1,
The add method takes arguments of type Equipment because all equipment types are subclasses of the Equipment class. Using a more specific argument type like Guitar would require overloading the add method with all possible arguments for different equipment types. It would also require creating a new add() method for every new equipment type added.

2,
String is used instead of Equipment in the HashMap because it allows equipments of the same type to have the same key value. Using Equipment objects would not work because adding two instances of the same class would create two separate keys in the hashmap. For example two Guitar objects would create two different keys in the HashMap.

3,
We use Integer instead of int in the HashMap because it doesn't take primitive types as keys or values, it only stores objects. The Integer class is a wrapper class for int that converts an int to an object.

4,
The default toString method is not used because it includes the package name which is not needed, and a hashcode value, so different instances of the same class may return different string values. Overriding the toString method also makes it more clear what the object is, and makes it easier to format string outputs that use the string representation of the object.

5,
It is possible to only define a custom toString method for the Equipment class, as its subclasses inherit the method. To make each subclass return a unique string, we can include a field in the Equipment class for the type and have the subclasses change the type field in the constructor method.