1. איך יוצרים תיקייה בגאווה?

כדי ליצור תיקייה בג׳אווה, נשתמש במחלקה `File` מתוך החבילה `java.io`. באמצעות השיטה `mkdir()` אפשר ליצור תיקייה חדשה אם היא לא קיימת כבר.

import java.io.File;

public class CreateFolderExample {

public static void main(String[] args) {

File folder = new File("myFolder");

if (folder.mkdir()) {

System.out.println("Folder created successfully.");

} else {

System.out.println("Folder creation failed or already exists");

}

}

}

1. איך יוצרים קובץ?

ליצירת קובץ חדש, אפשר להשתמש במחלקה File ובשיטה createNewFile(). חשוב לטפל בחריגות כי השיטה זורקת IOException.

import java.io.File;  
import java.io.IOException;  
  
public class CreateFile {  
 public static void main(String[] args) {  
 File file = new File("myFile.txt");  
 try {  
 if (file.createNewFile()) {  
 System.out.println("The file was created successfully.");  
 } else {  
 System.out.println("The file already exists.");  
 }  
 } catch (IOException e) {  
 System.out.println("An error occurred while creating the file.");  
 e.printStackTrace();  
 }  
 }  
}

1. איך בודקים אם קובץ/תיקייה קיימים ?

ב-Java ניתן לבדוק אם קובץ או תיקייה קיימים בעזרת השיטה exists() של המחלקה File, שבודקת אם הפריט כבר נמצא במערכת הקבצים.

import java.io.File;

public class CheckIfExistsExample {  
 public static void main(String[] args) {  
 File file = new File("myFile.txt");  
 if (file.exists()) {  
 System.*out*.println("The file exists.");  
 } else {  
 System.*out*.println("The file does not exist.");  
 }  
 }  
}

1. איך מציגים את תוכן תיקייה?

כדי להציג את התוכן של תיקייה, אפשר להשתמש בשיטה listFiles() שמחזירה מערך של אובייקטי File שמייצגים את הקבצים והתיקיות בתוכה.

import java.io.File;

public class ListDirectoryContents {

public static void main(String[] args) {

File folder = new File(".");

File[] files = folder.listFiles();

if (files != null) {

for (File f : files) {

System.out.println(f.getName());

}

}

}

}

1. ( רשות ) מה ההבדל בין File ל Path ?

ב-Java, המחלקות File ו־Path משמשות לעבודה עם קבצים ותיקיות.

File היא מחלקה ישנה מספריית java.io, שמאפשרת פעולות בסיסיות כמו יצירה, בדיקה ומחיקה.

Path שייכת ל-API החדש (java.nio.file), ומציעה עבודה מודרנית, בטוחה וגמישה יותר עם נתיבי קבצים.

במרבית הפרויקטים החדשים מומלץ להשתמש ב-Path ובמחלקה Files.

import java.io.File;

import java.nio.file.Path;

import java.nio.file.Paths;

public class FileVsPath {

public static void main(String[] args) {

File file = new File("example.txt");

Path path = Paths.get("example.txt");

System.out.println("File path: " + file.getPath());

System.out.println("Path path: " + path.toString());

}

}

קישורים למקורים :

<https://jenkov.com/tutorials/java-io/file.html>

<https://www.tutorialspoint.com/create-a-directory-in-java>

<https://stackoverflow.com/questions/6903335/java-path-vs-file>

<https://www.baeldung.com/java-path-vs-file>

<https://docs.oracle.com/javase/8/docs/api/java/io/File.html>