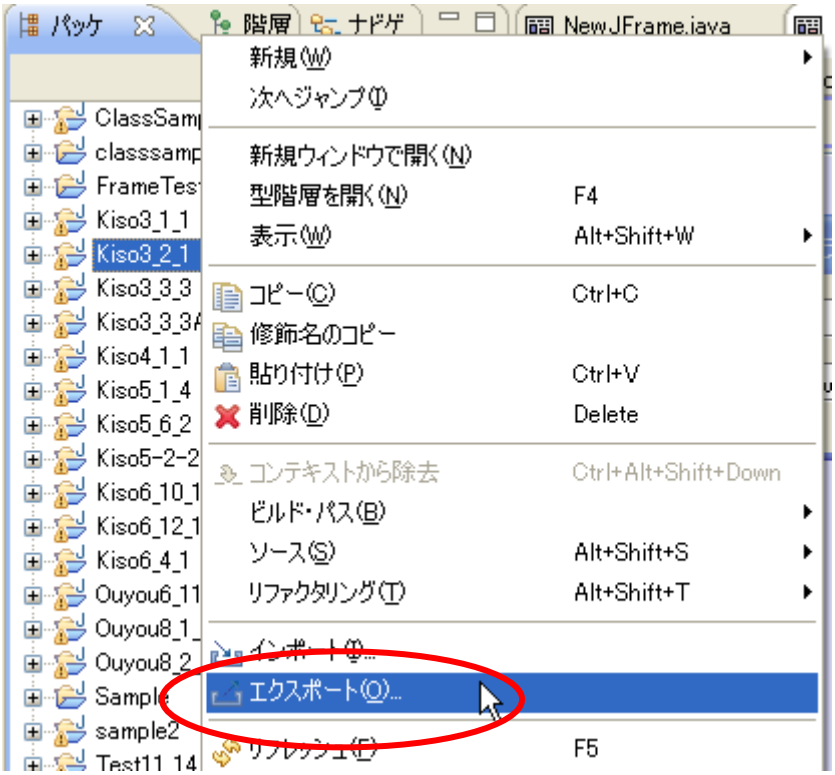
