What are Regular Expressions in JAVA?

-->Regular expressions are an extremely useful tool for manipulating text. Regular expressions are heavily used in the automatic generation of Web pages. A regular expression is a kind of pattern that can be applied to text .

-->Benefits of Regular Expressions -If a regular expression matches a part of the text, then you can easily find out which part -If a regular expression is complex, then you can easily find out which parts of the regular expression match which parts of the text -With this information, you can readily extract parts of the text, or do substitutions in the text

Package-java.util.regex

-->Parts of Regular Expression

Pattern Pattern p = Pattern.compile("[a-z]+");

Matcher Matcher m = p.matcher("Now is the time");

Some Simple patterns

abc exactly this sequence of three letters

[abc] any one of the letters a, b, or c

[^abc] any character except one of the letters a, b, or c (immediately within an open bracket, ^ means “not,” but anywhere else it just means the character ^)

[a-z] any one character from a through z, inclusive

[a-zA-Z0-9] any one letter or digit

Date and Time in JAVA 8?

Package

java.time

java.time.LocalDate

java.time.chrono

java.time.format

java.time.temporal

java.time.zone

These all classes have predefined methods that helps us to ease our work of inserting time and date in desired format in a java application.

JDBC - Java Database Connectivity

-->It is an API used to connect and execute statements on Database with the help of Drivers.

ResultSet in Java

-->A ResultSet is a Java object that contains the results of executing an SQL query.

Statements in JDBC

-->The statement interface is used to create SQL basic statements in Java it provides methods to execute queries with the database. There are different types of statements that are used in JDBC as follows:

-->Create Statement

-->Prepared Statement

-->Callable Statement

e

package RExpression;

import java.util.regex.\*;

public class RegexExample{

public static void main(String args[]){

System.out.println(Pattern.matches("[am]", "abcd"));//false (not a or m or n)

System.out.println(Pattern.matches("[am]", "a"));//true (among a or m or n)

System.out.println(Pattern.matches("[am]", "ammmna"));//false (m and a comes more than once)

}

}