.getChars()

String.equals equalsIgnoreCase compareTo regionMatches

startsWith endsWith

indexOf lastIndexOf

两件比较子串.
s3.startsWith(s4, 0, s4.length())

.substring() concat()

String.replace String.trim() 消除开头或结尾的空格

String.valueOf 类型转换 String x2 = x1.valueOf()

StringBuilder mutable 可以扩容.

StringBuilder x = new StringBuilder("hello")

length

capacity

ensureCapacity 确保有足够的容量.
setLength 根据 set 的值 { < 缩小
> null

同层的符号.

reverse

append

insert delete. split



1. static import 引入静态成员。

2. 对 main 方法 main 是 static

JVM 调用 main 方法 在这之前类的对象并未创建，
static 调用 main 不需要创建一个实例。

3. 参数传递。—— 传引用 Java 采用传值。

4. static 方法只能调用 同类中的其他 static 方法，
只能操作 类中的静态成员 (变量)

5. 参数提升。
float → double
int → float, double, long
double 不能提升

6. Random SecureRandom
可被预测

Random SecureRandom x = new SecureRandom();
x = new Random(x.nextInt());

7. Overloading 重载 有相同名但参数不同
不能通过 return 区分。

8. 初始化数组 int[] c = new int[10]; 多维: int[][] b = {{1}, {2}}
int[] a, b, c 均为数组。 int[][] b = new int[2][2]
b[0][0] = new int[5]

9. enhanced for for (int number : array)

10. 方法传递 arrays 参数是 array 则传引用
是值 传值。

11. 可变长参数列表。只能出现一次，参数列表

12. 封装与数据隐藏 限制对属性访问 private

13. this 自己的指针。static 不行。

可用 this 调用类中的其他方法或变量。

可用 this 作构造函数内容。

