Chapter 4

`if`, `if-else`, and `while` statements

`if` statement

```
if (true/false expression)
    statement
```

```
if (count <= 10)
    System.out.println("Too few");
```

Compiler does not care about indentation

```
if (count <= 10)
   System.out.println("Too few");
System.out.println("bye");


if (count <= 10)
   System.out.println("Too few");
   System.out.println("bye");
```

`if-else` statement

```
if (true/false expression)
    statement 1
else
    statement 2
```

```
if (count <= 10)
    System.out.println("Too few");
else
    System.out.println("Enough");
System.out.println("bye");
```

Compound statements

```
if (x <= 70)
{
   System.out.println("incrementing x");
   x++;
}
else
{
   System.out.println("decrementing x");
   x--;
}
System.out.println("bye");
```

Nested `if` statements

```
if (x == 3)
   if (y == 5)
      System.out.println("apple");
   else
      System.out.println("pear");
```

`else` associates with nearest unassociated `if`

Change `if` with which `else` associates

Use braces:

```
if (x == 3)
{
   if (y == 5)
      System.out.println("apple");
}
else
   System.out.println("pear");
```

Use null statement to change association

```
if (x == 3)
   if (y == 5)
      System.out.println("apple");
   else
      ;   // null statement
else
   System.out.println("pear");
```

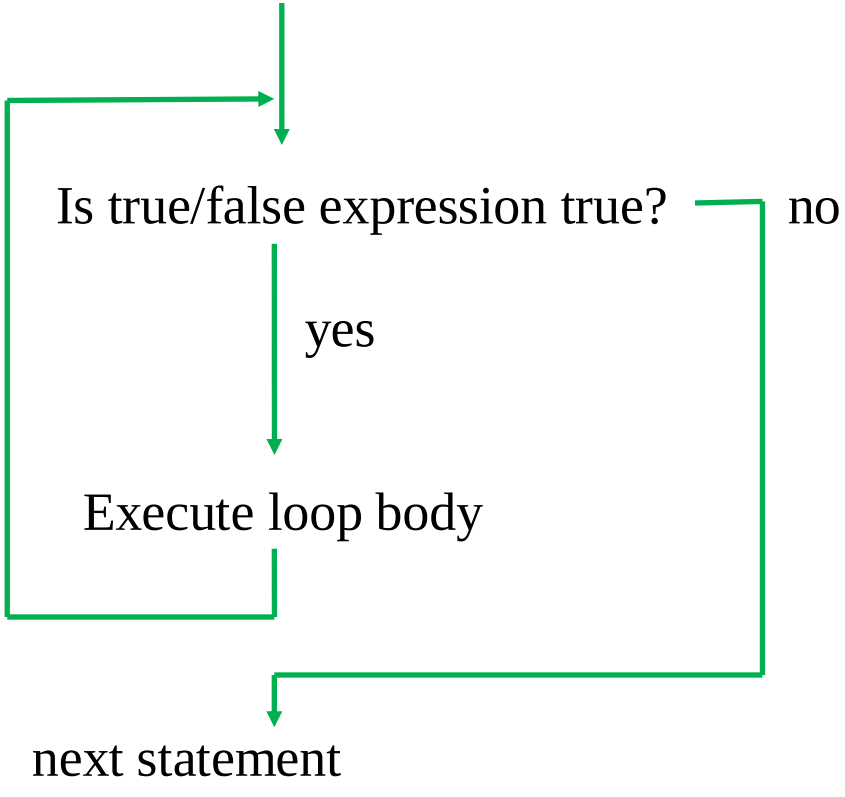
Multi-way Branches

```
if (x == 5)
   statement 1
else
   if (x == 6)
      statement 2
   else
      if (x == 7)
         statement 3
      else
         statement 4
```

`while` loop

```
while (true/false expression)
    statement
```

Action of `while` loop



Simple `while` loop

```
 1 class CountControlled1
 2 {
 3    public static void main(String[] args)
 4    {
 5       int count = 1;
 6       while (count <= 3)
 7       {
 8          System.out.println("star");
 9          System.out.println("moon");
10          count++;
11       }
12       System.out.println("good night");
13    }
14 }
```

Can also count down

```
count = 3;
while (count >= 1)
{
    ...
    count--;
}
```

Sum of the integers 1 to 10

```
 1 class CountControlled2
 2 {
 3    public static void main(String[] args)
 4    {
 5       int count = 1, sum = 0;
 6
 7       while  (count <= 10)
 8       {
 9          sum = sum + count;
10          count++;
11       }
12       System.out.println("sum = " + sum);
13    }
14 }
```

