1. Done
2. Done
3. 10 times.
4. 9 times. This is because you started the count with one more, so the loop initialized one less time.
5. E:\CS161\BlackE\_Wk6\Worksheet\_6.java:20: error: '(' expected

while count<10

^

E:\CS161\BlackE\_Wk6\Worksheet\_6.java:20: error: ')' expected

while count<10

// the while couldn’t use the count<10 without the parentheses

1. The loop continued running and outputting the println. Because the modify section was commented out, the count would stay at one, never reach greater than 10, and theoretically continue looping forever.
2. Done
3. 5 times. The count subtracts one with every loop instead of adding one.
4. System.out.println("How many times do you want me to go? ");

int Stuff=Fred.nextInt();

1. 2. It printed as many times as I told it to.
2. The output keeps repeating endlessly because ‘Stuff’ kept increasing in number and the loop would only end when it went below 0.
3. Done
4. 6 times. This number will be different depending on the size of the number. It takes whatever amount is needed to get to 500 tickets.
5. int Total=0;
6. Total=Total+Number;
7. while (Total<500)
8. The output has the same output as #11 and #6 because the loop can’t be modified because the braces kept the modify line tied to the while statement.
9. Total is 0

// The total was initialized to the same value as the test, so the loop didn’t continue.

1. int Ast=0;

while(Ast<10)

{ System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"); Ast=Ast+1;

}

1. int DL=0;

while(DL<40)

{ System.out.print("=");

DL=DL+1;

}

//just changed object, the test, and println to print

1. int DL=0;

while(DL<50)

{ System.out.print("@");

DL=DL+1;

}

1. Done
2. To take an integer and do nothing with it because the while has no parentheses to modify the loop.
3. This program will generate a table of powers of a number.

You just have to tell me what number:

Enter an integer please: 24

//This is what I expected

1. I added parentheses, a modifier for the count to end the loop and a println to separate the outputs.
2. int product=1, count=0;

System.out.println("This program will generate a table of powers of a number.");

System.out.println("You just have to tell me what number: \n\n");

System.out.print("Enter an integer please: ");

int MyNum=Fred.nextInt();

while(count<15)

{ product=product\*MyNum;

count=count+1;

System.out.println(product);

}

System.out.println("Bye for now.....");

1. int product=1, count=0;

System.out.println("This program will generate a table of powers of a number.");

System.out.println("You just have to tell me what number: \n\n");

System.out.print("Enter an integer please: ");

int MyNum=Fred.nextInt();

while(count<15)

{ product=product\*MyNum;

System.out.print(product+", ");

count++;

if(count%5==0)

System.out.println();

}

System.out.println("Bye for now.....");

1. This program will generate a table of powers of a number.

You just have to tell me what number:

Enter an integer please: 9

9, 81, 729, 6561, 59049,

531441, 4782969, 43046721, 387420489, -808182895,

1316288537, -1038305055, -754810903, 1796636465, -1010140999,

Bye for now.....

1. This program will generate a table of powers of a number.

You just have to tell me what number:

Enter an integer please: 75

75, 5625, 421875, 31640625, -1921920421,

1884856489, -369684093, -1956503199, -708851861, -1624282023,

-1562067437, -1190940783, 873754491, 1107077385, 1426425251,

Bye for now.....

1. This program will generate a table of powers of a number.

You just have to tell me what number:

Enter an integer please: 9

9, 81, 729, 6561, 59049,

531441, 4782969, 43046721, 387420489, 3486784401,

31381059609, 282429536481, 2541865828329, 22876792454961, 205891132094649,

Bye for now.....

// long uses more bytes than int and can fit the size of the number without a stack overflow error

1. A) verification of input

B) it verifies the input so it will only continue if it gets what it wants

1. A) I added the line: “Word=Word.toUpperCase();”, to force the y into uppercase whether it was lower or uppercase

B) toUpperCase

1. A) the loop will continue with any input that doesn’t have the character n at the beginning.

B) It is easiest to continue the loop but it isn’t difficult to end it.