Tinker Academy

Programming Using Java
(Java For Loop)

Spring 2015

Java For Loop

For Loops

```
for (int i = 0; i < cars.size(); i++) {
    Car car = (Car) cars.get(i);
}
```

Basic For Loop

Powerful, popular

```
for (Object obj : cars) {
    Car car = (Car) obj;
}
```

Enhanced For Loop

Useful when looping through list

Java For Loop

Basic For Loop

```
for (Initialization Statement;Expression;Update Statement) {
     One or More Statement
}
```

Basic For Loop

Basic Structure

If Expression is false, Loop not executed

Expression is not specified, then true

Initialization Statement Executed ONCE

Expression Evaluated BEFORE Loop

Update Statement Executed AFTER Loop

Java For Loop

Basic For Loop

```
for (;;) {
     System.out.println("Within for loop");
}
```

Basic For Loop

Expression is ALWAYS true, loops forever

```
for (;false;) {
     System.out.println("Within for loop");
}
```

Basic For Loop

Expression is ALWAYS false, never loops

Java For Loop

Basic For Loop

```
for (int i = 0;i < 10;) {
    System.out.println("Within for loop");
}
```

Basic For Loop

Expression is ALWAYS true, loops forever

```
for (int i = 0;i > 10;) {
    System.out.println("Within for loop");
}
```

Basic For Loop

Expression is ALWAYS false, never loops

Java For Loop

Basic For Loop

```
for (int i = 0;i < 10;i++) {
    System.out.println("Within for loop");
}
```

Basic For Loop

```
for (int i = 0;i < 10;i--) {
    System.out.println("Within for loop");
}
```

Basic For Loop

Loops 10 times

Loops forever. Why?

Java For Loop

Basic For Loop

```
for (int i = 0;i < 10;i = i + 2) {
    System.out.println("Within for loop");
}
```

Basic For Loop

```
for (int i = 0;i < 10;i + 1) {
    System.out.println("Within for loop");
}
```

Basic For Loop

Loops 5 times

Is this Right?

Java For Loop

Basic For Loop

```
for (int i = 0,j=10;i < 10;i++,j++) {
    System.out.println("Within for loop");
}
```

Basic For Loop

```
for (int i = 0;i < 10;i--) {
    System.out.println("Within for loop");
}
```

Basic For Loop

Multiple variables initialized

Loops forever. Why?

Java For Loop

Complex Initialization

```
int i = 0;
for (i=0,method1();i < 10;i++,method2()) {
    System.out.println("Within for loop");
}
```

Complex Initialization

Initialization Statement can invoke methods

Java For Loop

Complex Update

```
int i = 0;
for (i=0,method1();i < 10;i++,method2()) {
    System.out.println("Within for loop");
}
```

Complex Update

Update Statement can invoke methods

Java For Loop

break

```
01  int i = 0;
02  for (i=0,method1();i < 10;i++,method2()) {
03      if (i  >= 0) break;
04      System.out.println("Within for loop");
05  }
06  System.out.println("Outside for loop");
07
```

Basic For Loop

Loop executed once and breaks at line 03

"Within for loop" never printed

After loops break next line executed is 06

break useful for early termination

Java For Loop

break

```
int i = 0;
for (i=0,method1();i < 10;i++,method2()) {
    for (j=0;j<10;j++) {
        if (j  >= 0) break;
        System.out.println("Within inner for loop");
    }
}
System.out.println("Outside for loop");
```

Basic For Loop

break returns control to outer for loop

```
int i = 0;
for (i=0,method1();i < 10;i++,method2()) {
    for (j=0;j<10;j++) {
        if (j  >= 0) break;
        System.out.println("Within inner for loop");
    }
    System.out.println("Within outer for loop");
}
System.out.println("Outside for loop");
```

Basic For Loop

will "Within outer for loop" be printed

Java For Loop

continue

```
01  int i = 0;
02  for (i=0,method1();i < 10;i++,method2()) {
03       if (i  >= 0) continue;
04       System.out.println("Within for loop");
05  }
06  System.out.println("Outside for loop");
07
```

Basic For Loop

Loop executed 10 times

"Within for loop" never printed

After loops next line executed is 06

continue useful to skip current loop

Java For Loop

return

```
01  int i = 0;
02  for (i=0,method1();i < 10;i++,method2()) {
03      if (i >= 1) return;
04      System.out.println("Within for loop");
05  }
06  System.out.println("Outside for loop");
07
```

Basic For Loop

loops once, control returns to calling method

"Within for loop" printed once

line 06 never executed

return useful to stop executing method

Java For Loop

throw

```
01  int i = 0;
02  for (i=0,method1();i < 10;i++,method2()) {
03       if (i  >= 0) throw new Exception();
04       System.out.println("Within for loop");
05  }
06  System.out.println("Outside for loop");
07
```

Basic For Loop

loops once, throws controls out of for loop

"Within for loop" printed once

line 06 never executed

use throw to handle severe error looping

Java For Loop

spot the errors

```
public static void main(String[] args) {
    int i = 0;
    if (i  >= 0) break;
}
```

Error

```
public static void main(String[] args) {
    int i = 0;
    if (i  >= 0) continue;
}
```

Error

break needs to be within for loop

continue needs to be within for loop