

Compositore di stampa

- nuovo compositore di stampa
- proprietà della pagina
- oggetto finestra di mappa / proprietà della finestra di mappa
- Aggiungere una basemap XYZ:
Select language...
 - `http://tiles.wmflabs.org/bw-mapnik/{z}/{x}/{y}.png`
- ...
- oggetto legenda / proprietà della legenda
- oggetto indicatore di scala
- testi ed altri oggetti
- inserire un indicatore di orientamento ed orientarlo con la mappa
- Aggiungere una pagina
- Aggiungere un riferimento di posizione nella mappa (indicatore)
- inserire una tabella di attributi
- stampa
- Layout Manager (template di stampa)

Atlante base

- creare un layer di copertura con processing (crea reticolo)
- configurare il compositore come atlante
- configurare l'oggetto mappa per produrre un'atlante
- configurare una keymap con l'indicatore di posizione
- configurare un testo dinamico dell'atlante con le espressioni

Atlante intermedio

- Configurare un atlante per singolo oggetto.
- Configurare le scale di rappresentazione dinamiche
- Evidenziare l'oggetto rappresentato dinamicamente dall'atlante
- inserire una tabella relativa al contesto visualizzato
- Inserire Testo dinamico relativo all'oggetto dell'atlante

Atlante avanzato

- Aggiungere un'immagine dell'oggetto corrente tramite google streetview
- Analizziamo l'api per la visualizzazione di immagini statiche di Google streetview:
<https://developers.google.com/maps/documentation/streetview/intro>
- api key valida per workshop: AlzaSyCFmb_amfW4BZICpyNgtjLRmkXD9qADDCE
- Estrarre le coordinate del punto di vista e la direzione usando il layer roads
- definire un subset degli edifici (performance)
- definiamo un virtual layer:
 - includere osm_roads come "strade" e includere il subset degli edifici come "edifici"

```

o select edifici.fid, st_shortestline(st_centroid(edifici.geometry), strade.geometry)
   from edifici, strade
   group by edifici.fid
   having min (st_distance(st_centroid(edifici.geometry), strade.geometry))

```

- provare se il virtual layer funziona ed inseriamolo
- calcoliamo i campi heading, latitude e longitude come expression:
 - latitude
 - `y(end_point($geometry))`
 - longitude
 - `x(end_point($geometry))`
 - heading
 - `azimuth(end_point($geometry),start_point($geometry))*180/pi()`
- attacchiamo il virtual layer al layer edifici come join ed importiamo heading, latitude e longitude
- inseriamo un oggetto testo come html
- costruiamo l'URL in modo dinamico

```

o 

```

- configurando le dimensioni della finestra html viene visualizzata l'immagine dell'edificio in modo opportuno
- Aggiungiamo un indicatore del punto di ripresa:
 - alla simbologia della selezione di edifici aggiungiamo un geometry generator con simbolo con tipo di geometria PUNTO/MULTIPUNTO con la seguente stringa di generazione (è un'espressione di qgis):
 - `end_point(make_point("x" , "y"))`
 - Aggiustiamo il simbolo e la rotazione con un'espressione ed il campo "Heading"
 - Aggiorniamo la mappa