**数据类型和运算符作业**

1. **填空题**
2. Java语言规定标识符由字母、下划线、美元符号和数字组成，并且第一个字符不能是 **数字** 。
3. Java中整型变量有byte、short、int和long四种，不同类型的整数变量在内存中分配的字节数不同，数值范围也不同。对于int型变量，内存分配 **4** 个字节。
4. 在Java中浮点型变量有float和double两种，对于float型变量，内存分配4个字节，尾数可精确到7位有效数字，对于double型变量，内存分配 **8** 个字节。
5. char c=’a’;System.out.println(c+1);运行结果为: **98**  。
6. \_\_\_**&&**\_\_\_\_\_\_是短路与运算符，如果左侧表达式的计算结果是false，右侧表达式将不再进行计算。
7. 下面的语句是声明一个变量并赋值：boolean b1=5!=8； b1的值是\_\_\_**true**\_\_\_\_\_\_\_。
8. 使用位运算符来实现运算效率最高，所以最有效率的方法算出2乘以8等于多少的语句是 **2<<3** 。
9. 基本数据类型的类型转换中，要将double类型的常量3.14159赋给为整数类型变量n的语句是\_\_\_**int n = (int)3.14159;**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_。
10. 八进制整数012表示十进制的\_\_**10**\_，十六进制数0x3C表示十进制的\_\_**60**\_\_\_。
11. 一个十进制整数转换成八进制数后是1234，则它转为十六进制数后是  **29C** 。
12. **选择题**

|  |  |  |
| --- | --- | --- |
| **1.** | **在Java中，以下错误的变量名是（ D ）。（选择一项）** | |
|  |  |  |
|  | **A** | constant |
|  | **B.** | flag |
|  | **C.** | a\_b |
|  | **D.** | final |

|  |  |  |
| --- | --- | --- |
| **2.** | **以下选项中属于合法的Java标识符的是（ CD ）。（选择两项）** | |
|  |  |  |
|  | **A.** | public |
|  | **B.** | 3num |
|  | **C.** | name |
|  | **D.** | \_age |

|  |  |  |
| --- | --- | --- |
| **3.** | **在Java中，byte数据类型的取值范围是（ A ）。（选择一项）** | |
|  |  |  |
|  | **A** | -128 ~ 127 |
|  | **B.** | -228 ~128 |
|  | **C.** | -255 ~ 256 |
|  | **D.** | -255 ~ 255 |

|  |  |  |
| --- | --- | --- |
| **4.** | **下面的代码段中，执行之后i和j的值是（ C ）。（选择一项）** | |
|  | **int** i=1; **int** j;  j=i++; | |
|  |  |  |
|  | **A** | 1，1 |
|  | **B.** | 1，2 |
|  | **C.** | 2，1 |
|  | **D.** | 2，2 |

|  |  |  |
| --- | --- | --- |
| **5.** | **下面Java代码的执行结果是（ A ）。（选择一项）** | |
|  | **public** **class** Test {  **public** **static** **void** main(String args[]) {  System.***out***.println(100 % 3);  System.***out***.println(100%3.0);  }  } | |
|  |  |  |
|  | **A** | 1 1.0 |
|  | **B.** | 1 1 |
|  | **C.** | 1.0 1.0 |
|  | **D.** | 33 33.3 |

|  |  |  |
| --- | --- | --- |
| **6.** | 下面的赋值语句中错误的是（ **A** ）**。（选择一项）** | |
|  |  |  |
|  | **A** | **float** f = 11.1; |
|  | **B.** | **double** d = 5.3E12; |
|  | **C.** | **double** d = 3.14159; |
|  | **D.** | **double** d = 3.14D; |

