

03FYZ TECNICHE DI PROGRAMMAZIONE

Istruzioni per la preparazione dell'ambiente di sviluppo di base

- Scaricare ed installare la piattaforma di sviluppo *Java Platform (JDK) 8* dal sito <http://www.oracle.com/technetwork/java/javase/downloads/index.html>. Selezionare JDK e poi la versione del *Java SE Development Kit 8u121* corrispondente al vostro sistema operativo. NON è necessario scaricare ed installare JavaFX, in quanto il pacchetto è già incluso nell'installazione di Java versione 8.
- Scaricare l'ambiente di sviluppo *Eclipse NEON*, dal sito <http://www.eclipse.org/downloads/> scegliendo la versione *Eclipse IDE for Java EE Developers*.
- Dalla barra del menù, selezionare *Help -> Eclipse Marketplace*. Cercare “javafx” ed installare il pacchetto *e(fx)clipse 2.4.0*, che include una serie di plugin utili per sviluppare con JavaFX.
- Scaricare ed installare il tool *JavaFX Scene Builder 8.3.0*, raggiungibile tramite il sito <http://gluonhq.com/open-source/scene-builder/>

03FYZ TECNICHE DI PROGRAMMAZIONE

Esercitazione di Laboratorio 1 - 8 Marzo 2017

Obiettivi dell'esercitazione:

- Riprendere i concetti di sviluppo di applicazioni Java, già affrontati nel corso di Programmazione a Oggetti
 - Scaricare ed importare progetti Eclipse
 - Integrare le proprie classi in progetti esistenti
-

Esercizio 1

Importare il progetto previsto per questa esercitazione scaricando l'archivio zip dal seguente link:
<https://github.com/TdP-2017/Lab01>

Lo scopo del programma (Figura 1) è quello di permettere all'utente di:

- Inserire delle parole in una struttura dati (tramite il bottone *Inserisci*).
- Visualizzare le parole inserite nell'area di testo sottostante in **ordine alfabetico**.
- Permettere la cancellazione di tutte le parole tramite il bottone *Reset*.

A tal fine, si richiede di completare la classe *Parole* presente nel package *it.polito.tdp.parole.model* affinché il programma possa funzionare. La classe *Parole* definisce la struttura dati utilizzata dall'applicazione e offre i seguenti metodi (da implementare) sfruttati dall'interfaccia grafica:

- **public void** *addParola*(String p): consente di inserire una parola nella struttura dati.
- **public** List<String> *getElenco*(): ritorna l'elenco di parole correntemente inserite ordinate alfabeticamente.
- **public void** *reset*(): elimina tutte le parole dalla struttura dati.

In particolare, i metodi *addParola(String p)* e *getElenco()* sono richiamati dal bottone *Inserisci*, mentre il metodo *reset()* è richiamato dal bottone *Reset*.



Figura 1: Programma per la prima esercitazione

Esercizio 2

Modificando il file *Parole.fxml* e le classi *Parole* e *ParoleController*, integrare il programma con le seguenti funzionalità:

- Aggiungere, di fianco al bottone *Reset*, il bottone *Cancella*, che permette di eliminare dalla struttura dati solamente la parola selezionata nell'area di testo. Implementare l'opportuno metodo aggiuntivo nella classe *Parole*, che dovrà essere richiamato all'evento di pressione del nuovo bottone.
- Aggiungere, sotto all'area di testo per la visualizzazione delle parole, un'area di testo per visualizzare i tempi di esecuzione di ciascuna operazione eseguita dal programma (utilizzare la funzione *System.nanoTime()*).
- Provare il programma utilizzando come struttura dati per la classe *Parole* sia una *LinkedList* che un *ArrayList*. Ci sono differenze nelle prestazioni?