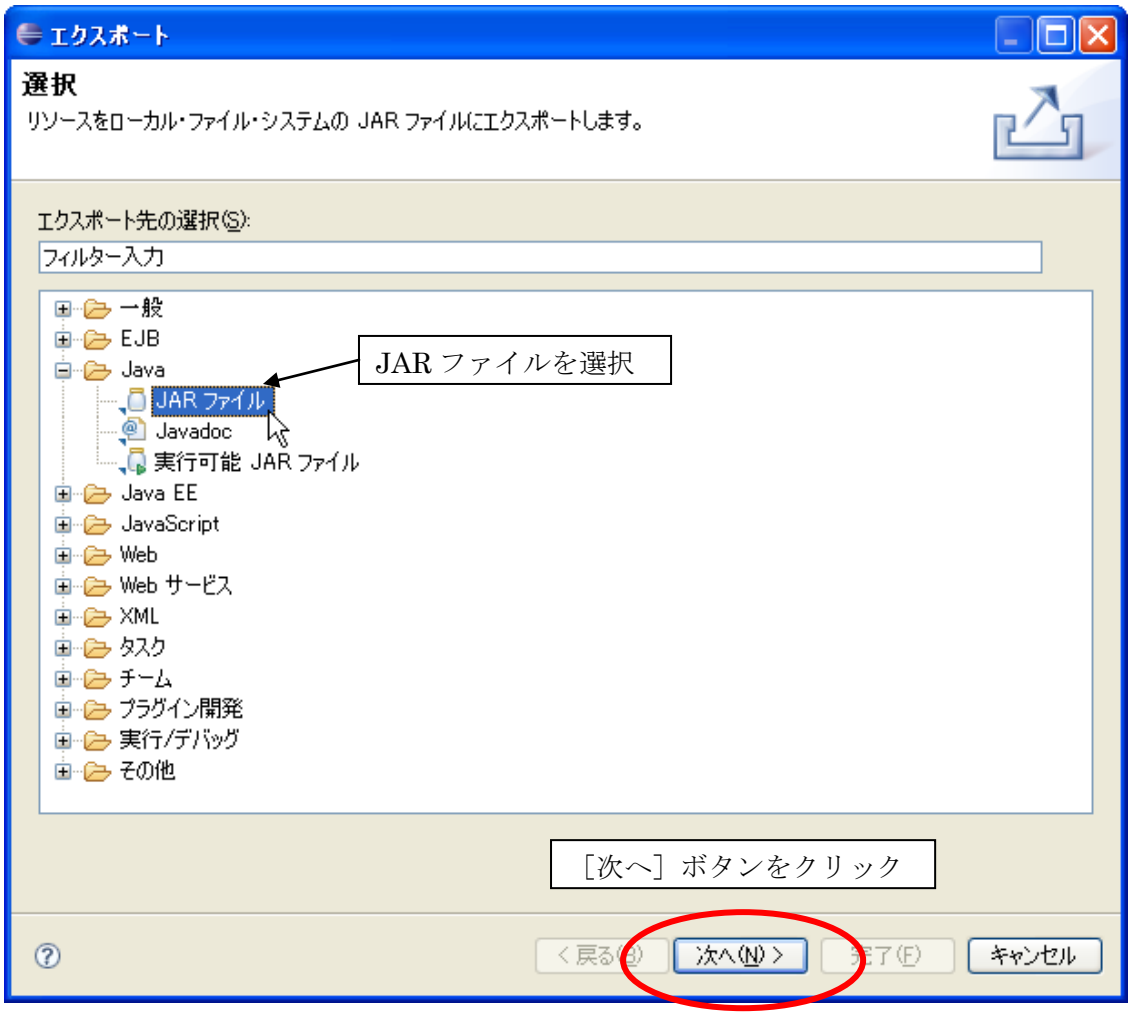
付録 B アーカイブファイル（jar ファイル）の作り方