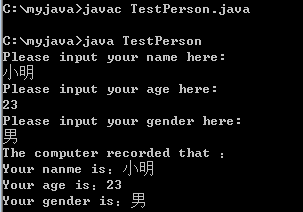
|  |  |  |
| --- | --- | --- |
| **7.** | **在Java中，下面（ AB ）语句能正确通过编译。（选择两项）** | |
|  |  |  |
|  | **A** | System.*out*.println(1+1); |
|  | **B.** | **char** i =2+'2';  System.*out*.println(i); |
|  | **C.** | String s="on"+'one'; |
|  | **D.** | int b=255.0; |

|  |  |  |
| --- | --- | --- |
| **8.** | **以下Java运算符中优先级别最低的两个选项是（ AB ）。（选择二项）** | |
|  |  |  |
|  | **A** | 赋值运算符= |
|  | **B.** | 条件运算符 ?= |
|  | **C.** | 逻辑运算符| |
|  | **D.** | 算术运算符+ |

|  |  |  |
| --- | --- | --- |
| **9.** | **有以下方法的定义，请选择该方法的返回类型（ D ）。（选择一项）** | |
|  | method(**byte** x, **double** y) {  **return** (**short**)x/y\*2;  } | |
|  |  |  |
|  | **A** | byte |
|  | **B.** | short |
|  | **C.** | int |
|  | **D.** | double |

|  |  |  |
| --- | --- | --- |
| **10.** | **关于以下Java程序中错误行的说明正确的是（ B ）。（选择一项）** | |
|  | **public** **class** Test2 {  **public** **static** **void** main(String[] args) {  **short** s1=1; //1  s1=s1+1; //2  s1+=1; //3  System.*out*.println(s1);  }  } | |
|  |  |  |
|  | **A** | 1行错误 |
|  | **B.** | 2行错误 从int类型转到short类型可能会有损耗 |
|  | **C.** | 3行错误 |
|  | **D.** | 1行，2行，3行都错误 |

1. **判断题**
2. Java中变量名不区分大小写，number和NuMbEr代表同一个变量。（ × ）
3. 在编译Java源程序时，计算机根本不会去识别各个变量名的具体含义，因此命名规范对编写Java程序而言不是必要的，而且有可能会降低编写Java源程序的效率。（ × ）
4. Java语言中不同数据类型的长度是固定的，不随机器硬件不同而改变。（ 〇 ）
5. char类型变量可以存储一个Unicode字符，在内存中占2个字节，该字符可以是一个汉字。（ 〇 ）
6. 运算符||和&&既是逻辑运算符，也是位运算符。根据两侧操作数的类型判断是逻辑运算符还是位运算符。（ × ）
7. ^ 是异或位运算符，运算规则是如果两个操作数相同，结果是0，否则结果是1。( 〇 )
8. 赋值运算符优先级别低于条件运算符，条件运算符优先级别低于算术运算符。( 〇 )
9. 赋值和条件运算符是运算级别最低的两种运算符，都具有自右向左的结合性。（ 〇 ）
10. 整型常量的默认类型是int类型，浮点常量的默认类型是float类型。（ × ）
11. 00101010 & 00010111语句的执行结果为00111111（ × ）
12. **简答题**
    1. Java是一种强类型语言，说明Java的数据类型分类。
    2. i++和++i的异同之处
    3. 运算符||和|的异同之处
    4. Java中基本数据类型转换的规则
13. **编码题**
    1. 输入自己的名字，年龄和性别，分别用不同的变量接收，并将输入的信息做输出。



第一种：

import java.util.Scanner;

public class HomeWork{

public static void main(String[] args){

Scanner input = new Scanner(System.in);

System.out.println("Please input your name here:");

String name = input.nextLine();

System.out.println("Please input your age here:");

String age = input.nextLine();

System.out.println("Please input your gender here:");

String gender = input.nextLine();

System.out.println("The computer recorded that:"+

"\nYour name is :"+name+"\nYour age is:"+age+

"\nYour gender is:"+gender);

}

}

第二种（老师提供的）：

import java.util.Scanner;

public class HomeWork{

public static void main(String[] args) {

System.out.println("Please input your name here:");

Scanner input = new Scanner(System.in);

String Name = (String) input.next();

System.out.println("Please input your age here:");

int age = input.nextInt();

System.out.println("Please input your gender here:");

String gender = input.next();

System.out.println("The computer recorded that ：");

System.out.println("Your nanme is：" + Name);

System.out.println("Your age is：" + age);

System.out.println("Your gender is：" + gender);

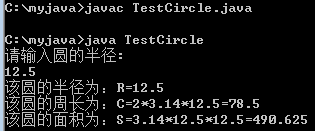
}

}

注意：next()方法读取到空白符就结束l；

nextLine()读取到回车结束(也就是“\r”)

