

Lettore MP3 Java

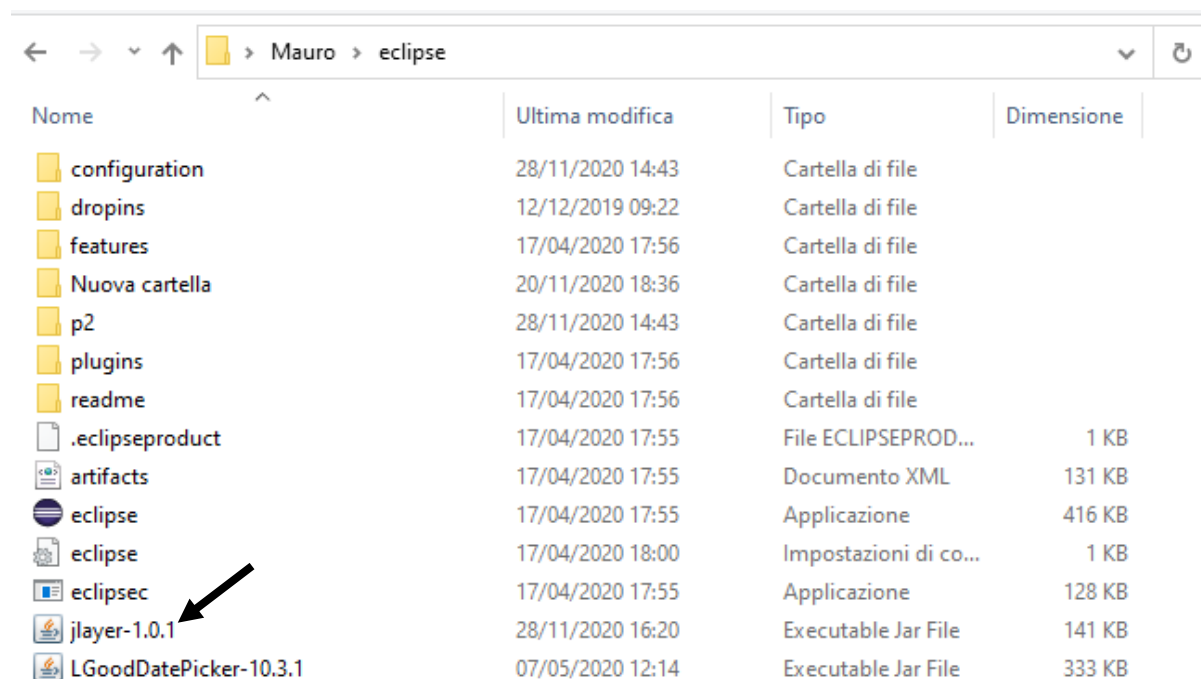
L'esercizio è stato realizzato utilizzando JLayer MP3

(rif: <http://www.javazoom.net/javayer/sources.html>), una libreria che decodifica, riproduce, converte files MP3 in tempo reale per la piattaforma JAVA.

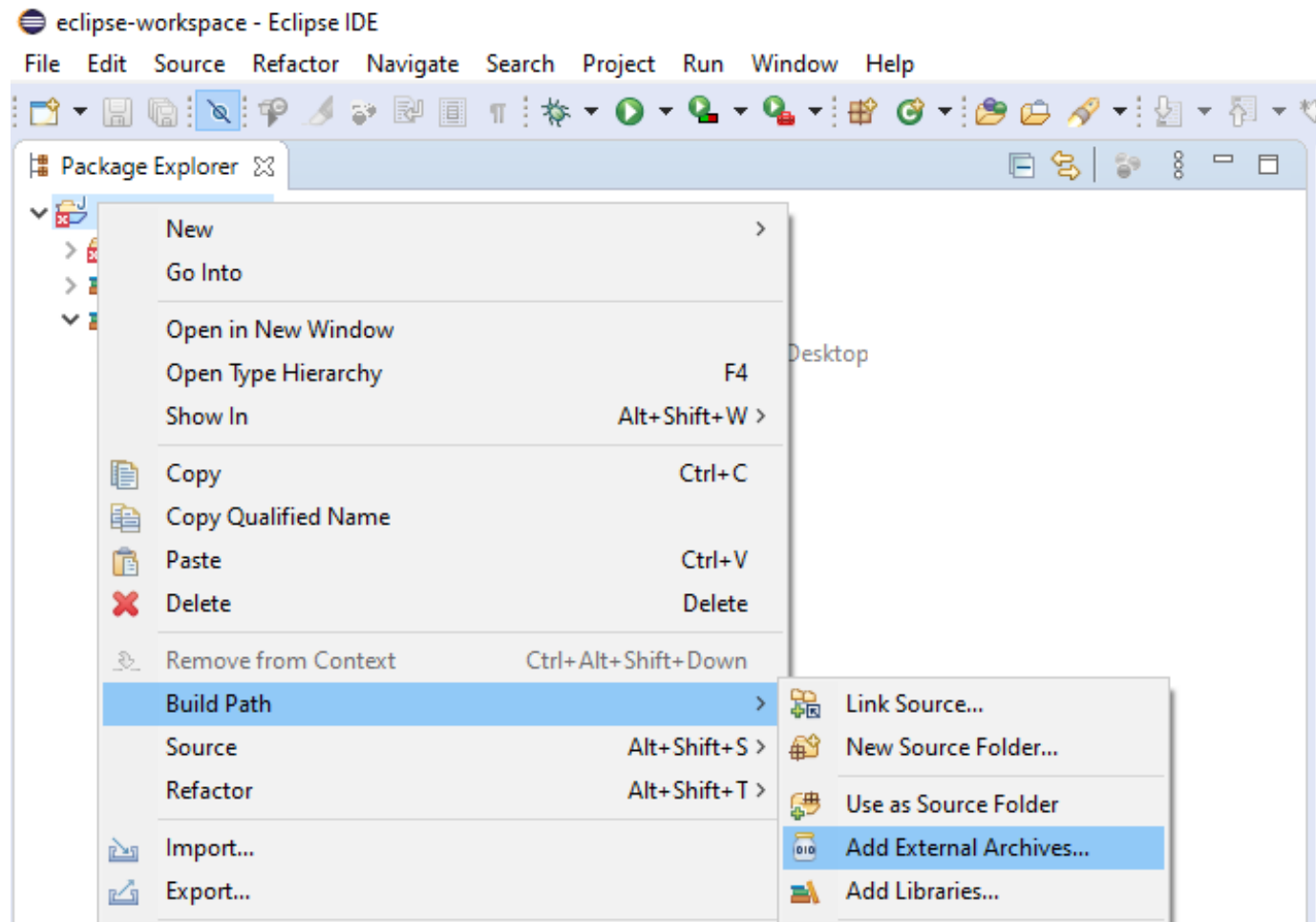
Si tratta di un progetto non commerciale e JLayer è concessa in **licenza LGPL (Lesser General Public License)**

