Java7:

public void testStringInSwitch(String param){

final String JAVA5 = "Java 5";

final String JAVA6 = "Java 6";

final String JAVA7 = "Java 7";

switch (param) {

case JAVA5:

System.out.println(JAVA5);

break;

case JAVA6:

System.out.println(JAVA6);

break;

case JAVA7:

System.out.println(JAVA7);

break;

}

}

**Binary Literals:**

Before:

public void testBinaryIntegralLiterals(){

int binary = 8;

if (binary == 8){

System.out.println(true);

} else{

System.out.println(false);

}

}

Java7:

public void testBinaryIntegralLiterals(){

int binary = 0b1000; //2^3 = 8

if (binary == 8){

System.out.println(true);

} else{

System.out.println(false);

}

}

And if you try to declare a constant number in the code, NetBeans will warn you to convert it into a Binary Literal:

to

public void testUnderscoresNumericLiterals() {

int oneMillion\_ = 1\_000\_000;

int oneMillion = 1000000;

if (oneMillion\_ == oneMillion){

System.out.println(true);

} else{

System.out.println(false);

}

}

**Underscore Between Literals:**

public void testUnderscoresNumericLiterals() {

int oneMillion\_ = 1\_000\_000; //new

int oneMillion = 1000000;

if (oneMillion\_ == oneMillion){

System.out.println(true);

} else{

System.out.println(false);

}

}