* 1. 输入圆形半径，求圆形的周长和圆形的面积,并将结果输出。



第一种方法：

import java.util.Scanner;

public class HomeWork{

public static void main(String[] args) {

System.out.println("请输入圆的半径:");

Scanner input = new Scanner(System.in);

String r = input.nextLine();

float R = Float.valueOf(r);

System.out.println("该圆的半径为：R=" + R);

System.out.println("该圆的周长为：C=2\*3.14\*" + R+"="+2\*3.14\*R);

System.out.println("该圆的面积为：S=3.14\*" + R+"\*"+R+"="+3.14\*R\*R);

}

}

第二种方法（老师提供的）：

**import** java.util.Scanner;

**public** **class** HomeWork {

**public** **static** **void** main(String[] args) {

// 定义圆周率

**final** **double** PI = 3.14;

Scanner input = **new** Scanner(System.***in***);

// 输入半径

System.***out***.println("请输入圆的半径:");

// 计算周长和面积

**float** r = input.nextFloat();

**double** c = 2 \* PI \* r;

**double** s = PI \* r \* r;

// 输出周长和面积

System.***out***.println("该圆的半径为：R=" + r);

System.***out***.println("该圆的周长为：C=" + 2 + "\*" + PI + "\*" + r + "=" + c);

System.***out***.println("该圆的面积为：S=" + PI + "\*" + r + "\*" + r + "=" + s);

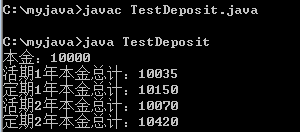
}

}

* 1. 银行利率表如下表所示，请计算存款10000元，活期1年、活期2年，定期1年，定期2年后的本息合计。



结果如下图所示。（结果四舍五入，不保留小数位。使用Math.round(double d)实现）



第一种方法：

import java.util.Scanner;

public class HomeWork{

public static void main(String[] args){

int capital = 10000;

int currentDeposit\_1Year = (int)(Math.round(capital\*(1+0.0035)));

int fixedDeposit\_1year = (int)(Math.round(capital\*(1+0.015)));

int currentDeposit\_2Year = (int)(Math.round(capital\*(1+0.0035\*2)));

int fixedDeposit\_2year = (int)(Math.round(capital\*(1+0.021\*2)));

System.out.println("本金：" + capital);

System.out.println("活期1年本金合计：" + currentDeposit\_1Year );

System.out.println("定期1年本金合计：" + fixedDeposit\_1year );

System.out.println("活期2年本金合计：" + currentDeposit\_2Year);

System.out.println("定期2年本金合计：" + fixedDeposit\_2year);

}

}

第二种方法（老师提供的）：

**public** **class** HomeWork {

**public** **static** **void** main(String[] args) {

//给出本金并输出

**int** money = 10000;

System.***out***.println("本金："+money);

//计算活期1年本金并输出

**double** result1 = money\*(1+0.35/100);

System.***out***.println("活期1年本金总计："+Math.*round*(result1));

//计算定期1年本金并输出

**double** result2 = money\*(1+1.50/100);

System.***out***.println("定期1年本金总计："+Math.*round*(result2));

//计算活期2年本金并输出

**double** result3 = money\*(1+0.35/100\*2);

System.***out***.println("活期2年本金总计："+Math.*round*(result3));

//计算定期2年本金并输出

**double** result4 = money\*(1+2.10/100\*2);

System.***out***.println("定期2年本金总计："+Math.*round*(result4));

}

}

* 1. 某个公司采用公用电话传递数据，数据是四位的整数，在传递过程中是加密的，加密规则如下：每位数字都加上5,然后用和除以10的余数代替该数字，再将第一位和第四位交换，第二位和第三位交换。结果如图所示。



第一种方法：

import java.util.Scanner;

public class HomeWork{

public static void main(String[] args){

Scanner input = new Scanner(System.in);

