

Manuale d'uso e installazione

Indice

1. INSTALLAZIONE & AVVIAMENTO	2
2. COME USARE L'INTERFACCIA UTENTE	2
3. DATABASE	4

1. INSTALLAZIONE & AVVIAMENTO

Per installare il progetto “PurchaseOrders” basta avere installato o installare un’ambiente di sviluppo integrato che permetta l’utilizzo di codice Java. Consiglio: usare Eclipse, lo stesso ambiente in cui il progetto è stato realizzato.

Per completare l’installazione bisognerà importare il progetto all’interno dell’ambiente e controllare che il Database denominato “DataBase.accdb” sia in locale come il progetto. Se questo non fosse vero, basta cercare il Database e portarlo all’interno del progetto.

Per avviarlo basterà effettuare il “Run”. Se non si avvia, bisogna fare un click destro sopra alla classe “Interfaccia.java” all’interno del progetto situata nel package “interfaccia”. Si aprirà un menù. Selezionare “Run as” e poi “Java application”. Il progetto si avvierà.

Lo scopo del progetto è quello di dare in input dei dati che rappresentano una merce e la loro quantità e ricevere in output, in ordine, i migliori fornitori che hanno a disposizione e vendono i dati messi in input.

2. COME USARE L’INTERFACCIA UTENTE

La figura 2.1 mostra come si presenta l’interfaccia utente.

Per far funzionare l’applicazione c’è bisogno di:

- Scrivere nella casella di testo vicino alla voce “Inserire merce:” il nome della merce da voler cercare.
- Scrivere nella casella di testo vicino alla voce “Inserire quantità:” il numero della merce da voler cercare.

Una volta inseriti questi due parametri si dovrà schiacciare il tasto “Cerca”.

Apparirà una nuova finestra che mostrerà i risultati ottenuti come quella della figura 2.2.

Per cancellare i dati inseriti nelle caselle di testo in maniera rapida bisognerà premere il tasto “Reset”.

I risultati mostrati nella figura 2.2 variano a causa di due fattori:

- Input.
- Database.

Per sapere come è formattato il Database e come modificarlo bisogna andare al capitolo 3.

Per sapere da quali input si ottengono determinati output, guardare l’uml dei casi d’uso del tasto “Cerca” presente nella prossima pagina.

