

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 *Java SE 8u201/ 8u202* e poi *JDK Download*, ed infine la versione di *Java SE Development Kit 8u202* per il 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 Oxygen* dal sito <http://www.eclipse.org/downloads/>. Si suggerisce di selezionare *Get Eclipse Oxygen* e scegliere di installare *Eclipse IDE for Java Developers* (oppure *Eclipse IDE for Java EE Developers* se si intende sviluppare anche applicazioni Web con Java -- non necessario in questo corso).
- Dalla barra del menù di Eclipse, selezionare *Help -> Eclipse Marketplace*. Cercare "javafx" ed installare il pacchetto *e(fx)clipse 3.1.3*, che include una serie di plugin utili per sviluppare con JavaFX.
- Scaricare ed installare il tool *JavaFX Scene Builder*, raggiungibile tramite il sito <http://gluonhq.com/open-source/scene-builder/>

03FYZ TECNICHE DI PROGRAMMAZIONE

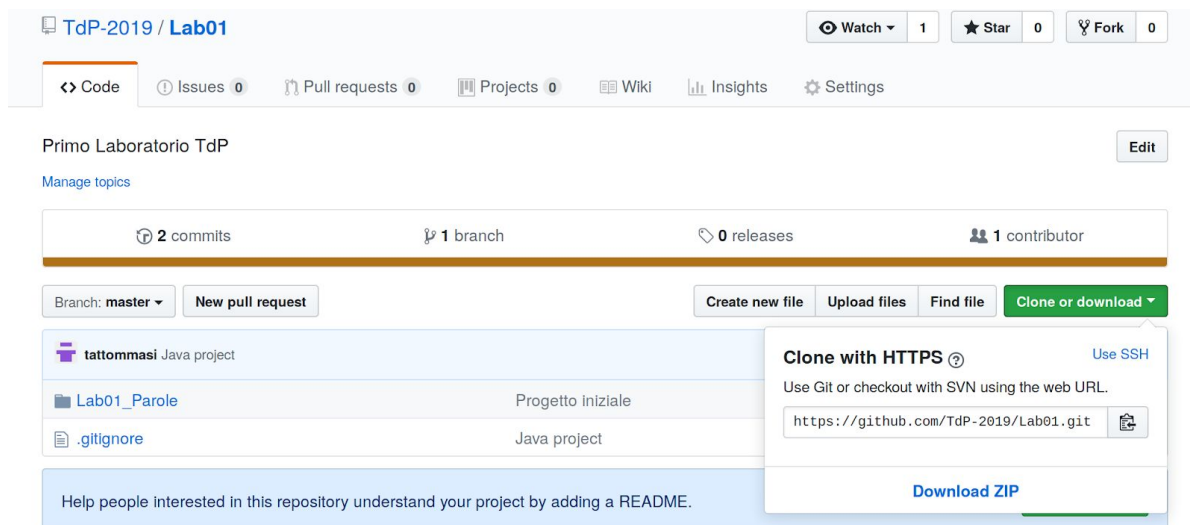
Esercitazione di Laboratorio, 6 Marzo 2019

Obiettivi dell'esercitazione:

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

Esercizio 1

Importare in Eclipse il progetto previsto per questa esercitazione scaricando l'archivio zip dal link: <https://github.com/TdP-2019/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`, e di utilizzare i metodi di tale classe a partire dal controller dell'applicazione (`ParoleController`). La classe `Parole`, definisce la struttura dati utilizzata dall'applicazione e offre i seguenti metodi (da implementare):

public void `addParola(String p)`: consente di inserire una parola nella struttura dati

public `List<String> getElenco()`: ritorna l'elenco di parole correnti inserite ordinate alfabeticamente

public void `reset()`: elimina tutte le parole dalla struttura dati.

I metodi *addParola(String p)* e *getElenco()* dovranno essere richiamati nell'event handler associato al bottone *Inserisci* (metodo *doInsert* della classe *ParoleController*), mentre il metodo *reset()* dovrà essere richiamato dall'event handler del bottone *Reset* (metodo *doReset*).



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:

1. 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 bottone *Cancella*.
2. Aggiungere, sotto all'area di testo per la visualizzazione delle parole, un'ulteriore area di testo per visualizzare i tempi di esecuzione di ciascuna operazione eseguita dal programma (utilizzare la funzione *System.nanoTime()*).
3. Creare due versioni della classe *Parole*, basandole rispettivamente su una *LinkedList* e un *ArrayList*. Sperimentare con le due versioni del programma: ci sono differenze nelle prestazioni?