**1. How do you create a directory in Java?**

To create a directory in Java, use the File class and call the mkdir() method. It returns true if the folder was created successfully.

**Code example:**

File folder = new File("MyFolder");

boolean created = folder.mkdir();

Reference link:

[File (Java Platform SE 8 )](https://docs.oracle.com/javase/8/docs/api/java/io/File.html#mkdir--)

**2. How do you create a file in Java?**

Use the File class with the createNewFile() method. This method may throw an IOException, so it's used inside a try-catch block.

* **Code example:**

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

try {

boolean created = file.createNewFile();

} catch (IOException e) {

e.printStackTrace();

}

Reference link:

[File (Java Platform SE 8 )](https://docs.oracle.com/javase/8/docs/api/java/io/File.html#createNewFile--)

**3. How do you check if a file or folder exists in Java?**

**Short explanation:**  
Use the exists() method from the File class to check if a file or directory exists.

* **Code example:**

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

if (file.exists()) {

System.out.println("File exists.");

} else {

System.out.println("File does not exist.");

}

**Reference link:**

[File (Java Platform SE 8 )](https://docs.oracle.com/javase/8/docs/api/java/io/File.html#createNewFile--)

**✅ 4. How do you list the contents of a directory in Java?**

Use the listFiles() method from the File class to get an array of all files and folders inside a directory.

* **Code example:**
* File[] files = currentDirectory.listFiles()**;**for (int i = **0;** i < files.length**;** i++) {  
   File file = files[i]**;** if (file.isDirectory() == true) {  
   System.*out*.println("[DIR] " + file.getName())**;** } else if (file.isFile() == true) {  
   System.*out*.println("[FILE] " + file.getName())**;**

**Reference link:**[File (Java Platform SE 8 )](https://docs.oracle.com/javase/8/docs/api/java/io/File.html#createNewFile--)

5. What is the difference between File and Path in Java?

File is an older class used for file and directory handling. Path is part of the newer java.nio.file package and offers more advanced and flexible operations for working with file paths.

* Code example:

Using File:

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

Using Path:

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

* Reference link:

[**Java – Path vs File | Baeldung**](https://www.baeldung.com/java-path-vs-file)