Importare jlayer-1.0.1 in Eclipse

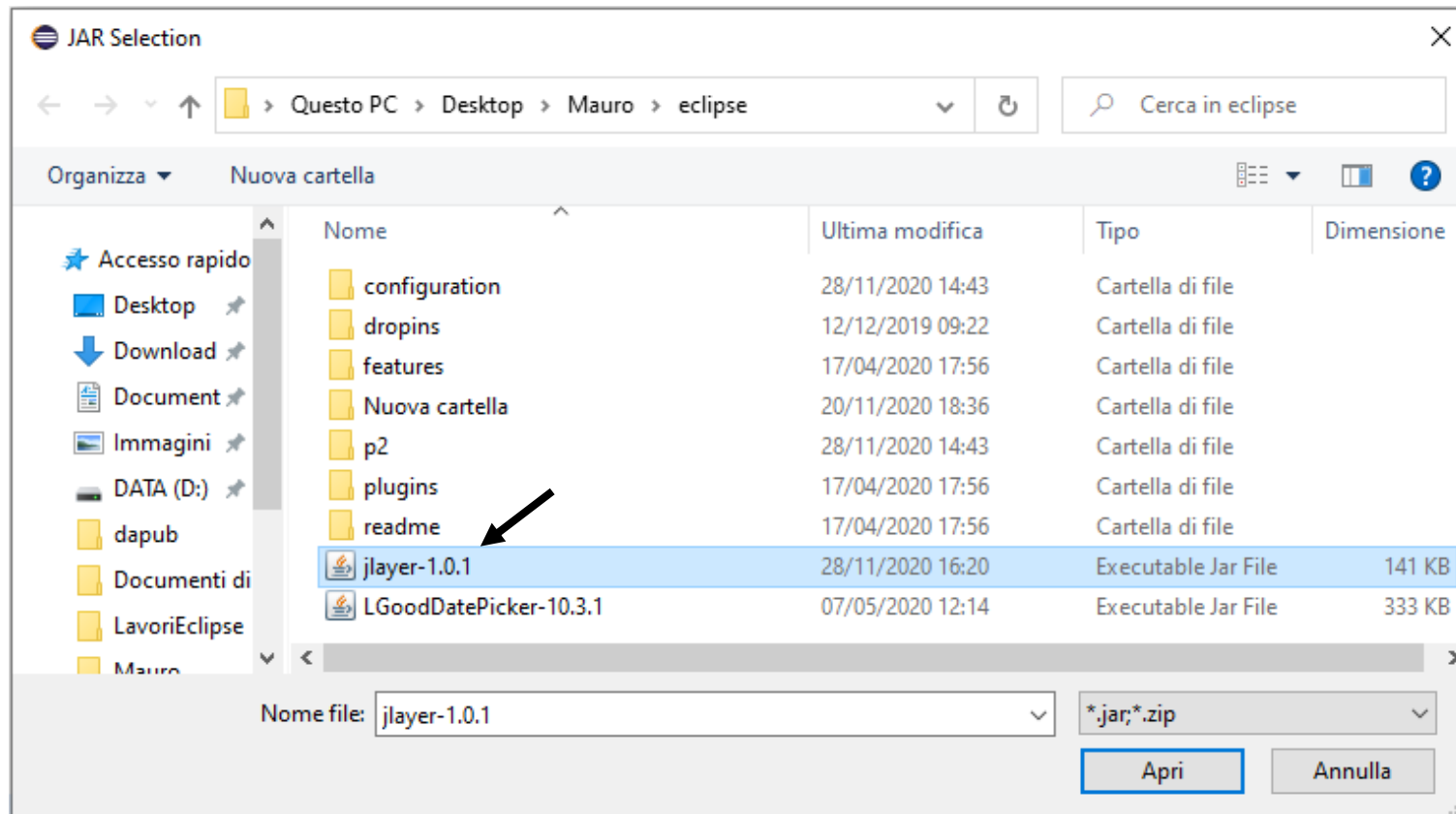
- 1) Decomprimere il file compresso lettoremp3 e salvare jlayer-1.0.1 in una cartella, ad esempio nella cartella di Eclipse