一般に Java プログラムを配布する場合には、関連するファイルを一括してまとめた**アーカイブファイル**という形式にして行います。そして、アーカイブファイルを受け取った側は、(Java の実行環境があるコンピュータであれば) それをダブルクリックするだけでプログラムを起動できる様になります。その際、相手先に Eclipse などの統合開発環境は必要ありません。本章では、（代表的なアーカイブファイルである）**jar ファイル**を作成する方法を解説します。

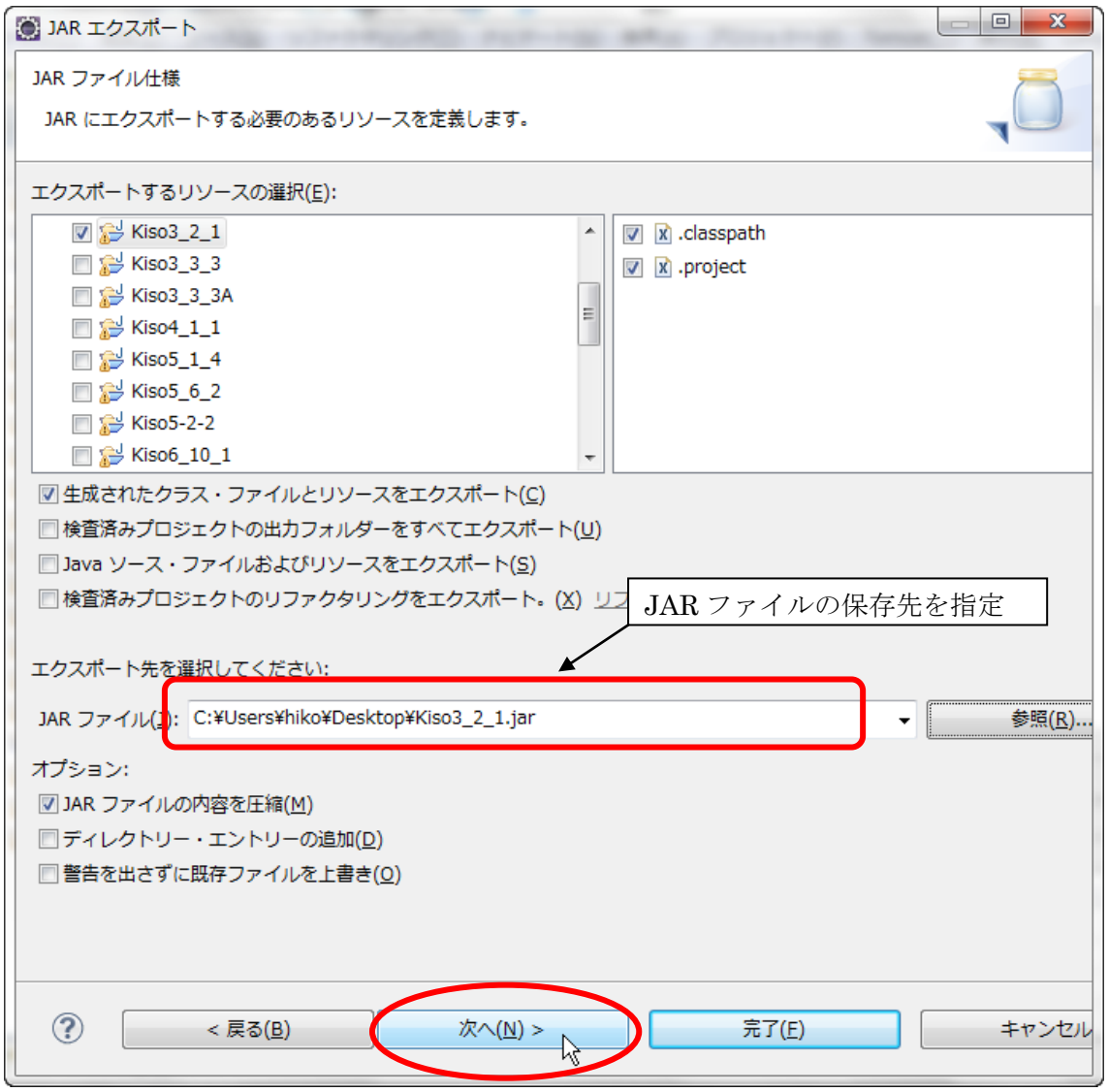
① Eclipse のパッケージエクスプローラから jar ファイルを作る対象となるプロジェクトを右ボタンクリックで選択し（下の例では Kiso3_2_1 を選択しています)、現れたメニューからエクスポートを選択します。