System.out.print("请输入一个4位正整数：");

int num = input.nextInt();

int a1 = num/1000;

int a2 = (num-a1\*1000)/100;

int a3 = (num-a1\*1000-a2\*100)/10;

int a4 = num-a1\*1000-a2\*100-a3\*10;

a1 = (a1+5)%10;

a2 = (a2+5)%10;

a3 = (a3+5)%10;

a4 = (a4+5)%10;

int b1 = a4;

int b2 = a3;

int b3 = a2;

int b4 = a1;

int code = b1\*1000+b2\*100+b3\*10+b4;

System.out.println("加密后的数字为："+code);

}

}

第二种方法（老师提供的）：

**import** java.util.Scanner;

**public** **class** HomeWork {

**public** **static** **void** main(String[] args) {

//输入一个4位整数

Scanner s = **new** Scanner(System.***in***);

System.***out***.print("请输入一个4位正整数：");

**int** num = s.nextInt();

//获取各个位上的数

**int** bit4 = (num / 1000); // 取千位的数字

**int** bit3 = (num / 100) % 10; // 取百位的数字

**int** bit2 = (num / 10) % 10; // 取十位的数字

**int** bit1 = num % 10; // 取个位的数字

//每位数字加5

bit4 = (bit4+5)%10;

bit3 = (bit3+5)%10;

bit2 = (bit2+5)%10;

bit1 = (bit1+5)%10;

//交换第一位和第四位

**int** temp;

temp = bit4;

bit4 = bit1;

bit1 = temp;

//交换第二位和第三位

temp = bit2;

bit2 = bit3;

bit3 = temp;

// 输出加密后数字

//System.out.println("加密后的数字为："+bit4+bit3+bit2+bit1);

**int** ennum = bit4\*1000+bit3\*100+bit2\*10+bit1;

System.***out***.println("加密后的数字为："+ennum);

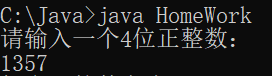
}

}

System.out.print(“请输入一个4位正整数：”)是在当行输入



System.out.println(“请输入一个4位正整数：”)是在下一行输入



1. **可选题**
   1. 网上查询资料，了解如下内容
      1. ASCII、ISO8859-1、GB2312、GBK、GB18030、BIG5、UNICODE、ANSI等字符集的特点
      2. 常用进制（二进制、八进制、十六进制、十进制）的特点及其转换
      3. 源码、反码、补码含义、作用及其转换
   2. 利用”异或”运算的性质,对几个字符进行加密并输出密文,然后再解密。加密算法是：密钥是字符’8’，明文的每个字符和密钥进行异或运算，得到密文。密钥和密文的每个字符再次进行异或运算，重新得到明文。结果如图所示。



第一种方法：

public class HomeWork{

public static void main(String[] args){

char a1 = '十';

char a2 = '点';

char a3 = '进';

char a4 = '攻';

String original = ""+a1+a2+a3+a4;

char b1 = (char) (a1 ^ '8');

char b2 = (char) (a2 ^ '8');

char b3 = (char) (a3 ^ '8');

char b4 = (char) (a4 ^ '8');

String encryption = ""+b1+b2+b3+b4;

char c1 = (char) (b1 ^ '8');

char c2 = (char) (b2 ^ '8');

char c3 = (char) (b3 ^ '8');

char c4 = (char) (b4 ^ '8');

String decrypt = ""+c1+c2+c3+c4;

System.out.println("加密前原文："+original);

System.out.println("密文："+encryption);

System.out.println("解密后原文："+decrypt);

}

}

第二种方法（老师提供的）：

**public** **class** HomeWork {

**public** **static** **void** main(String args[]) {

//原文

**char** a1 = '十', a2 = '点', a3 = '进', a4 = '攻';

System.***out***.println("加密前原文:" + a1 + a2 + a3 + a4);

//加密

**char** secret = '8';

a1 = (**char**) (a1 ^ secret);

a2 = (**char**) (a2 ^ secret);

a3 = (**char**) (a3 ^ secret);

a4 = (**char**) (a4 ^ secret);

System.***out***.println("密文:" + a1 + a2 + a3 + a4);

//解密

a1 = (**char**) (a1 ^ secret);

a2 = (**char**) (a2 ^ secret);

a3 = (**char**) (a3 ^ secret);

a4 = (**char**) (a4 ^ secret);

System.***out***.println("解密后原文:" + a1 + a2 + a3 + a4);

}

}