

Guida all'utilizzo delle librerie e delle direttive di compilazione

Di P. Foglia

1.0 Utilizzo dei Package in JSDK

Per poter usare i package, è necessario:

- 1) indicare nel programma il nome dei package (ossia delle librerie) contenenti le definizioni degli oggetti e dei metodi che si vogliono utilizzare, tramite la direttiva di import
- 2) indicare al compilatore (comando javac) dove trovare i file contenenti tali definizioni, tramite l'opzione di compilazione `-classpath`. L'opzione `-classpath` deve indicare il percorso assoluto delle directory contenenti i nomi dei file che definiscono le classi del package. Tali file possono essere compresse in un file con estensione `.jar`. In tal caso occorre indicare, come opzione del classpath, non la directory ma il percorso completo (con il nome) del file `.jar`
- 3) indicare alla JVM dove trovare le classi che costituiscono il programma (sempre tramite l'opzione `-classpath`)

Facciamo un esempio.

I metodi `leggiInt()` e `leggiDouble()` dell'oggetto `Lettore.in` sono definiti nel package `fiji.io`, contenuto nel file `fiji.jar`.

Dunque, nel programma occorre aggiungere la direttiva **`import fiji.io.*;`** e l'opzione da dare al compilatore è `-classpath "percorso completo del file fiji.jar"`

Supponiamo di avere la seguente situazione:
nella cartella `C:\Documents and Settings\Administrator\Desktop\cabibbo\esempi` è contenuto il file compresso `fiji.jar`. In tale directory è contenuto inoltre il file `prova.java`, che contiene il seguente programma java:

```
import fiji.io.*; // direttiva per importare il metodo leggiInt()

public class prova
{

    public static void main(String[] args)
    {

        int a;

        System.out.println("inserisci un numero");
        a = Lettore.in.leggiInt(); //legge da tastiera un int e lo assegna ad a
```

```
        System.out.println(a);
    }
}
```

Allora:

- 1) nel programma si inserisce la direttiva import, come sopra
- 2) si apre il prompt dei comandi e ci si colloca nella directory C:\Documents and Settings\Administrator\Desktop\cabibbo\esempi dove è contenuto sia il file .java contenente il programma sia il file fiji.jar contenente il package
- 3) il comando per compilare il programma è:
javac -classpath .\fiji.jar prova.java
- 4) il comando per eseguire il programma è:
java -classpath .\fiji.jar;. prova
- 5) nota: per mandare in esecuzione la JVM, è necessario indicare nel classpath non solo il percorso del file fiji.jar, ma anche il percorso del file .class contenente il programma, separato dal precedente tramite ; (in questo caso è la directory punto)

Ecco il risultato dell'esecuzione dei due passaggi:

```
C:\WINNT\system32\cmd.exe
C:\Documents and Settings\Administrator\Desktop\cabibbo\esempi>
C:\Documents and Settings\Administrator\Desktop\cabibbo\esempi>C:\jdk1.4.2_03\
bin\javac.exe -classpath .\fiji.jar prova.java
C:\Documents and Settings\Administrator\Desktop\cabibbo\esempi>java -classpath .
.\fiji.jar;. prova
Inserisci un numero
12
12
C:\Documents and Settings\Administrator\Desktop\cabibbo\esempi>_
```

