**String Patterns**

You should compare the string using a regular expression, for example:

* str.matches("^[A-Z]{2}\\d{4}") will give you a boolean value as to whether it matches or not.

The regular expression works as follows:

* ^ Indicates that the following pattern needs to appear at the beginning of the string.
* [A-Z] Indicates that the uppercase letters A-Z are required.
* {2} Indicates that the preceding pattern is repeated twice (two A-Z characters).
* \\d Indicates you expect a digit (0-9)
* {4} Indicates the preceding pattern is expected four times (4 digits).

Using this method, you can loop through any number of strings and check if they match the criteria given. You should read up on regular expressions though, there are more efficient ways of storing the pattern if you are worried about performance.

An example for your Pattern could be

* ^[A-Z]{2}\d{4}$

**Java Source Code**

**import** java.io.\*;

**public** **class** StringPatterns {

**public** **static** **void** main(String[] args) **throws** IOException {

**boolean** validInput=**false**;

// Pattern = "TV1523";

System.***out***.println("Enter code");

BufferedReader br = **new** BufferedReader((**new** InputStreamReader(System.***in***)));

String tocompare = br.readLine();

**if**(tocompare.matches("^[A-Z]{2}\\d{4}")) {

validInput=**true**;

} **else** {

validInput=**false**;

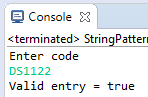
}

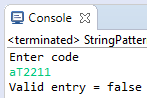
System.***out***.println("Valid entry = " + validInput);

}

}

**Test Data**





Further reading: <https://docs.oracle.com/javase/tutorial/essential/regex/>