

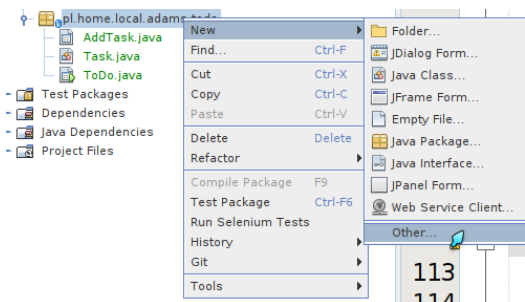
Proste potwierdzanie usunięcia zadania z `JList`:

Używamy `JOptionPane` – element przydatny do potwierdzania operacji wykonywanych przez użytkownika np.:

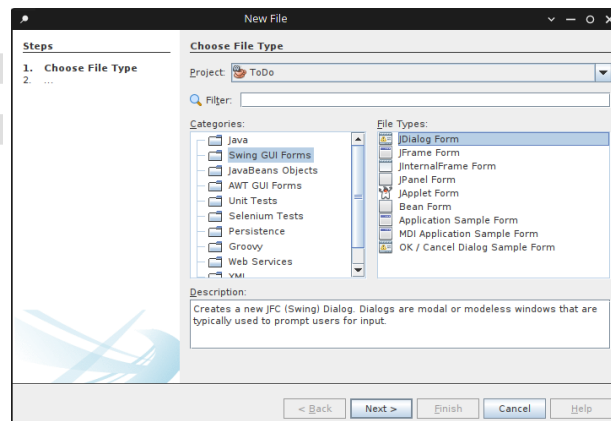
- usunięcie elementu z listy – wypada potwierdzić czy na pewno użytkownik chce usunąć wybrany element z `JList`. Poniższy kod to ilustruje.

```
private void jBRemoveTaskActionPerformed(java.awt.event.ActionEvent evt) {  
    int index = jLTask.getSelectedIndex();  
    String[] options = {"TAK", "NIE"};  
    int chosen = JOptionPane.showOptionDialog(this,  
        "Na pewno chcesz usunąć: "+listOfTasks.get(index).getTitle(), //treść  
        "POTWIERDZASZ USUNIĘCIE?", //tytuł okna  
        JOptionPane.DEFAULT_OPTION, //Defaultowe opcje  
        JOptionPane.WARNING_MESSAGE, //Wbudowana ikona  
        null, //Można swoją ikonę wprowadzić Icon()  
        options, //JAKIE OPCJE MAMY DO WYBORU?  
        options[0]); //Która opcja jest zaznaczona domyślnie - TAK (patrz options)  
    if(chosen == 0){ //Ponieważ TAK jest pierwsze w options  
        dlm.removeElementAt(index);  
        listOfTasks.remove(index);  
    }  
}
```

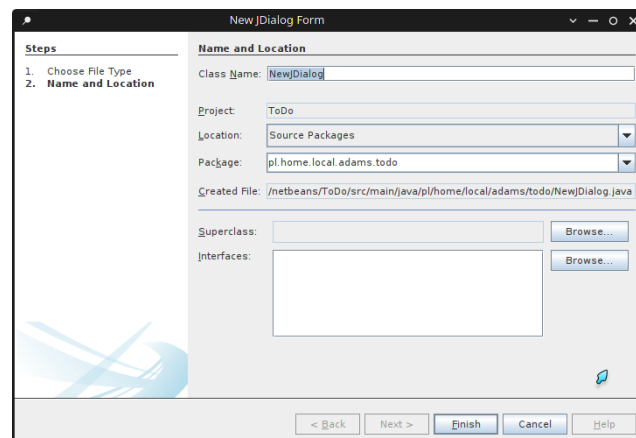
Stworzenie własnego `JDialog` – czyli okna w oknie. Najszybciej robimy to w NetBeans PP na pakiet gdzie chcemy dodać plik → wybieramy :



Pojawi się okno poniżej, wybieramy jak poniżej:

