

(یہ) خدا کا وعدہ (ہے) خدا اپنے وعدے کے خلاف نہیں کرتا لیکن اکثر لوگ نہیں جانتے۔ یہ تو دنیا کی ظاہری زندگی کو جانتے ہیں۔ اور آخرت (کی طرف) سے غافل ہیں۔ الروم آیت ۶-۵

Get mad, then get over it. Colin Powell

Lecture 12

Break, Continue, Use of Flag in Loop and Intro to Nested Loop

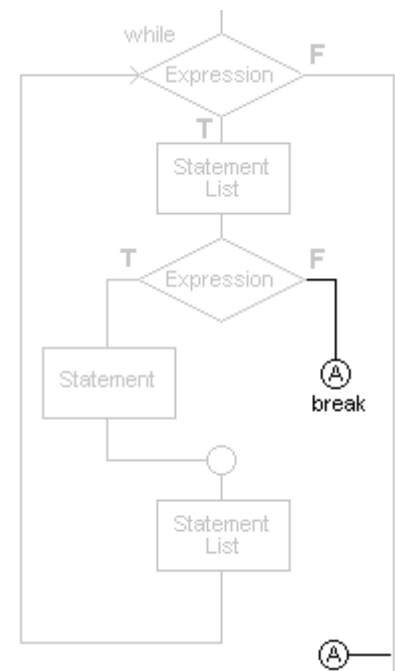
In this lecture we will discuss two key words **break & continue** can be used inside loop, next we will see concept of flag used within loop and finally we will get introduction of nested loops.

Break & Continue

break and **continue** are two keywords provided by many languages to be used within loop, however, both disturb the normal flow of code, therefore, using them is not a good programming practice. Therefore, we will not discuss them in detail. **break** is used to terminate loop where loop condition may hold true value. Typically people used them in nested loop to break inner loop but once again as we can avoid them we should not use them for the reason discuss before.

Once again we will learn how to use break but we will also discuss that using break statement is not a good practice. "break" is a java keyword written typically inside loop to terminate loop. In some scenarios loop condition may hold true but using break terminates loop. Above figure shows the flow of break so you can observe that loop has one entry point and two exit points. One exit point is loop condition other is break statement. To understand break consider code to find number is prime or not. A number is prime if it not divisible by any number other than 2 and itself. Therefore we have to check divisibility by 2 to n-1 (logic can be further improved). A code is:

```
- input n
- for a=2 to n-1
-   if n is divisible by a
-     print "Number is not prime"
-   else
-     print "Number is prime"
```



<http://c.comsci.us/syntax/statement/break.html>

Above code is a typical sample to show where students lack logic. There are two problems in this code, one is if we have a number which is prime code will print Number is prime again and again. Second big problem is that if number is not prime in the earlier output code will say it is prime and after some statements it may start saying it is not prime and again previous statement. Therefore to avoid this problem we can modify this code by using break statement:

```
- input n
- for a=2 to n-1
-   if n is divisible by a
-     print "Number is not prime"
-     break
-   if a equal n
-     print "Number is prime"
```

Now in this code if number is not prime after printing number is not prime loop will terminate because of break statement. However, if loop terminate because of loop condition not because of break statement which essentially means number is prime, variable *a* have a value equal to *n*, therefore, out of loop we are checking whether loop is terminated by break or otherwise. In case of break number is not prime, otherwise it is. Similarly **continue** statement is use to skip next lines in loop. This also disturb flow of code it means a jump in the code. To discourage this statement I am not going to discuss any example.

Use of Flag in Loop

There are certain scenarios where we want to know some result from a loop. Flag is implemented using variable normally boolean type but other types may use. Consider the same code for finding prime number. Using flag we can take print statement out of loop. Here see java code because using flag is very much encouraged:

```
int n=in.nextInt();
boolean isPrime=true;
for (i=2;i<n-1;i++){
    if (n%i==0)//n is divisible by i
        isPrime=false;
if (isPrime)
    System.out.println("Number is prime");
else
    System.out.println("Number is not prime");
```

In this code user can take input and check program for different values of *n*. Here **isPrime** is used as flag variable. Out of loop it has true value and inside if number is divisible then we are assigning false value. The idea is once a number is divisible false value is assigned and for all subsequent values it will not change; whether number is divisible or not. Outside we are checking if (isPrime) because flag is a boolean variable, writing if (isPrime==true) is useless, so we should avoid, however, we may write if (!isPrime) for if (isPrime==false)

Nested Loop

Nested loop means loop inside loop. Before starting discussion on nested loop consider following code:

<pre>for (i=1,j=1;i<=3;j++){ System.out.print(j+" "); if (j==4){ j=0; i=i+1; System.out.println(); } }</pre>	<p>Output: 1 2 3 4 1 2 3 4 1 2 3 4</p>
---	---

This code may seem tricky, but let me try to explain it for you. We have placed loop condition on variable *i*; whereas; we are incrementing variable *j*. However, when *j* reaches to 4 we increment *i* and initialize *j* each time we increment *i*. Also we move to next line using **System.out.println**. Such a code can be implemented through nested loop where code seems to be comparatively simple. See same code using nested loop:

```
for (i=1;i<=3;i++){
    for (j=1,j<=4;j++)
```

```

System.out.print(j+" ");
System.out.println();
}
    
```

Hopefully you will acknowledge this code is simpler than above one. Not only it is easy to construct but also it is more readable. Both outer and inner loop can be understood individually as well as together. For example in outer loop *i* is initialized by 1, incremented by 1 and condition is *i* less than equal 3; whereas; in inner loop *j* is initialized by 1, incremented by 1 and condition is *j* less than equal 4. Therefore inner loop will run 4 times, outer loop will runs 3 times; whereas statement to print *j* is inside both loop, therefore, it will run 12 times and **println** statement is in outer loop only, therefore, it will run only 3 times that's why you see 3 rows and 4 columns in output.