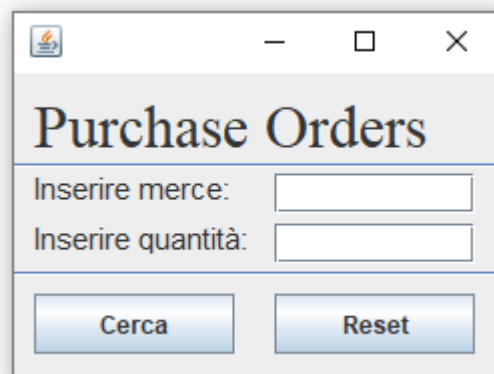


Figura 2.1

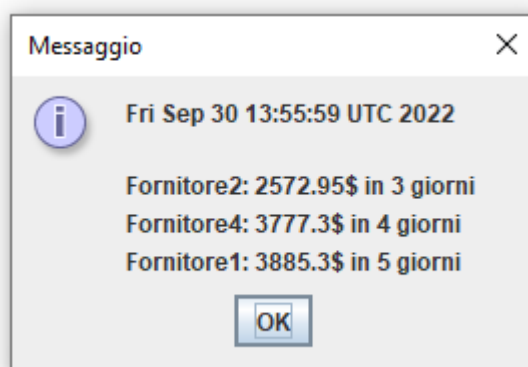


Figura 2.2

UML DEI CASI D'USO	
Nome	Cerca
Attore	Utente che schiaccia il tasto "Cerca"
Scenario Principale	<ol style="list-style-type: none"> 1. L'utente inserisce il nome della merce da cercare. 2. L'utente inserisce il numero di quantità della merce da cercare. 3. L'utente preme "Cerca". 4. L'app mostra a video i risultati. Prima la data, poi in ordine, dal meno caro al più caro, i vari fornitori e il totale che si dovrebbe spendere da ognuno di loro per acquisire la merce. Il totale comprende i possibili sconti. A fianco di queste informazioni sono riportati i giorni di spedizione. In caso di totale uguale tra più fornitori, si mostra prima il fornitore con tempo di spedizione minore. 5. L'utente preme "ok" per terminare la ricerca. 6. L'utente chiude l'app. <p>Fine.</p>
Scenario Alternativo	<p>6a. L'utente inserisce nuovi dati. Torna al punto 3 dello Scenario Principale.</p>
Scenario Alternativo	<p>4a. Precondizione: il nome della merce non può essere vuoto. "Inserito nessun nome merce!" apparirà al posto delle informazioni finali. Torna al punto 5 dello Scenario Principale.</p>
Scenario Alternativo	<p>4a. Precondizione: la quantità inserita deve essere un intero. "Inserito numero non valido!" apparirà al posto delle informazioni finali. Torna al punto 5 dello Scenario Principale.</p>
Scenario Alternativo	<p>4a. Precondizione: almeno un fornitore deve avere la merce desiderata. "Nessun fornitore ha la quantità desiderata della merce inserita!" apparirà al posto delle informazioni finali. Torna al punto 5 dello Scenario Principale.</p>
Scenario Alternativo	<p>4a. Precondizione: almeno un fornitore deve avere almeno la quantità inserita della merce desiderata. "Nessun fornitore ha la quantità desiderata della merce inserita!" apparirà al posto delle informazioni finali. Torna al punto 5 dello Scenario Principale.</p>
Scenario Alternativo	<p>4a. Precondizione: Il database deve essere strutturato come descritto nel prossimo paragrafo. "Nessun fornitore ha la quantità desiderata della merce inserita!" apparirà al posto delle informazioni finali. In Eclipse si vedrà l'errore nella struttura del Database. Torna al punto 5 dello Scenario Principale.</p>

UML DEI CASI D'USO	
Nome	Reset
Attore	Utente che schiaccia il tasto "Reset"
Scenario Principale	<ol style="list-style-type: none"> 1. L'utente preme "Reset". 2. Le caselle di testo si svuotano. <p>Fine.</p>

3. DATABASE

Per costruire il Database è stato utilizzato il software “Access” del pacchetto di Microsoft.

Ogni tabella del database deve avere il nome composto da una stringa “Fornitore” e continuare l’enumerazione presente senza spazi.

Esempio: Al momento sono presenti i fornitori dal 1 al 4, la prossima tabella si dovrà denominare “Fornitore5”, quello dopo “Fornitore6” e così via.

Ricordare: Se si inserisce una nuova tabella bisognerà andare nella classe “Importa.java” e inserire il numero della nuova tabella nella costante “ULTIMA_TABELLA”. Se non si rispetta questo procedimento; i calcoli verranno effettuati solo sui fornitori già esistenti in precedenza.

Le tabelle sono strutturate nel seguente modo:

1. Fornitore#_ID. Dove # rappresenta il numero della tabella. Numero del record.
2. Nome. Chiave primaria. Nome merce.
3. Quantità. Numero comprabile di tale merce.
4. Prezzo. Prezzo della merce singola.
5. Sconto. Sconto che si può applicare al totale.
6. Giorni. Giorni di spedizione per far arrivare la merce.

Se si inseriscono strutture diverse non è possibile garantire il corretto funzionamento dell’app. Avendo cura di riempire tutti i campi si possono inserire quanti record si vuole all’interno di ogni tabella.

Per inserire gli sconti c’è bisogno di sapere la loro sintassi:

- “/”. Serve per dire che non ci sono sconti sulla merce del record.
- “+” “NUMERO” “%” “NUMERO” “!”. Scritto senza spazi e inseriti i numeri dove è presente “NUMERO” è possibile andare a inserire uno sconto basato sul totale che si va a spendere. Il primo “NUMERO” rappresenta la somma da spendere per avere lo sconto, il secondo rappresenta lo sconto che si viene ad applicare (da 0 a 100).
- “-” “NUMERO” “%” “NUMERO” “!”. Scritto senza spazi e inseriti i numeri dove è presente “NUMERO” è possibile andare a inserire uno sconto basato sul totale della merce che si va a vendere. Il primo “NUMERO” rappresenta il numero di merce da prendere per avere lo sconto, il secondo rappresenta lo sconto che si viene ad applicare (da 0 a 100).
- “\$” “NUMERO” “!”. Scritto senza spazi e inserito il numero dove è presente “NUMERO” è possibile andare a inserire uno sconto manualmente sulla merce di quel record. Lo sconto è rappresentato dal numero che si va ad inserire (da 0 a 100).

In caso di valori al di fuori del range degli sconti, lo sconto non viene considerato per il calcolo del totale di quella merce. Se si inseriscono valori diversi dai simboli iniziali sopra elencati oppure si descrive male un campo sconto di una merce, la merce verrà considerata senza sconto quindi a prezzo pieno.

Dopo la colonna “Giorni” è possibile andare a inserire tutte le colonne che si vogliono senza andare ad interferire con l’app. Qualora si inseriscano colonne che permettono di inserire dati che andrebbero a modificare i prezzi o gli sconti o la quantità della merce vendibile bisognerà andare ad aggiornare l’app. Se non avviene questo aggiornamento, l’app continuerà a funzionare come in precedenza ignorando questi nuovi dati.