# **Etivity 3 Davide De Angelis**

## **Normalizzazione:**

Nel contesto dello schema concettuale, il processo di normalizzazione non risulta applicabile in quanto non si riscontrano ridondanze significative nei dati. La struttura attuale delle entità e delle loro relazioni è già conforme ai principi della modellazione relazionale ottimizzata, garantendo coerenza e minimizzando la duplicazione delle informazioni, pertanto, non vi è la necessità di ulteriori decomposizioni o ristrutturazioni per il raggiungimento delle forme normali.

## **CRUD:**

#### CREATE

meta = MetaData()  
  
  
  
facolta = Table( 'facolta', meta, Column ('idfacolta', Integer, primary\_key = True, autoincrement = True), Column ('Nome', String(100)), Column('Sede', String(255)), )  
  
  
  
meta.create\_all(engine)

#### READ

connSelect = engine.connect()  
  
selezione = facolta.select()  
  
selezionex = "facolta.select()"  
  
risultatoSelect = connSelect.execute(selezione)  
  
riga = risultatoSelect.fetchall()  
  
rigaex = "risultatoSelect.fetchall()"  
  
print(riga)  
  
print('\v')  
  
time.sleep(1)  
  
print(f'Risultato della query {selezionex} utilizzando {rigaex}')

#### UPDATE

conn4 = engine.connect()  
  
updateRecordFacolta = facolta.update().values({facolta.c.Sede:'Via del Tritone 75'}).\  
where(facolta.c.idfacolta == 15)  
  
  
conn4.execute(updateRecordFacolta)  
conn4.commit()  
  
print('Record aggiornati correttamente' )

#### DELETE

conn2 = engine.connect()  
  
eliminaRecordFacolta = facolta.delete().where(facolta.c.idfacolta == 15)  
  
conn2.execute(eliminaRecordFacolta)  
conn2.commit()  
  
print('Record eliminati' )  
  
time.sleep(3)  
  
print('\v')

## **Strategia di progettazione utilizzata:**

La strategia di progettazione ha previsto la presentazione di uno schema concettuale già ottimizzato, senza necessità di ulteriori normalizzazioni, e implementa operazioni CRUD efficienti per la gestione del database.