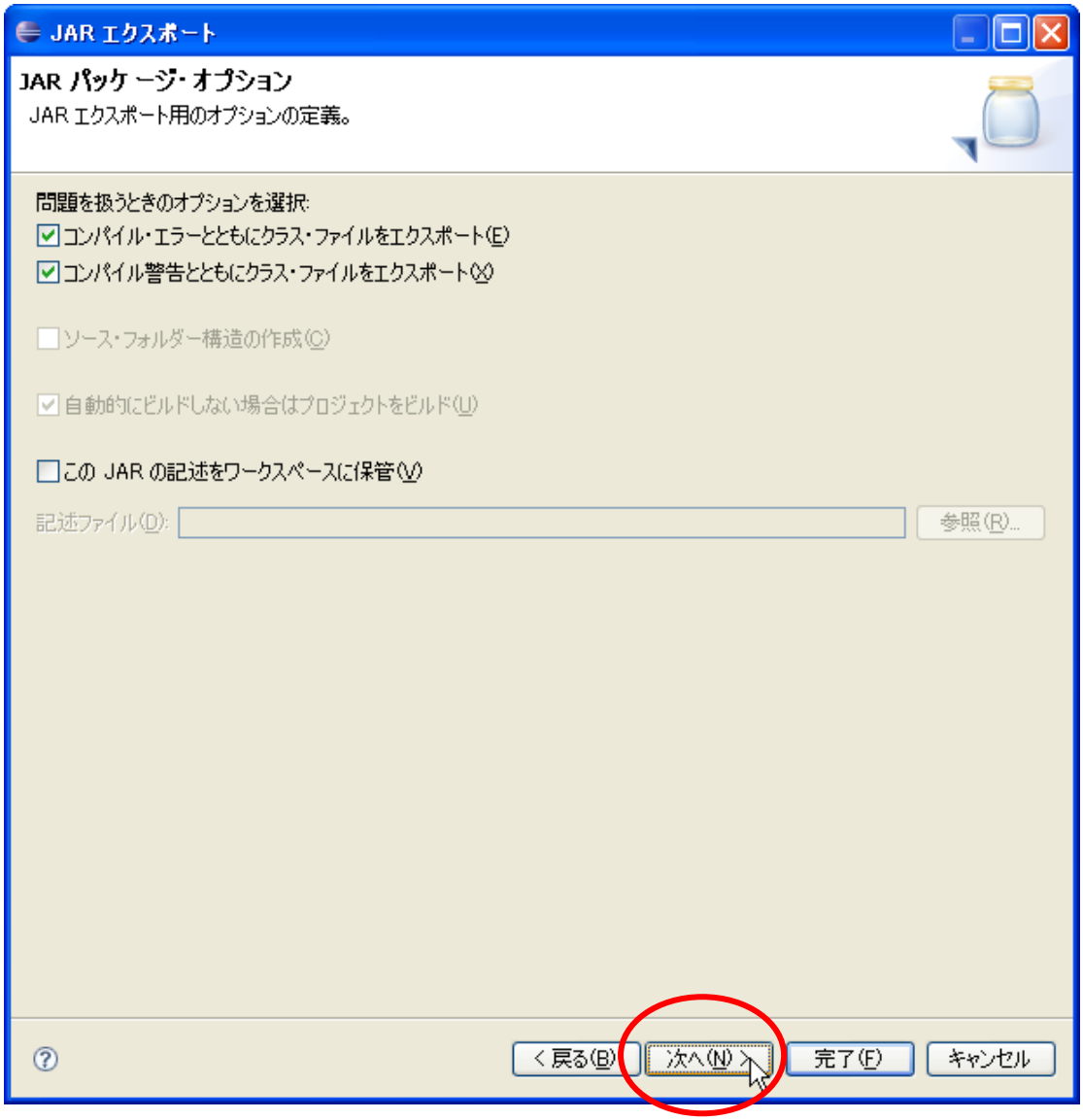
② 続いて現れる画面から次のように「Java」→「JAR」ファイルと選択します。選択後［次へ］ボタンをクリックして次画面に進みます。



③　続いて現れる「JAR ファイル仕様」画面では、作成する JAR ファイルの保存先を指定します。［参照］ボタンをクリックして、適当な保存先を指定します。下の例では、デスクトップに「Kiso3_2_1.jar」と言う名前で保存する、という形で指定しています（［参照］ボタンをクリックして保存先を指定することができます）。指定後は［次へ］ボタンをクリックして次へ進んで下さい。

