(i) The integer class is final, meaning it is immutable and cannot be extended.

(ii) Integer is a primitive wrapper class and allowing for it to be extended could create cases where its basic way of handling integers is modified. Also, the methods that are present in Integer would never need to be changed or overridden.

An example of a scenario where things could go really bad if they did not do it this way is overriding a static method in the integer class. For the remainder of the runtime of the program the method that will be run when a user calls on the static method will be that of the subclass.

ex. Overiding public static int **bitCount**(int i) to return 0 no matter what will affect every call to the Integers bitCount method.

(iii) A solution to this problem that doesn't involve subclassing is, creating a class that holds an Integer object as a variable and has a new method that returns the string representation of the integer object. The method can simply call on the Integer object for the data it needs.