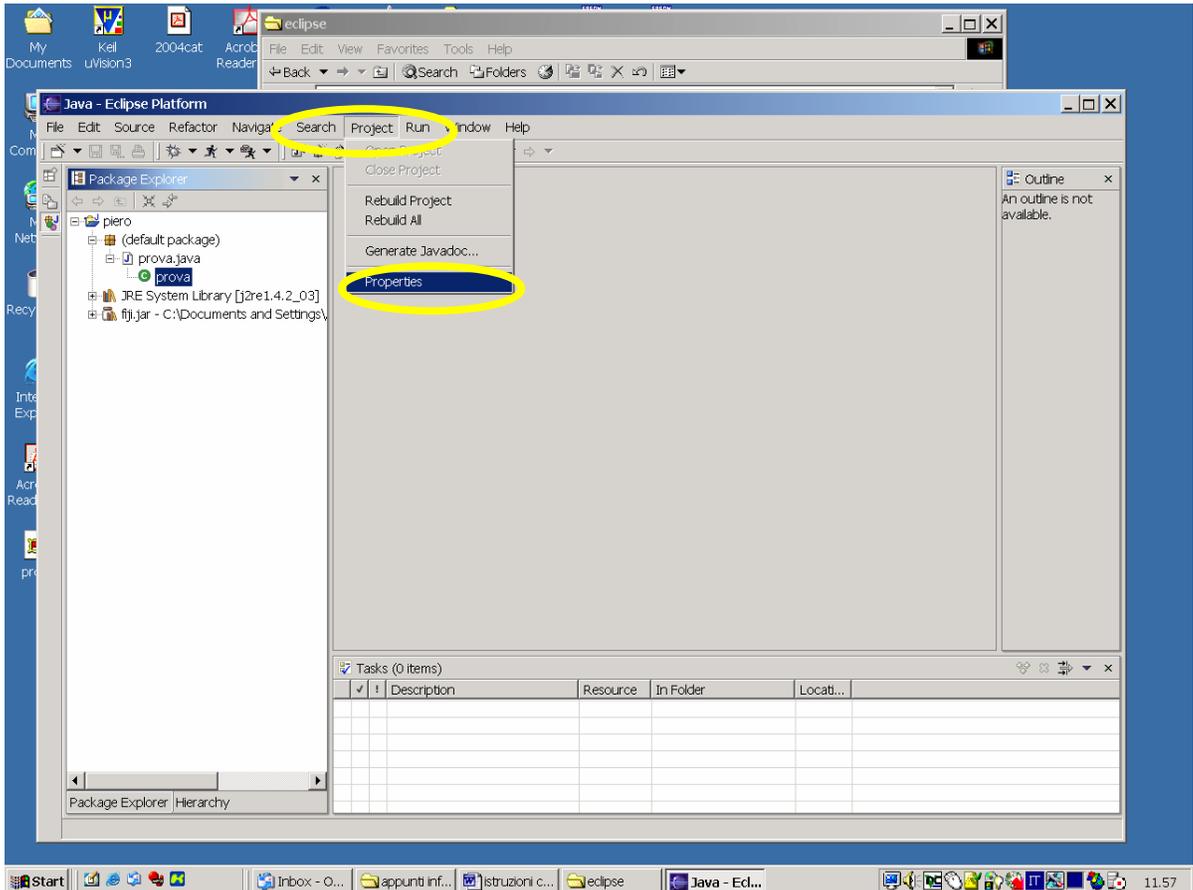
Specifica del classpath per javac (.\fiji.jar)

Specifica del classpath per la JVM (.\fiji.jar;.)