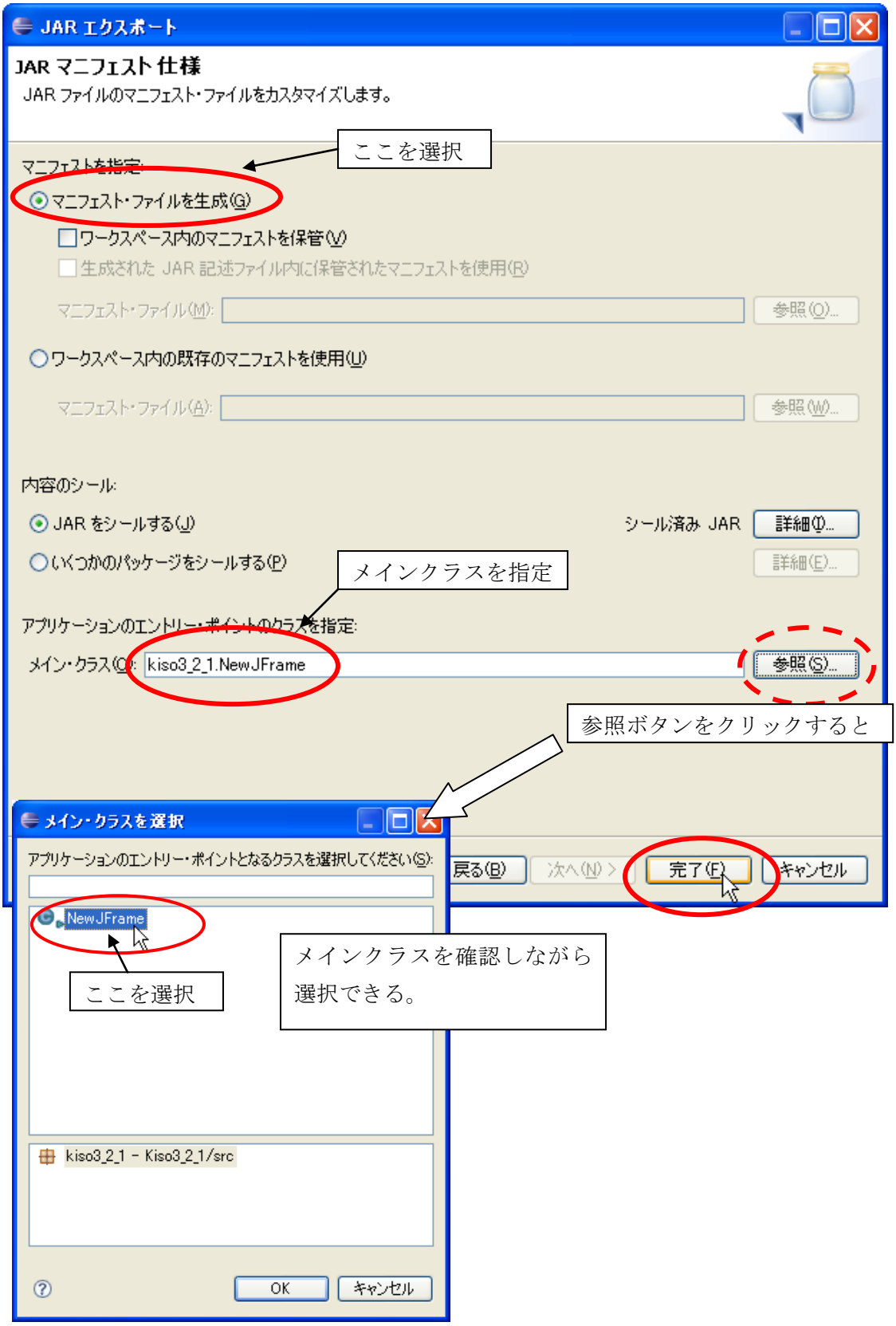


④　次の「JAR パッケージオプション」画面では、特に変更の必要はないのでそのまま［次へ］ボタンをクリックします。

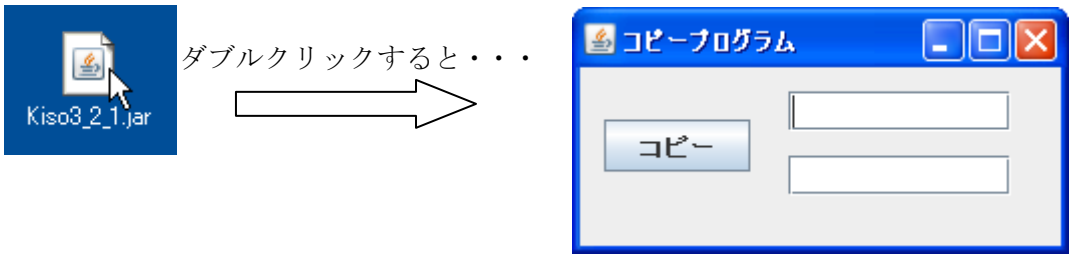


⑤　最後の「JAR マニフェスト仕様」画面では、「マニフェスト・ファイルを生成」を選択し、メインクラスを指定します。メインクラスとは `main()` メソッドを有するクラスのことで、今の場合「kiso3_2_1.NewJFrame」となります。一般にメインクラスは「パッケージ名．クラス名」という形で指定するので、クラス名やパッケージ名に応じて異なって来ます。そこで、［参照］ボタンをクリックしてメインクラスを確認しながら指定した方が安全です。指定後［完了］ボタンをクリックすると JAR ファイル作成完了

です。


JAR エクスポート
JAR マニフェスト仕様
JAR ファイルのマニフェスト・ファイルをカスタマイズします。
ここを選択
マニフェストを指定:
マニフェスト・ファイルを生成(G)
ワークスペース内のマニフェストを保管(V)
生成された JAR 記述ファイル内に保管されたマニフェストを使用(R)
マニフェスト・ファイル(M):
参照(O)...
ワークスペース内の既存のマニフェストを使用(U)
マニフェスト・ファイル(A):
参照(W)...
内容のシール:
JAR をシールする(J)
シール済み JAR
詳細(D)...
