**YouTube tutorial 48 – final**

**1st class – apples.java:**

**class** apples{

**public** **static** **void** main(String[] args){

tuna tunaObject = **new** tuna(10);

**for** (**int** i=0; i<5;i++){

tunaObject.add();

System.*out*.printf("%s", tunaObject);

}

}

}

**2nd class – tuna.java:**

**public** **class** tuna{

**private** **int** sum;

**private** **final** **int** NUMBER;

**public** tuna(**int** x){

NUMBER = x;

}

**public** **void** add(){

sum+=NUMBER;

}

**public** String toString(){

**return** String.*format*("sum = %d\n", sum);

}

}

**Result:**

sum = 10

sum = 20

sum = 30

sum = 40

sum = 50

If “final” is used, that variable’s value will be absolute. It cannot be changed, nor can it be inherited.

But when I used a different constructor, the value changed depending on varying objects:

**class** apples{

**public** **static** **void** main(String[] args){

tuna tunaObject = **new** tuna(10);

System.*out*.println(tunaObject.NUMBER);

tuna tunaObject2 = **new** tuna(7);

System.*out*.println(tunaObject2.NUMBER);

}

}

**Result:**

10

7