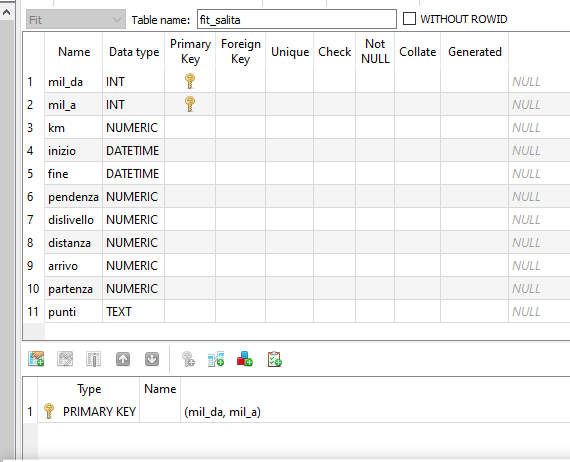
Visualizzazione salite

Vengono visualizzate le salite da un milestone di partenza al successivo inteso come punto di partenza e arrivo.

Archivio salite:



L’archivio contiene i dati di identificazione,

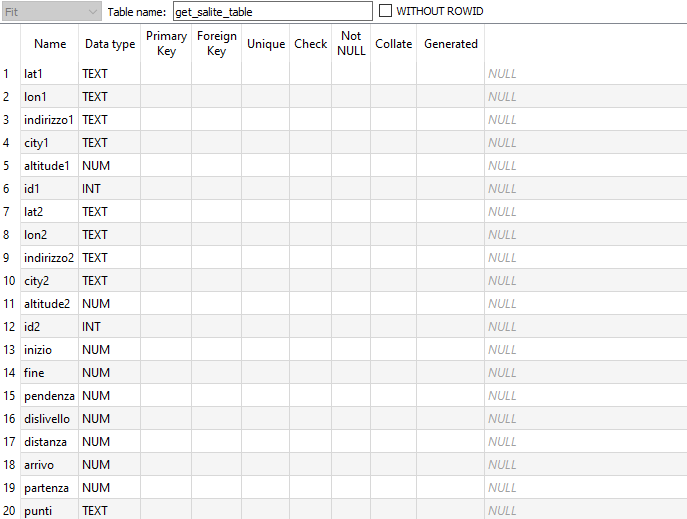
I dati di riepilogo

I punti in formato JSON per visualizzare il grafico altimetrico della salita.

# Visualizzazione di tutte le salite ( Punto di arrivo e descrizione )

* async get\_milestones()

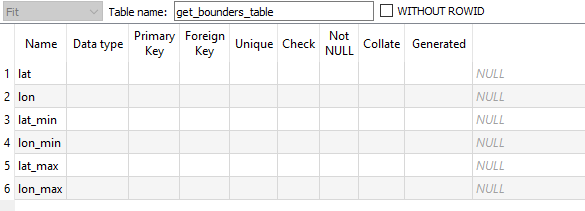
La funzione viene richiamata senza parametri.Vengono visualizzate tutte le salite codificate.

* fnc\_wait(divParm) funzione di visualizzazione pannello di attesa (bike) nel div indicato come parametro.
* hidden\_div() funzione di visualizzazione dei div ( arrayNascondi,arrayVisualizza).
  + “div\_menu1” – nascosto
  + “div\_visualizzazione” - visualizzato
  + get\_sqlsalite() impostazione ed esecuzione con load(pr) della lettura di tutti le uscite i dati presenti sul file salite.
  + 
  + Crea\_layers(salite\_rows) creazione dei CircleMarker e dei tooltip
    - cm(el) – Creazione punto di arrivo
    - toolTipSalita(el record di salite\_rows ) con variabile 1 per punto di arrivo e 2 per punto di partenza viene visualizzato solo il tooltip di arrivo

Vengono generati 4 Gruppi Layes per la visualizzazione sopra o sotto i 600 metri dei CircleMarker e dei tooltip

# Seleziona salita – generazione popup

Elaborazione dei gruppi Layers:

* await initMap() impostazione Map e pulizia eventuali precedenti
* await get\_bounders() selezione i limiti di Latitudine e Longitudine di tutti i milestones presenti in tutte le salite codificate.
* 
* Impostazione\_limiti\_mappa(bounder\_rows) impostazione limiti sulla map
* Settaggio map e aggiunta layer (tileMap)

Vengono letti tutti i Circle Marker creati e aggiunta la routine di generazione popup per punto di partenza.

Nella routine generata, viene selezionato l’itinerario e creata la popup di visualizzazione.

ChartPopup\_itinerario() visualizzazione itinerari. Selezionando un itinerario selectChartItinerario(), vengono generati i dati per la popup:

* Aggiorna\_elevation\_salita(). Verifica che esista l’altitudine in tutti i punti della salita.
* Get\_dati\_salita() recupero dei dati da file fit\_itinerario\_w per i dati di riassunto delle

Uscite per la salita dc.dati\_tabella.

* Se l’elemento uscita non contiene i dati di dettaglio, eseguo sul server SQL\_NODE

E imposto i dati della salita

* routingCreate – imposta l’itinerario scelto
* Chart\_table\_salita()
* Chart\_dislivello()

* + hideOverlappingTooltips() cancellazione Tool Tip che si sovrappongono
  + lettura di tutti i circle marker e aggiunta della popup al click per la selezione della salita nel punto arrivo