回顾：

1. Scanner:

1) import java.util.Scanner;

2) Scanner scan = new Scanner(System.in);

3) System.out.println(“请输入年龄”);

int age = scan.nextInt();

double price = scan.nextDouble();

1. 分支结构: 基于条件执行某语句

1) if: 1条路

2) if..else: 2条路

3) if..else if: 多条路

4) switch..case: 多条路

优点: 效率高、结构清晰

缺点: 整数、相等

break: 跳出switch

正课：

1. 循环: 反复多次执行一段相同或相似的代码
2. 循环三要素：

1) 循环变量的初始化

2) 循环的条件(以循环变量为基础)

3) 循环变量的改变(向着循环的结束变)

循环变量: 在整个循环过程中所反复改变的那个数

1. 循环结构：

1) while结构：先判断后执行，有可能一次都不执行

2) do..while结构：先执行后判断，至少执行一次

要素1与要素3相同时，首选do..while

3) for结构：应用率最高，与次数相关的首选for

1. break: 跳出循环

练习:------------------------------------每人两次

1. 9的乘法表-----for版(正序、倒序、奇数)
2. 完成猜数字游戏----do…while版

7点晚课开始

//. 1 2 3

for( int times=0 ; times<10 ; times++ ) {

System.out.println(“武汉加油，中国加油”); 4

}

12432432432432432432432432

12432432432432432432432432

int times=0; //1

while(times<10){. //2

System.out.println(“武汉加油，中国加油”); //4

times++; //3

}

听得懂，但是写不出来-------------正常的

代码不是看懂的，一定是写懂的

循环的结束条件:

1. 条件为false
2. 碰到break

while( boolean ){

语句块

}

do{

语句块

} while(boolean);

1到1000之内

Math.random()-------------------0.0到0.99999999999999…

\*1000---------------------------0.0到999.999999999999…

+1------------------------------1.0到1000.99999999999…

(int)-----------------------------1到1000

(int)-----------------------------0到999

+1------------------------------1到1000

int num = 4;

int num = scan.nextInt();

int num = ?; //系统随机生成的

int num = 250; //手里藏着的数

循环变量: 用户所猜的那个数guess

1) Systen.out.println(“猜吧!”);

int guess = scan.nextInt();

2) guess != num

3) System.out.println(“猜吧!”);

guess = scan.nextInt();

猜吧!

300

太大了

猜吧!

200

太小了

猜吧！

251

太大了

猜吧！

250

恭喜你，猜对了

案例一：

输出10次“武汉加油，中国加油”

武汉加油，中国加油

武汉加油，中国加油

武汉加油，中国加油

武汉加油，中国加油

武汉加油，中国加油

武汉加油，中国加油

武汉加油，中国加油

武汉加油，中国加油

武汉加油，中国加油

武汉加油，中国加油

循环变量: 输出的次数times

1) int times=0;

2) times<10

3) times++;

times=0/1/2/3/4/5/6/7/8/9/ 10时结束

案例二：

输出9的乘法表：

1\*9=9

2\*9=18

3\*9=27

4\*9=36

5\*9=45

6\*9=54

7\*9=63

8\*9=72

9\*9=81

循环变量: 乘数num //9,8,7,6,5,4,3,2,1

1) int num=9;

2) num>=1

3) num--;

num=9/8/7/6/5/4/3/2/1/ 0时结束

1) int num=1;

2) num<=9

3) num++; num+=2;

num=1/2/3/4/5/6/7/8/9/ 10时结束

4) 循环变量的初始化

5) 循环的条件(以循环变量为基础)

6) 循环变量的改变(向着循环的结束变)

循环变量: 在整个循环过程中所反复改变的那个数

循环变量: 所打份数num

1) int num=0;

2) num<5

3) num++;

num=0/1/2/3/4/ 5时结束

份数为0

够5份吗？ 不够 打印一份 份数为1

够5份吗？ 不够 打印一份 份数为2

够5份吗？ 不够 打印一份 份数为3

够5份吗？ 不够 打印一份 份数为4

够5份吗？ 不够 打印一份 份数为5

够5份吗？ 够了

循环变量: 所跑的圈数count

1) int count=0;

2) count<3

3) count++;

count=0/1/2/ 3时结束

圈数为0

够3圈吗？ 不够 跑一圈 圈数为1

够3圈吗？ 不够 跑一圈 圈数为2

够3圈吗？ 不够 跑一圈 圈数为3

够3圈吗？ 够了

1) 顺序结构：从上往下逐行执行，每句必走

2) 分支结构：有条件的执行某语句一次，并非每句必走

3) 循环结构：有条件的执行某语句多次，并非每句必走

if( boolean ){ //判断一次

语句块 //执行一次

}

while( boolean ){ //判断多次

语句块. //执行多次

}