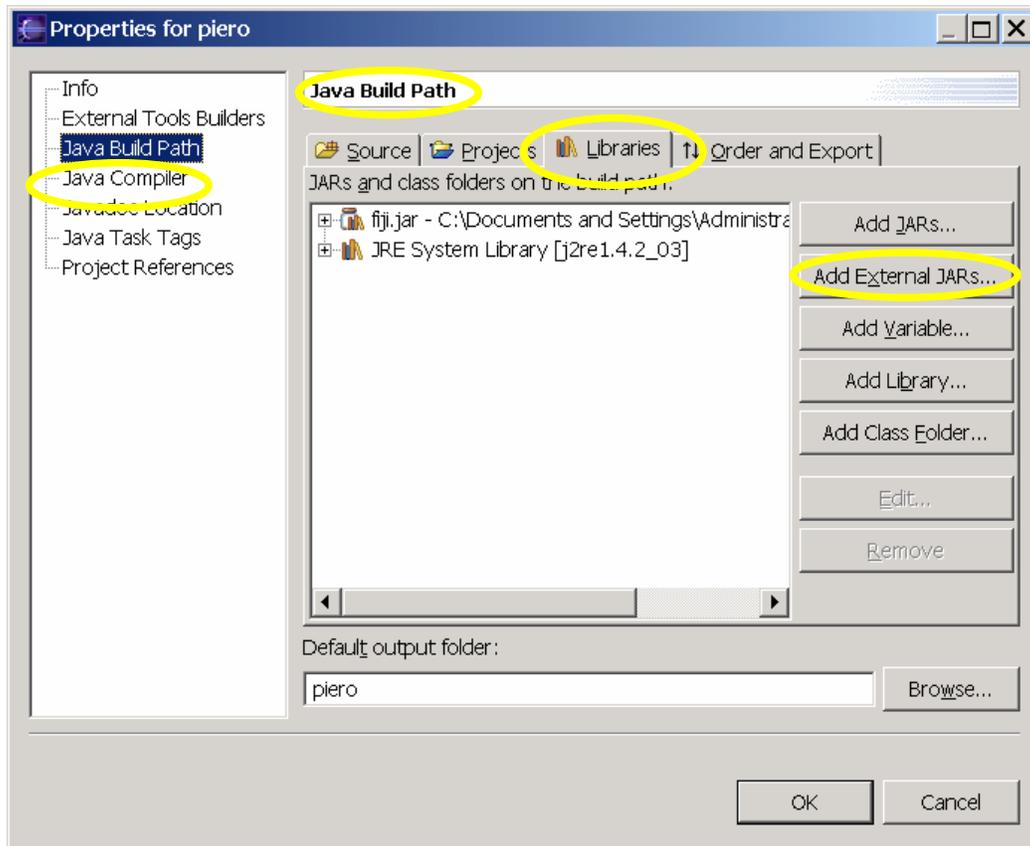
2.0 Utilizzo dei package in eclipse

Occorre lavorare con le proprietà del progetto; in particolare indicare che un jar esterno deve essere assegnato al “building path” del progetto (ossia al path che il compilatore e la JVM utilizzano per trovare le librerie).

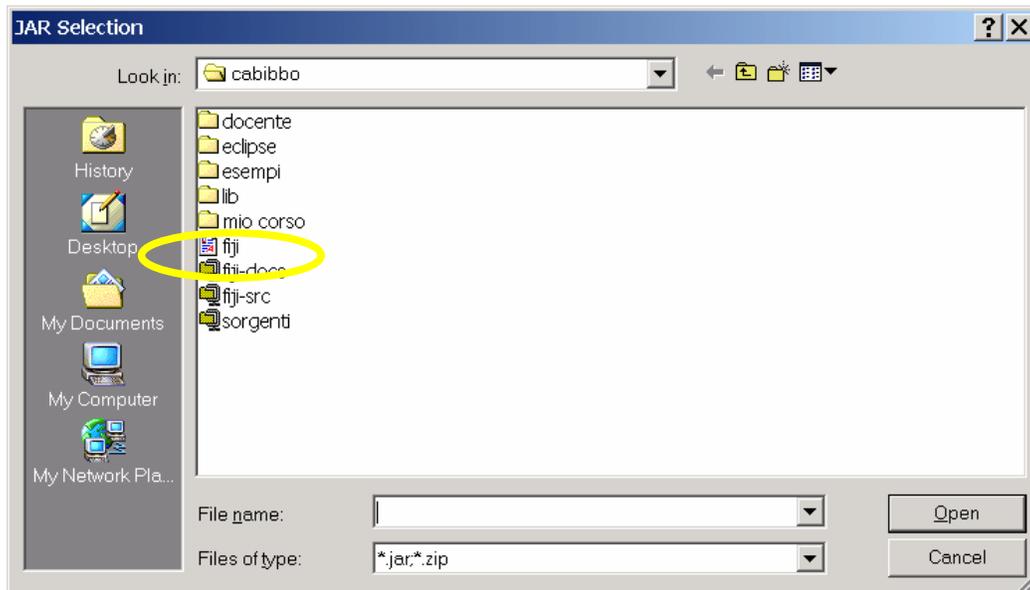
Si apre un progetto e si selezionano le proprietà come in figura:



Dalla lista delle proprietà si seleziona il Java Building path ed il menù libraries come nella figura sottostante.



Infine si seleziona il comando add External Jars. Si apre la finestra sottostante con cui si seleziona il file jar da aggiungere.



A questo punto, la configurazione è finita e si può pertanto compilare ed eseguire il codice.