いくつかのパッケージをシールする(P)
詳細(E)...
メインクラスを指定
アプリケーションのエントリー・ポイントのクラスを指定:
メイン・クラス(C):
kiso3_2_1.NewJFrame
参照(S)...
参照ボタンをクリックすると
メイン・クラスを選択
アプリケーションのエントリー・ポイントとなるクラスを選択してください(S):
戻る(B)
次へ(N) >
完了(F)
キャンセル
NewJFrame
ここを選択
メインクラスを確認しながら
選択できる。
kiso3_2_1 - Kiso3_2_1/src
OK
キャンセル

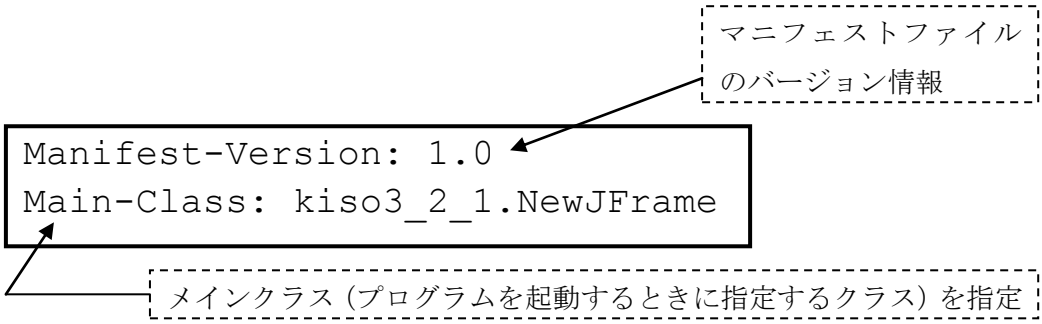
⑥　作成後、当該 jar ファイルをダブルクリックすると、プログラムが起動します。



＜補足＞

上で作成した「Kiso3_2_1.jar」ファイルを解凍（展開）すると、中には次のようなファイルが入っています。

この内、[kiso3_2_1] の中にはクラスファイルが、[META-INF] には「MANIFEST.MF」というマニフェストファイルが入っています。マニフェストファイルは jar ファイルからプログラムを起動するときに必要なファイルです。今の場合マニフェストファイルの中身は次のようになっています。



前ページの画面で指定した情報によって Eclipse がこのマニフェストファイルを作成してくれたのです。Eclipse を用いない場合は自分でこのようなマニフェストファイルを記述しなければなりません。