

for loops



© A+ Computer Science - www.apluscompsci.com

What is a LOOP?

© A+ Computer Science - www.apluscompsci.com

the loop

**Until I can hear the song
Make it louder**



**Loops repeat as long
as something is true**

© A+ Computer Science - www.apluscompsci.com

If statements are just simple decision-making statements.

A condition is checked and something may or may not happen based on the evaluation of that condition.

the loop

**As long as I am hungry
I eat something**



**While I have pretzels
I eat one**

© A+ Computer Science - www.apluscompsci.com

If statements are just simple decision-making statements.

A condition is checked and something may or may not happen based on the evaluation of that condition.

Loop Demonstration

condition

```
loop if ( student is hungry  
         and more pretzels left )  
{  
    student will eat  
    1 pretzel stick  
}
```

© A+ Computer Science - www.apluscompsci.com

Loop Definition

A loop is a tool used to repeat a block of code. As long as the loop condition is true, the block of code associated with the condition is executed.

© A+ Computer Science - www.apluscompsci.com

Loop Demonstration

total = 1

condition

loop if (total < 2)

{

block

of

code

The class will :

- 1. stand up**
- 2. sit down**

total = total + 1

}

total

1

2

Loop Condition

Every loop has a condition that must be true before the loop can run.

total < 2 was the loop condition in the loop demonstration.

© A+ Computer Science - www.apluscompsci.com

The for loop

© A+ Computer Science - www.apluscompsci.com

For Loop Definition

A for loop is a block of code associated with a condition. The block of code will run a set number of times depending on the loop condition and increment/decrement value.



© A+ Computer Science - www.apluscompsci.com

the for loop

```
for(init value; boolean condition placed here; inc/dec)
{
    do something 1;
    do something 2;
}
```



© A+ Computer Science - www.apluscompsci.com

do something 1 and do something 2 will occur if the condition is true.

If the condition is true, do something 1 and do something 2 will occur at least once.

do something 1 and do something 2 will continue to occur as long as the loop condition is true.

for loop

start **stop** **inc/dec**
for(int run=1; run<= 5; run=run+1)
{
 out.println(run);
}

You have to tell the loop where to start, when to stop, and how much to change run.

OUTPUT

1
2
3
4
5

© A+ Computer Science - www.apluscompsci.com

The for loop above starts run at 1. As long as run is less than or equal to 5 (`run<=5`), the loop will continue to run and print out the value of variable run. Run increases by one each iteration.

run begins with the value 1

Iteration 1 – print run(1) `run = 1 + 1`

Iteration 2 – print run(2) `run = 2 + 1`

Iteration 3 – print run(3) `run = 3 + 1`

Iteration 4 – print run(4) `run = 4 + 1`

Iteration 5 – print run(5) `run = 5 + 1`

The loop condition fails when run reaches the value 6 as 6 is not less than or equal to 5.

for loop

Some languages use a < start, stop, step > structure with for loops.

start – starting value of the loop

stop – ending value of the loop

step – amount to change the loop variable

```
for x = start to stop step y - Visual Basic  
do something  
next x
```

© A+ Computer Science - www.apluscompsci.com

Start-Initialization

```
      start-0
for (int run = 1;  //stop-1 ;  //inc-3 )
{
    //code-2
}
```

The start value tells the loop where to start.
run will start with a value of 1.

© A+ Computer Science - www.apluscompsci.com

The start section of the loop happens only once. The start section sets up the loop.

Stop-Condition

```
for ( //start-0; run ≤ 5; //inc-3 )  
{  
    //code-2  
}
```

stop-1

This condition must be true in order for the loop to execute.

Each time through the loop, the condition is evaluated. As long as run is less than or equal to 5, the loop continues.

© A+ Computer Science - www.apluscompsci.com

The stop section sets the terms by which the loop can run. As long as run is less than or equal to 5, the loop will continue to run.

Step-Inc/Dec

```
                                inc-3
for ( //start-0; //stop-1; run=run+1)
{
    //code-2
}
```

The increment/decrement value tells the loop how much of a change to make to run.

© A+ Computer Science - www.apluscompsci.com

The step section of the loop either increases or decreases the value of run. In the example above, run is increased by 1 each iteration.

for loop

start **stop** **inc**
for (int run=1; run<=6; run=run+1)
{
 out.println(run);
}

How many times does this loop run?

OUTPUT

1
2
3
4
5
6

© A+ Computer Science - www.apluscompsci.com

This loop starts run at 1 and increments run by one each iteration. The loop will continue to run as long as run is less than or equal to 6.

The loop will stop when the condition $run \leq 6$ fails.

run begins with the value 1

Iteration 1 – print run(1) $run = 1 + 1$

Iteration 2 – print run(2) $run = 2 + 1$

Iteration 3 – print run(3) $run = 3 + 1$

Iteration 4 – print run(4) $run = 4 + 1$

Iteration 5 – print run(5) $run = 5 + 1$

Iteration 6 – print run(6) $run = 6 + 1$

The loop condition fails when run reaches the value 7 as 7 is not less than or equal to 6.

Open for one.java

© A+ Computer Science - www.apluscompsci.com

for loop 2

```
for(int run=1; run<7; run=run+2)
{
    out.println("loop");
    out.println(run);
}
```

OUTPUT

```
loop
1
loop
3
loop
5
```

© A+ Computer Science - www.apluscompsci.com

This loop starts run at 1 and increments run by two each iteration. The loop will continue to run as long as run is less than 7.