2) in Eclipse Tasto destro sul progetto→Build Path→Add External Archives



- 3) Nella finestra di dialogo successiva, trovare la cartella Eclipse (o quella in cui è stato salvato il file) e aprire jlayer-1.0.1



Codice Java

```

import java.awt.Color;
import java.awt.Container;
import java.awt.GridLayout;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.io.BufferedInputStream;
import java.io.FileInputStream;
import javax.swing.JButton;
import javax.swing.JFileChooser;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.filechooser.FileNameExtensionFilter;

import javazoom.jl.player.Player; // dalla libreria JLayer

public class LettoreMp3 extends JFrame implements ActionListener {
    Container container;
    JLabel lFileMp3;
    JPanel p1,p2;
    JButton bSeleziona,bPlay,bStop;
    String nomefile,pathfile;
    final String labeliniziale="Seleziona un file Mp3";
    final String button1="  Seleziona file M3  ";
    final String button2="  Play  ";
    final String button3="  Stop  ";
    FileInputStream file;
    BufferedInputStream bufferfile;
    Player player;
    String fileLocation;

```



```
public LettoreMp3()
{
    container=this.getContentPane();
    container.setLayout(new GridLayout(2,1));
    // il contenitore è organizzato in 2 righe e 1 colonna su cui saranno aggiunti
    // due panel p1 e p2 (p1 con la label del nome file e p2 con i tre bottoni)
    Color sfondo1 = new Color(0, 64, 0);//verde scuro
    p1=new JPanel();
    p2=new JPanel();
    p1.setBackground(sfondo1);
    lFileMp3=new JLabel(labeliniziale);
    lFileMp3.setForeground(new Color(255,255,255));
    p1.add(lFileMp3);
    bSeleziona=new JButton(button1);
    bPlay=new JButton(button2);
    bStop=new JButton(button3);
    p2.add(bSeleziona);
    p2.add(bPlay);
    p2.add(bStop);
    container.add(p1);
    container.add(p2);
    bSeleziona.addActionListener(this);
    bPlay.addActionListener(this);
    bStop.addActionListener(this);
    this.setTitle("Lettore Java MP3   MDB2020");
    this.setResizable(false);
    this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    this.pack();
    this.setSize(480,120);
    this.setVisible(true);

} // chiude costruttore -----
```

```

public static void main(String[] args) {
    new LettoreMp3();
} // chiude main() -----

@Override
public void actionPerformed(ActionEvent e) {
    String azione=e.getActionCommand();
    switch(azione) {
        case(button1):
            JFileChooser fc = new JFileChooser();
            fc.setDialogTitle(labeliniziale);
            FileNameExtensionFilter filtro=new FileNameExtensionFilter("Files Mp3", "Mp3");
            fc.setFileFilter(filtro);
            if(fc.showOpenDialog(bSeleziona)==JFileChooser.APPROVE_OPTION)
            {
                nomefile=fc.getSelectedFile().getName();
                pathfile=fc.getSelectedFile().getPath();
                lFileMp3.setText(nomefile);
            }
            break;
        case(button2):
            if(lFileMp3.getText()==labeliniziale)
            {
                return;
            }
            try {
                file = new FileInputStream(pathfile);
                bufferfile= new BufferedInputStream(file);
                player = new Player(bufferfile);
            }
            catch (Exception exc) {
                JOptionPane.showMessageDialog(this, exc.getMessage());
                exc.printStackTrace();
            }
    }
}

```

```
        new Thread() { // nuovo Thread per eseguire il brano in background
            public void run() {
                try {
                    player.play();
                }
                catch (Exception exc) {
                    exc.printStackTrace();
                }
            }
        }.start();

        break;
    case(button3):
        if(player!=null){
            player.close();
            lFileMp3.setText(labeliniziale);
        }
        break;
    } //---chiude switch(azione)-----
} //-----chiude actionPerformed()-----

} // chiude class LettoreMp3 -----
```