Java                                    Name –
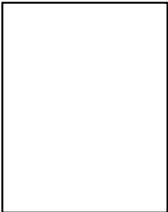while loop worksheet #5                 Period -

Trace the following loops with columns for each variable & circling the last value. Show the output in the rectangles. Write "compile error", "runtime error" or "infinite loop" where appropriate. The ++ operator adds one, the -- operator subtracts one and the % operator means modulus (like Mod in VB). The ! operator means Not, the != means "not equal to", & the == operator means "equal to". Note that these code segments do not necessarily use good style & follow our Coding Standards. Note that if statements in Java work like they do in VB but skim the if statements lecture notes for help if necessary. Be wary of integer division in Java which occurs when an integer such as 3 is divided by 2 the result is truncated to 1 instead of 1.5.
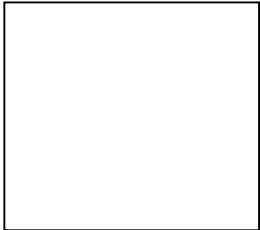
1.

```java
int num = 0;
int sum = 0;

while (num != -1) // -1 is sentinel value
{
    sum += num;
    num = [value inputted by user, assume 3, 2, 4, -1 inputted in that order]
}

System.out.println(sum);
```
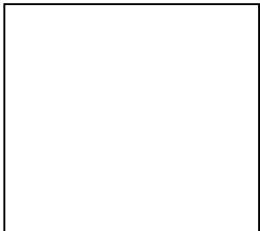
| num | sum |
|-----|-----|
| 3 | 0 |
| 2 | 3 |
| 4 | 5 |
| (1) | (9) |

2.

```java
boolean divisibleBy5 = false; // flag variable
int num = 5;
int iterations = 0;
while (!divisibleBy5)
{
    num += 1;
    iterations++;
    if (num % 5 == 0)
        divisibleBy5 = true;
}

System.out.println(iterations);
```

| divisibleBy5 | iterations | num |
|--------------|------------|-----|
| false | 1 | 6 |
| false | 2 | 7 |
| (false) | 3 | 8 |
| | 4 | 9 |
| | (5) | (10) |

3.

```java
int num = 123;
int reverse = 0;
while (num > 0)
{
    reverse = reverse * 10;
    reverse = num % 10 + reverse;
    num = num / 10; // integer division truncates decimal places
}
System.out.println(reverse);
```

| num | reverse |
|-----|---------|
| 3 | 3 |
| | 32 |
| | (321) |

4.

```java
int num = 8905;
int rev = 0;

while (num > 0)
{
    rev = rev * 10;
    rev = num % 10 + rev;
    num = num / 10;
}

System.out.println(rev);
```

| num | rev |
|-----|-----|