The loop will stop when the condition $run < 7$ fails. The condition will fail when run equals 7.

run begins with the value 1

Iteration 1 – print run(1) $run = 1 + 2$

Iteration 2 – print run(3) $run = 3 + 2$

Iteration 3 – print run(5) $run = 5 + 2$

The loop condition fails when run reaches the value 7 as 7 is not less than 7.

for loop 3

```
for(int run=7; run>2; run=run-2)
{
    out.println("loop");
    out.println(run);
}
```

OUTPUT

```
loop
7
loop
5
loop
3
```

© A+ Computer Science - www.apluscompsci.com

This loop starts run at 7 and decrements run by two each iteration. The loop will continue to run as long as run is greater than 2.

The loop will stop when the condition $run > 2$ fails. The condition will fail when run equals 1.

run begins with the value 7

Iteration 1 – print run(7) $run = 7 - 2$

Iteration 2 – print run(5) $run = 5 - 2$

Iteration 3 – print run(3) $run = 3 - 2$

The loop condition fails when run reaches the value 1 as 1 is not greater than 2.

for loop

```
for(int bin=1; bin<=32; bin=bin*2)
{
    out.println(bin);
}
```

OUTPUT

1
2
4
8
16
32

Why is this
loop a $\log_2 N$
loop?

© A+ Computer Science - www.apluscompsci.com

This loop starts bin at 1 and multiplies bin by 2 each iteration. The loop will continue to run as long as bin is less than or equal to 32.

The loop will stop when the condition $\text{bin} \leq 32$ fails. The condition will fail when bin equals 64.

**Open
forttwo.java**

**Open
forthree.java**

© A+ Computer Science - www.apluscompsci.com

for loop

```
out.println("cs contests are fun!");  
for(int uil=5; uil>=1; uil--)  
{  
    out.print("state-");  
}  
out.println("\nchamps");
```

OUTPUT

```
cs contests are fun!  
state-state-state-state-state-  
champs
```

© A+ Computer Science - www.apluscompsci.com

This loop starts uil at 5 and decrements uil by 1. The loop will continue to run as long as uil is greater than or equal to 1.

The loop will stop when the condition `uil>=1` fails. The condition will fail when uil equals 0.

Open foruill.java

© A+ Computer Science - www.apluscompsci.com

Open
graphicsrunner.java
randomcoloredboxes.java

© A+ Computer Science - www.apluscompsci.com

sum / total

**Many times you will use a loop
to total up a run of values.**

total = total + run;

total is totaling up all values of run.

© A+ Computer Science - www.apluscompsci.com

sum / total

```
int total = 0;
for(int run=1; run<6; run++)
{
    total=total+run;
}
out.println(total);
```

© A+ Computer Science - www.apluscompsci.com

total starts at zero.

For each iteration of the for loop, the current value of run is added to total. Runs values would be 1,2,3,4,5,6

The loop fails when run reaches 6.

Iteration 1 – total = 1 + 1 total is 1

Iteration 2 – total = 2 + 2 total is 3

Iteration 3 – total = 3 + 3 total is 6

Iteration 4 – total = 6 + 4 total is 10

Iteration 5 – total = 10 + 5 total is 15

The loop condition fails when run reaches the value 6 as 6 is not less than 6.

sum / total

```
int total=0;
for(int x=1; x<6; x++)
{
    total=total+x;
}
out.println(total);
```

OUTPUT

15

TRACE

x	total	output
	0	
1	1	
2	3	
3	6	
4	10	
5	15	
6		15

© A+ Computer Science - www.apluscompsci.com

total starts at zero.

For each iteration of the for loop, the current value of run is added to total. Runs values would be 1,2,3,4,5,6

The loop fails when run reaches 6.

Iteration 1 – total = 1 + 1 total is 1

Iteration 2 – total = 2 + 2 total is 3

Iteration 3 – total = 3 + 3 total is 6

Iteration 4 – total = 6 + 4 total is 10

Iteration 5 – total = 10 + 5 total is 15

The loop condition fails when run reaches the value 6 as 6 is not less than 6.

Open fortotal.java

© A+ Computer Science - www.apluscompsci.com

Strings and loops

```
String s = "compsci";  
for(int i=0; i<s.length(); i++)  
{  
    out.println(s.charAt(i));  
}
```

OUTPUT

c
o
m
p
s
c
i

© A+ Computer Science - www.apluscompsci.com

The for loop starts `i` at 0. As long as `i` is less than `s.length()`, the loop continues to run. For each iteration through the loop, `i` is incremented by 1.

The loop will print out each character in the String starting at 0 and going up through the String to `s.length()`.

new for loop

```
String s = "compsci";  
for(char c : s.toCharArray())  
{  
    out.println(c);  
}
```

OUTPUT

c
o
m
p
s
c
i

© A+ Computer Science - www.apluscompsci.com

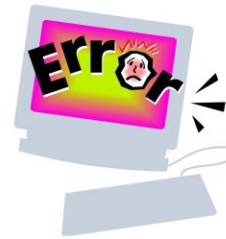
The new for loop iterates over the String starting at character c. Each character from the String is placed into c. Each letter is printed as it is encountered. The loop continues to run as long as there are more letters in the String.

**Open
forstrings.java
newfor.java**

© A+ Computer Science - www.apluscompsci.com

COMMON ERRORS

```
for(int run=0; run>5; run++)  
{  
    //do something  
}
```



© A+ Computer Science - www.apluscompsci.com

The loop condition should always agree with the loops starting value.

COMMON ERRORS

```
For(int run=0; run<5; run++)  
{  
}
```

```
for(int run=0; run<5; run++);  
{  
}
```



© A+ Computer Science - www.apluscompsci.com

For is always all lowercase. Never capitalize the f on for.

NEVER put a semi-colon before an OPEN BRACE.

{ and ; rule

Never put a ;
before an open { brace

:{ **illegal**

}; **legal**



© A+ Computer Science - www.apluscompsci.com

Start work on the labs

© A+ Computer Science - www.apluscompsci.com