**Documentazione “Python3App”**

Per richiamare da linea di comando l’applicazione, digitare:

*java -jar [path di Python3App.jar] [-args]*

In alternativa, se si hanno a disposizione solo i file .class, digitare:

*java [path di Python3App.class] [-args]*

Qui sotto le opzioni disponibili, ciascuna corrispondente a un diverso caso d’uso:

1. ***-c***
2. ***-w***
3. ***-r***
4. ***-v***

***Caso d’uso 1: opzione “calcolo”***

Con questa opzione è possibile leggere un file .gpx e calcolare i dati salienti dell’escursione:

[-args]: -c [stringa:path del file gpx]

*Esempio*: -c “C:\TraceGpx\gita.gpx”

I dati calcolati non sono trascritti su database.

***Caso d’uso 2: opzione “scrivi su db”***

Con questa opzione è possibile leggere un file .gpx, calcolare i dati salienti dell’escursione e serializzarli su database:

[-args]: -w [stringa:path del file gpx] [stringa:nome di chi carica i dati]

*Esempio*: -w “C:\TraceGpx\gita.gpx” “Pinco Pallino”

***Caso d’uso 3: opzione “leggi su db****”*

Con questa opzione è possibile leggere i dati salvati da db.

Ci sono due sottocasi:

[-args]: -r

[-args]: -r [stringa:path file gpx] [stringa:timestamp gita] [stringa: chi ha caricato i dati]

Nel primo sottocaso, vengono stampate a video tutte le escursioni presenti nel database.

Nel secondo sottocaso, viene stampata una escursione specifica, identificata dalle sue coordinate.

*Esempio*: -r “C:\TraceGpx\gita.gpx” “2021-02-27 06:17:39” “Pinco Pallino”

Successivamente alla visualizzazione dei dati, viene chiesto all’utente se vuole stampare i dati su file:

se così si vuole, digitare “s”,”S”,”y”,”Y”.

Viene quindi richiesta la path dove scrivere il file.

Il formato di default del file è CSV, ma è possibile personalizzare il delimitatore dei campi. Allo scopo viene chiesto all’utente se desidera usare un separatore diverso dalla virgola.

***Caso d’uso 4: opzione “manuale d’uso”***

Con questa opzione è possibile leggere un sunto di questo manuale:

[-args]: -v