

PROVA 2 - QUESITO n. 1

Considerare il seguente database relazionale di una porzione di archivio documentale:

| Tabella: DOCUMENTI | | |
|--------------------|-------------|-------------------------------------|
| <i>campo</i> | <i>tipo</i> | <i>commento</i> |
| ID_DOC | INTEGER | codice univoco del documento |
| ANNO_DOC | INTEGER | anno di registrazione |
| NUM_DOC | INTEGER | numero di registrazione |
| DATA_DOC | DATE | data di registrazione |
| OGGETTO_DOC | VARCHAR2 | oggetto del documento |
| TIPO_DOC | INTEGER | codice della tipologia di documento |

| Tabella: PRATICA | | |
|------------------|-------------|--|
| <i>campo</i> | <i>tipo</i> | <i>commento</i> |
| ID_PRATICA | INTEGER | codice univoco della pratica/fascicolo |
| ANNO_PRA | INTEGER | anno della pratica/fascicolo |
| NUM_PRA | INTEGER | numero della pratica/fascicolo |
| OGGETTO_PRA | VARCHAR2 | oggetto della pratica |
| CLASS_PRA | INTEGER | codice della classificazione della pratica |
| UTE_RESP | VARCHAR2 | Nominativo responsabile della pratica |
| ID_DOC_APERTURA | INTEGER | codice del documento che ha determinato l'apertura della pratica |

| Tabella: ATTIVITA_PRATICA | | |
|---------------------------|-------------|---|
| <i>campo</i> | <i>tipo</i> | <i>commento</i> |
| ID_PRATICA | INTEGER | codice univoco della pratica/fascicolo |
| NUM_ATTIVITA | INTEGER | numero dell'attività della pratica |
| DATA_ATTIVITA | DATE | data in cui so è svolta l'attività della pratica |
| ID_DOC_ATTIVITA | INTEGER | codice del documento collegato all'attività della pratica |
| NOTE_ATTIVITA | VARCHAR2 | note/descrizione dell'attività svolta |

- A) scrivere il comando SQL per inserire nella pratica 999/2019 (già presente) un'attività per ogni documento ancora non gestito in nessuna pratica/attività [la data dell'attività deve corrispondere alla data del documento, così come la descrizione dell'attività l'oggetto del documento. Il numero attività deve essere un numero progressivo congruente con la data dell'attività. Il candidato deve descrivere eventuali assunzioni]
- B) scrivere una procedura PL/SQL che riceva in input una tipologia di documento e permetta di conoscere quante pratiche negli ultimi 3 anni hanno avuto origine da un documento della tipologia in input, quante attività negli ultimi 3 anni hanno avuto origine da un documento della tipologia in input, quanti documenti della tipologia indicata non hanno dato origine nè a una pratica nè a una attività. (la tecnica di restituzione dell'informazione di output è a scelta del candidato)

PROVA 2 - QUESITO n. 2

Un utente con buone conoscenze di Excel ha sviluppato un proprio sistema di registrazione di accessi alla struttura che andrebbe messo in sicurezza e condiviso con diversi utenti aziendali.

- Descrivere schematicamente una o più possibili modalità di sviluppo di un prodotto idoneo per l'uso in multiutenza e in completa sicurezza, utilizzando sistemi cloud (SaaS) e/o basi dati centralizzate, descrivendo punti di forza e di debolezza delle varie soluzioni.
- Descrivere anche alcune possibili implementazioni per realizzare un servizio di pubblicazione su Internet dei dati, tenendo disaccoppiato e in sicurezza il DB interno. Accennare alle precauzioni da adottare per salvaguardare la disponibilità del servizio e tutelare la protezione di eventuali dati personali.

PROVA 2 - QUESITO n. 3

Un utente lamenta grossi problemi di performance su un sistema software web-based, scritto in .NET, application server IIS e database server Oracle. I server sono attestati in una server farm virtualizzata (tecnologia VmWare).

Descrivere le azioni e le verifiche che si ritengono necessarie per poter individuare il problema e i relativi tentativi di risoluzione.

PROVA 2 - QUESITO n. 4

Dato il seguente programma Python

```
import random, unittest
from flask import Flask, jsonify

app = Flask(__name__)
numbers_list = random.sample(range(0, 100), 100)

@app.route("/numberlist/<startindex>/<endindex>/json")

def numberlist(startindex, endindex):
    a = int(startindex)
    b = int(endindex)
    return jsonify({
        "numbers": numbers_list[a:b]
    })

class TestApp(unittest.TestCase):

    def test_numberlist_service(self):

        client = app.test_client()

        response = client.get("/numberlist/1/10/json")

        content = response.data
        raise NotImplementedError()

if __name__ == '__main__':
    print(unittest.main())
```

Rispondere alle seguenti domande:

1. Descrivere cosa fa il programma. Spiegare cosa significa la penultima riga del codice.
2. Descrivere il framework Flask e le differenze fondamentali rispetto al framework Django. Esporre brevemente due scenari: uno in cui si reputi più appropriato usare Flask ed uno in cui si reputi più appropriato usare Django.
3. Cosa fa la funzione “numberlist”? Quali errori si potrebbero verificare all'interno della funzione? Cosa significa il simbolo “@” nella riga subito prima?
4. Descrivere una possibile implementazione del metodo “test_numberlist_service”. Descrivere cosa si intende per Test-Driven Development (TDD), quali sono i suoi vantaggi e uno o più moduli Python che si possono utilizzare per i test.
5. Poniamo che su un server Linux (preferibilmente, distribuzione basata su RedHat o Debian) si riscontrino problemi di rete sull'accesso al programma e si renda necessaria un'analisi sul server che lo ospita. Si descrivano gli strumenti che si potrebbero utilizzare per verificare la porta su cui è in ascolto il programma, per un'analisi del traffico o per generiche valutazioni sul carico di rete.