Using a loop

1) Initialization
2) Loop
3) Use the result computed by loop

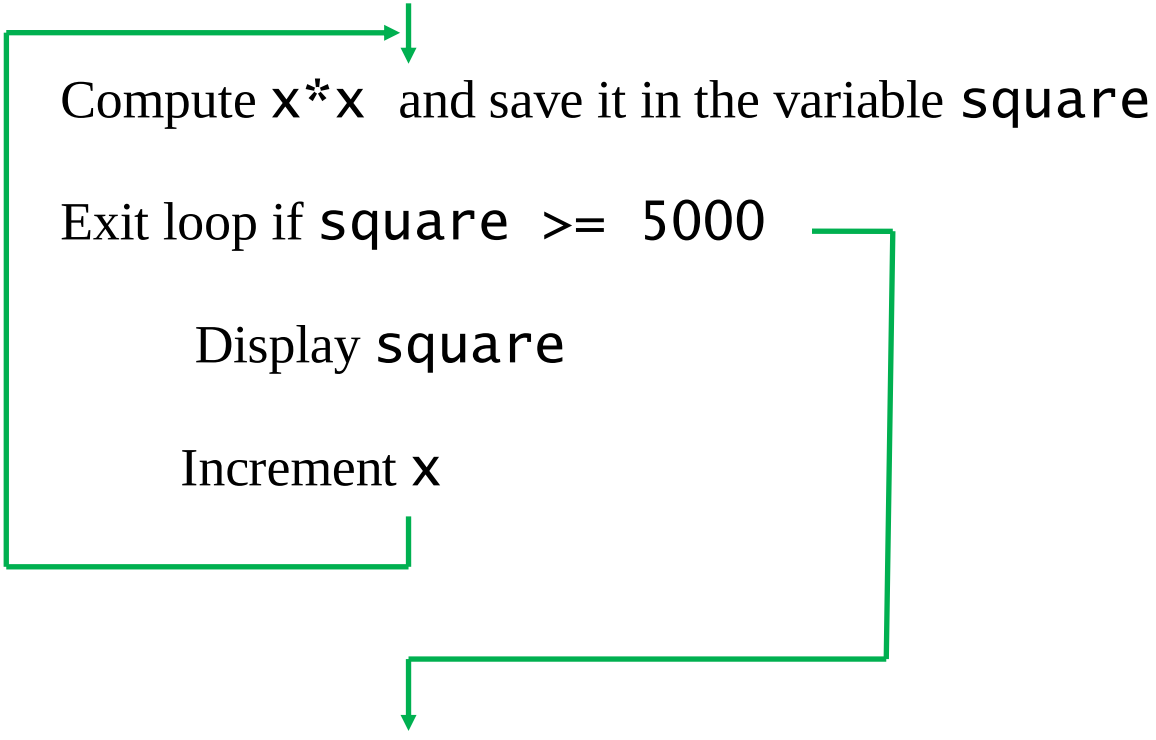
Condition-controlled Loop

```
 1 class ConditionControlled1
 2 {
 3    public static void main(String[] args)
 4    {
 5       int x = 1;
 6
 7       while (x*x < 5000)
 8       {
 9          System.out.println(x + " squared = " + x*x);
10          x++;
11       }
12    }
13 }
```

Display squares less than 500

```
1 squared = 1
2 squared = 4
      .
      .
      .
70 squared = 4900
```

Compute square once



Exit with break

```
 1 class ConditionControlled2
 2 {
 3    public static void main(String[] args)
 4    {
 5       int x = 1, square;
 6
 7       while (true)
 8       {
 9          square = x*x;
10          if (square >= 5000) break;
11          System.out.println(x + " squared = " + square);
12          x++;
13       }
14    }
15 }
```

Interior exit with `while`

```
 1 class ConditionControlled3
 2 {
 3    public static void main(String[] args)
 4    {
 5       int x = 1, square;
 6       boolean keepGoing = true;
 7
 8       while (keepGoing)
 9       {
10          square = x * x;
11          if (square >= 5000)       // exit test
12             keepGoing = false;     // causes exit from loop
13          else
14          {
15             System.out.println(x + " squared = " + square);
16             x++;
17          }
18       }
19    }
20 }
```

Using priming statement

```
 1 class ConditionControlled4
 2 {
 3    public static void main(String[] args)
 4    {
 5       int x = 1, square;
 6
 7       square = x*x;
 8       while (square < 5000)
 9       {
10          System.out.println(x + " squared = " + square);
11          x++;
12          square = x*x;
13       }
14    }
15 }
```

Keyboard input

```
Enter integer greater than 0
10
Sum of 1 to 10 = 55


Enter integer greater than 0
20
Sum of 1 to 20 = 210
```

```
 1 import java.util.Scanner;
 2 class KeyboardInput
 3 {
 4    public static void main(String[] args)
 5    {
 6       int count = 1, sum = 0, n;
 7
 8       // create scanner object that represents keyboard
 9       Scanner kb = new Scanner(System.in);
10
11       // prompt user of program for an integer greater than 0
12       System.out.println("Enter integer greater than 0");
13
14       // read in an integer from the keyboard, assign to n
15       n = kb.nextInt();
16
17       //now sum the integers from 1 to n
18       while (count <= n)
19       {
20          sum = sum + count;    // add count to sum
21          count++;              // add 1 to count
22       }
23       // display the sum
24       System.out.println("Sum of 1 to " + n + " = " + sum);
25    }
26 }
```