

在Java中产生随机数的两个方法Java认证考试 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/629/2021_2022__E5_9C_A&Java_E4_B8_AD_c104_629762.htm

一、利用random方法来生成随机数。在Java语言中生成随机数相对来说比较简单，因为有一个现成的方法可以使用。在Math类中，Java语言提供了一个叫做random的方法。通过这个方法可以让系统产生随机数。不过默认情况下，其产生的随机数范围比较小，为大于等于0到小于1的double型随机数。虽然其随机数产生的范围比较小，不能够满足日常的需求。如日常工作中可能需要产生整数的随机数。其实，只要对这个方法进行一些灵活的处理，就可以获取任意范围的随机数。如我们可以先通过random方法生成一个随机数，然后将结果乘以10。此时产生的随机数字即为大于等于0小于10的数字。然后再利用Int方法进行转换(它会去掉小数掉后面的数字，即只获取整数部分，不是四舍五入)。最后即可获取一个0到9的整数型随机数字。其实现方法很简单，就是对原有的random方法按照如下的格式进行变型：`(int)(Math.Random()*10)`即可。其实我们还可以对这个方法进行扩展，让其产生任意范围内的随机数。至需要将这个10换成n即可，如改为`(int)(Math.Random()*n)`。此时应用程序就会产生一个大于等于0小与n之间的随机数。如将n设置为5，那么其就会产生一个0到5之间的整数型的随机数。如果将这个写成一个带参数的方法，那么只要用户输入需要生成随机数的最大值，就可以让这个方法来生成制定范围的随机数。在Java中定义自己的工具库 有时候程序员可能需要生成一个指定范围内的随机偶数或者奇数。此时是否可

以通过这个方法来实现呢?答案是肯定的。如现在程序要需要生成一个1-100范围内的偶数。此时该如何实现?首先,需要生成一个0到99之内的随机数(至于这里为什么是99,大家耐心看下去就知道原因了)。要实现这个需求,很简单吧,只要通过如下语句就可以实现: `i=1+(int)(Math.Random()*100)`。其中`(int)(Math.Random()*99)`产生0到99的整数型随机数。然后再加上1就是产生1到100之间的随机整数。然后将产生的随机数赋值给变量i。但是此时其产生的随机数既有偶数,又有奇数。而现在程序员需要的是一个随机的偶数。那么我们可以在后面加上一个if判断语句。将这个随机数除以2,如果没有余数的话(或者余数为0)则表明这个随机数是偶数,直接返回即可。如果其返回的余数不为零,那么就表明其是奇数,我们只要加上1就变为了偶数,返回即可。注意,在上面的随机数生成中,笔者采用的范围是0到99,然后再加上1让其变为1到100的随机数。最后的结果就是生成1到100之间的随机偶数。其实,如果要范围随机奇数的话,至需要对上面的语句进行稍微的修改即可。Java: 改变你我的世界 假设现在用户想生成一个任意范围内的奇数或者偶数,能够实现吗?假设现在用户想实现一个m到n之间的任意偶数(其中m<n)。可见虽然random方法其自身产生的随机数有比较严格的范围限制。但是只要对其进行合理的转换,程序员仍然可以采用这个方法产生用户所需要的随机数据。

二、通过Random类来生成随机数。

在Java语言中,除了可以通过random方法来产生随机数之外,还可以通过一个Random类来产生随机数。程序开发人员可以通过实例化一个Random对象来创建一个随机数的生成器。如 `Random r=new Random()`。通过这条语句就利用

了Random类创建了一个随机数的生成器。不过以这种方法创建随机数时，与采用Random方法产生随机数的机制不同。利用现在这种方式实例化对象时，Java编译器会以系统当前的时间作为随机数生成器的种子。由于时间时时刻刻在变化的。若以这个时间作为生成器的种子，就可以保证生成的随机数真的是随机的，其生成的随机数重复率会大大的降低。利用这种方法其比较方便。如可以利用提供的关键字，让程序返回一个随机的整数(采用int nextInt(10))等等。不过其返回控制要比Random方法困难一点。如现在需要系统提供一个10到50之间的随机奇数，利用这个Random类就无法完成。也就是说，利用这个Random类来生成随机数，其只能够控制上限，而不能控制下限。换一句话说，其可以指定最大的随机数范围，而不能指定最小的随机数范围。所以，在灵活性上，其比Random方法要稍微差一点。另外利用这个方法来实现的话，必须先创建一个对象。也就是说利用Random类来创建对象。这跟Random方法不同。像上面举的例子中，Random方法本身就是一个Math类中方法，可以直接调用，省去对象创建的方法。为此笔者建议各位读者与程序开发人员，最好还是使用Random方法来创建随机数。只有在生成一些比较特殊的随机数时采用Random类。如现在需要生成一个概率密度为高斯分布的双精度值随机数时，则通过采用Random类的方法来创建随机数相对来说比较简单一点。

三、产生随机的字符。

上面介绍的两种方法，产生的都是随机的数值型数据。但是有时候用户可能还需要产生随机的字符。其实也可以利用random方法来产生随机字符。如可以利用代码生成一个随机的小写字符：`(char)(' a ' + Math.random()*(' z ' - ' a ' + 1))`

。其实这跟生成任意两个数之间的随机数类似。通过以上的代码就可以生成一个范围内的任意随机字符。通过对这个代码进行适当的修整，还可以生成任意两个字符之间的随机字符与任意大写字符的随机字符。其转换的方式跟上面提到的任意范围内的随机数类似。各位读者若感兴趣的话，可以自己进行测试一下。师傅领进门，修行在自身。如果笔者在这里一股脑儿将所有的答案告诉大家，大家的印象不会很深。大家若回去自己动手试试看，反而更容易记住。笔者在这里给大家一个提示，java认证网，加入收藏只需要根据 `m` `(int)(Math.Random()*(n-m))` 这条语句来调整 `(char)('a' + Math.random()*('z' - 'a' + 1))` 这个代码即可。最后笔者需要强调的一点就是在用户传入参数的时候，最好要对用户传入的参数进行合法性检查。以免用户传入的参数不符合既定的要求，如参数需要的是数值型的数据而传入的却是字符型的数据，而导致程序运行出错。100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com