

Java Zip ファイルの処理

[Java][Java コード片]

例

参照

```
import java.io.File;
import java.io.FileInputStream;
import java.io.IOException;
import java.util.zip.ZipEntry;
import java.util.zip.ZipInputStream;

public abstract class LogAnalyzer{

    // ファイルの配列を処理する
    public void analyze(File[] logs) {
        try {
            for (File log : logs) {
                if (log.getName().endsWith("zip")) {
                    analyzeZipFile(log);
                } else {
                    analyzeTextFile(log);
                }
            }
        } catch (Exception e) {
            //TODO
            e.printStackTrace();
        }
    }

    // Zip ファイルの場合
    private void analyzeZipFile(File zippedLog) throws IOException {
        ZipInputStream zin = new ZipInputStream(new FileInputStream(zippedLog));
        ZipEntry entry = null;

        while ((entry = zin.getNextEntry()) != null) {
            System.out.println("*****" + entry.getName());
            analyze(zin);
            zin.closeEntry();
        }
        zin.close();
    }

    // テキストファイルの場合
    private void analyzeTextFile(File textFile) throws IOException {
        FileInputStream fis = new FileInputStream(textFile);
        analyze(fis);
        fis.close();
    }

    // ログの処理
    public void analyze(InputStream in) throws IOException {
        BufferedReader br = new BufferedReader(new InputStreamReader(in));

        String line = null;
        while ((line = br.readLine()) != null) {
            System.out.println(line);

            // ログの処理
        }
    }
}
```

圧縮

```
// アーカイブ対象ファイル格納 Key: ファイル名、Value: 対象ファイル
Map<String, List<File>> zipFileMap = new HashMap<>();
:

for(Map.Entry<String, List<File>> zipFileEntity : zipFileMap.entrySet()) {
```

```

File zipFile = new File(zipFileEntity.getKey());
List<File> zipContais = zipFileEntity.getValue();

try(ZipOutputStream
    zout = new ZipOutputStream(new BufferedOutputStream(new FileOutputStream(zipFile)))) {

    byte[] buf = new byte[1024];

    for (File zipContain : zipContais) {
        ZipEntry zipEntry = new ZipEntry(zipContain.getName());
        zout.putNextEntry(zipEntry);
        try(InputStream is = new BufferedInputStream(new FileInputStream(zipContain))) {
            @SuppressWarnings("unused")
            int size = 0;
            while ((size = is.read(buf)) != -1) {
                zout.write(buf);
            }
        }
    }
    for (File zipContain : zipContais) {
        if (zipContain.delete()) {
            delFileCnt++;
        }
    }
} catch (FileNotFoundException e) {
    new BatchException(e.getMessage(), e);
} catch (IOException e) {
    new BatchException(e.getMessage(), e);
}
}

```