It is not at all necessary that there should present any similarity between inner and outer loop. Both loop may have different initialization values, different conditions and different increment factors. Also outer loop and inner loop both can be counting loops, outer loop can be counting, inner loop can be conditional, inner loop can be counting and outer loop can be conditional and both loop can be conditional. It all depends on problem description and way programmer implementing it. See different examples where code is given on left hand side and output on right hand side:

<pre> for (i=1;i<=5;i++){ for (j=5,j<=9;j=j+2) ? (j+" "); ?ln(); } </pre>	<pre> 5 7 9 5 7 9 5 7 9 5 7 9 5 7 9 </pre>	<pre> for (i=1;i<=3;i++){ do{ j=(int)(Math.random()*10); System.out.print (j+" "); }while (j<6); System.out.println(); } </pre>	<pre> 0 0 8 3 1 0 9 8 </pre>
<p>Outer loop repeat 5 times; whereas; inner loop print 5, 7 and 9</p>		<p>Outer loop repeat 3 times; whereas; inner loop generate random numbers 0 to 9 until value is less than 6</p>	
<pre> sum=0; while (sum<100){ for (i=1,i<=4;i=i+1){ n=(int)(Math.random()*8); ? (n+" "); sum=sum+n; } ?ln(); } ? ("Sum:"+sum); </pre>	<pre> 2 6 6 7 4 4 2 2 5 2 7 3 6 0 7 6 3 3 3 7 2 5 1 3 5 6 1 3 Sum:111 </pre>	<pre> sm=0; Random r=new Random(); while (sm<50){ do{ n=r.nextInt()%10; ? (n+" "); sm=s+Math.abs(n); } (Math.abs(n)<5); ?ln(); } ? ("Sum:"+sm); </pre>	<pre> 1 -6 0 -8 1 0 4 9 2 1 -8 9 7 </pre>
<p>Outer loop repeat till sum is less than 100; whereas inner loop will repeat 4 times, inside inner loop each random number is added to sum</p>		<p>Here inner loop is generating random numbers till a random number generated greater than equal +5 or less than equal -5. In outer loop we are checking sum and repeating loop. We have used a new class Random from java.util</p>	

Don't bother if you unable to understand above codes, there are many things in this world you are unable to understand. Consider some examples starting from description. Suppose we want to print 1 – 9 odd numbers in single row 5 times. For 5 times simply take an outer loop starting from 1 and moving towards 5 with an increment of 1. For 1-9 odd numbers take an inner loop starting from 1 and moving towards 9 with an increment of 2. Finally use print inside inner loop and write println in outer loop.

```

for (i=1;i<=5;i++){
    for (j=1;j<=9;j++)
        System.out.print(j+" ");
    System.out.println();
}

```

1 3 5 7 9
1 3 5 7 9
1 3 5 7 9
1 3 5 7 9

Simple code with output is given. You just need to add 1 line to declare i and j variables. Next consider an example to print a pattern of digits given on right hand side. Here we may run outer loop 6 times. Whereas in inner loop we may start from 1 but each time we have to reduce by 1. We may use 6-i as terminating condition of inner loop because each time i is increment by 1. Finally code is:

```

for (i=0;i<6;i++){
    for (j=1;j<=6-i;j++)
        System.out.print(j+" ");
    System.out.println();
}

```

1 2 3 4 5 6
1 2 3 4 5
1 2 3 4
1 2 3
1 2
1

Once again simple code using nested loop. We will see some complex scenarios of nested loop in next lecture. In the mean time student should practice lab and homework shared. The problem with students is that they don't want to spend time which is indispensable. Second problem is that most of you have developed the habit of rote learning a crush of human mind. It's time to start recover your mind state. To learn programming best way is to spend some considerable time on each problem and if you fail then consult some other person. Better to consult seniors take their email address and share your code with them, instead of delaying by finding them.

Many of you have not configured java tools on their systems and unfortunately other than lab class they are not going to labs. It is a reminder you to spend at least 10 hours in front of computer doing programming every week, once again solid 10 hours in front of computer doing programming, closing every other thing. Allocate your recreation time to use social media on internet. Last but not least seek help of Allah by fulfilling his obligations that is:

- offer prayers 5 times regularly
- observe Islamic etiquettes [correct dressing, speaking truth, avoid bad language, avoid cheating]
- avoid mixing with opposite gender
- respecting your parents, elders and teacher by heart not just by show off
- finally don't cheat with yourself

I am sending you reminders to implement Islam in your lives by sharing different verses, do you think you have a choice to leave Islamic obligations and still remaining Muslim. If you think so kindly consult some sincere and authentic Islamic Scholar.

Many of you do sign on quiz that they are not cheating but still they are cheating, I have written that thinking I am teaching to Muslims.

For further practice see homework 6 coming soon on website InshaAllah.