What type of loop do we need? An indefinite loop because we don’t know how many digits are in the integer we will be processing. We’re not sure what the test should be just yet…so let’s leave that blank for now.

What are some tests we can run right away to make sure we actually need to loop over the digits of the integer?

* If the integer is zero then return 0 (no need to loop)
* If the integer is NOT zero, but negative then let’s negate it so we are dealing with a positive number (we don’t care about the sign for this particular problem).

Before we enter a loop we must “PRIME” the loop (initialize a variable that guarantee the loop is entered (or to contain the final result). Here we need to declare the sum and initialize it at zero (because we haven’t added up any digits yet).

What are some strategies to loop over the integer one digit at a time? We read left to right and that feels normal, but it is hard to process numbers that way. It is easier to process from right to left (start with the digit in the 0 -9 range). How can we isolate that digit? Once we have used the digit we will need a way to mark it as done (or better yet remove it).

A trick to isolate the rightmost digit in base 10 is to use modulo 10 against the number. This will always return the rightmost digit.

int d = num % 10;

Then we can add that digit to our sum

sum += d;

Now that we have processed that digit we need to remove it…this can be done by dividing the original integer by 10 (which, when using integer arithmetic, gets rid of the rightmost number). Now, the next number to process is the NEW rightmost number…so we return to the beginning of our loop.

It is now clear what our test should be…if the number is > 0 we should continue to isolate the rightmost digit and add it to our sum…once the adjusted original number (after dividing by 10 the final time) is zero then we are done and can exit our loop.

Once we exit the loop we return our sum.

public static int digitSum(int num) {

if (num == 0) return 0; // handle trivial solution

if (num < 0) num \*= -1; //make it a positive integer

int sum = 0; // initialize sum at zero (the default prior to summing any digits)

while (num > 0) {

int d = num % 10; //extract the last digit

sum += d; //process the last digit

num /= 10; // truncate (or "chop off") the last digit (because it is already processed)

}

return sum; // return the sum to calling program

}