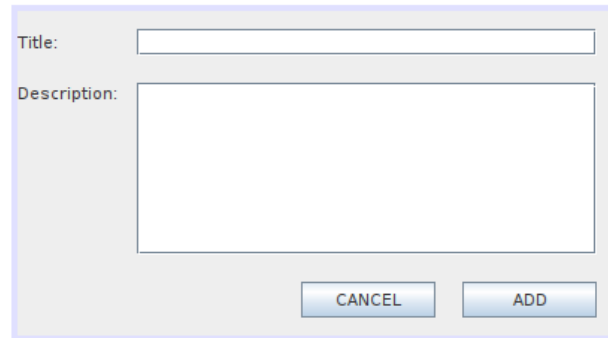


Klikamy next i pojawi się okno w nim podajemy tylko nazwę dla naszego `JDialog`:



UWAGA: po kliknięciu Finish w kodzie klasy która się pojawi kasujemy metodę main

Stwórz w Design okno jak poniżej:

A Java Swing dialog box with a title field, a description text area, and CANCEL and ADD buttons.

Teraz nadaj temu oknu następujące preferencje [jak dla JFrame]:

- Tytuł: Add Task
- Resizable: no
- Pojawić ma się na środku ekranu

Teraz zajmiemy się przekazaniem danych z Okna Dialogowe do Naszego głównego okna, czyli dodaniem elementu do JList.

W oknie dialogowym potrzebujemy metody publicznej, która zwróci nam obiekt **Task**, w którym znajduje się temat i opis naszego zadania!!!

```
public Task getTask(){  
    return new Task(title, description);  
}
```

Oczywiście po naciśnięciu przycisku **Add** okno musi stać się niewidoczne.

```
private void jBAddActionPerformed(java.awt.event.ActionEvent evt) {  
    title = jTFTitle.getText();  
    description = jTADescription.getText();  
    setVisible(false);  
}
```

Po naciśnięciu przycisku **cancel** okno dialogowe też musi stać się niewidoczne. (ostatni linijka w kodzie powyżej wystarczy).

Teraz w oknie głównym programu musimy pobrać nasz obiekt **Task** aby zapisać dane do listy i dodać do widoku (JList).

```
//Metoda pokazująca Nowe Okno dialogowe do dodania zadania, zwraca  
//obiekt TASK z danymi z okna dialogowego  
private Task showJDialog(){  
    AddTaskDialog addTaskDialog = new AddTaskDialog(this,true);  
    addTaskDialog.setVisible(true);  
    //Poniższe zadziała dopiero jak w okno dialogowe zamkniemy  
    //przyciskami lub krzyżykiem  
    Task task = addTaskDialog.getTask();  
    addTaskDialog = null;  
    return task;  
}
```

Powyższa metoda zwraca **Task** – do dalszego wykorzystania w innej metodzie np.: zapisanie do listy i odświeżenie widoku.

```
//Metoda okna głównego. Wyświetla okno dialogowe po naciśnięciu przycisk  
//AddTask następnie sprawdza czy wprowadzono temat i naciśnięto Add  
//w oknie dialogowym  
private void jBAddTaskActionPerformed(java.awt.event.ActionEvent evt) {  
    Task task = showJDialog();  
    if(!task.getTitle().equals("")){  
        listOfTasks.add(task);  
        dlm.addElement(task.getTitle());  
    }  
}
```

Zadania:

- Stwórz BMI ale dane do obliczeń wprowadzaj za pomocą okna dialogowego, wszystkie wprowadzone dane zapisuj do listy w formacie: wzrost, waga.
Wynik w JList:
180, 90, M, BMI = 27.78
160, 52, K, BMI = 20.31
- Stwórz zamianę jednostek w taki sposób, że wprowadzamy poprzez okno dialogowe Celsjusze, a w głównym programie do listy zapisujemy historię zamiany jednostek w JList. Wynik w JList:
20 C = 68 F = 293 K
-10 C = 14 F = 263 K