

# REPORT xmlAnalyzer.py

xmlAnalyzer.py è uno script utile , dato un file fcd\_dump.xml ,per generare un output sui movimenti di un pedone.

Corretto il calcolo della cella corrente per ogni dato . Non viene generata una griglia fissa come in **xmlGridstats.py** ma dato che abbiamo delle variazioni a tempo , viene calcolata la cella corrente di ogni id e confrontato con i dati del dizionario sappiamo se il pedone ha cambiato cella e quante celle ha visitato della stessa dimensione.

```
celle_visit_person = {}

for person in root.findall('.//person'):
    id_person = str(person.attrib['id'])
    x = float(person.attrib['x'])
    y = float(person.attrib['y'])

    # determina la cella corrente in base alle coordinate x e y
    cella_x = dimensione_cella* math.floor(x / dimensione_cella)
    cella_y = dimensione_cella* math.floor(y / dimensione_cella)
    cella_x2 = cella_x+dimensione_cella
    cella_y2 = cella_y+dimensione_cella
    cella_corrente = (cella_x, cella_y, cella_x2, cella_y2)

    if id_person not in celle_visit_person:
        celle_visit_person[id_person] = {
            'celle_visit': {cella_corrente},
            'num_cambi_cella': 0
        }
    else:
        celle_visit_person[id_person]['celle_visit'].add(cella_corrente)
        # verifica se la cella corrente è diversa dall'ultima visitata
        celle_visit = celle_visit_person[id_person]['celle_visit']
        if len(celle_visit) > 1:
            ultima_cella = list(celle_visit)[-2]
```

Anche in questo codice la cella è ora definita da quattro punti e il calcolo della cella corrente del pedone è fatto allo stesso modo del codice xmlGridstats.py.

## **OUTPUT**

Per comodità anche l'output di questo script è salvato in un file csv tramite la libreria pandas.

Il file README.md contiene le informazioni su come poter utilizzare i due script, consultabile sul [Github](#).