

Keep Sitting with YOUR EX04 PARTNER at your Assigned Tables!

Modern Tools Discussion and RDBMS 101

Warm-up: Code Review COMP110 Student Code

A COMP110 student passes the autograder with the function below.

How would your table Code Review it?

```
def is_even(num):  
    while num % 2 != 0:  
        return False  
    return True
```

```
# Example usage  
print(is_even(4))  
print(is_even(5))
```

Write down your table's responses and submit as a CL in groups of up to 4.

You post your CR to the student and the question, "why did you choose a while loop here?" This is the student's response:

"This design succinctly leverages the Python control flow constructs to provide an implicit gate that filters out odd numbers, making the code highly readable and efficient by minimizing the number of operations performed. The approach elegantly demonstrates how logical operators and control flow can be combined to express complex logic in a concise manner, reflecting a deep understanding of both Python syntax and computational logic."

Write down your table's response to this student.

Table

- Collection of Columns
- Columns have names, types, and constraints
- Every table has a PRIMARY KEY
- Foreign keys create a relationship with another table's Primary K

account

id	name	balance
1	Spongebob	110.0
2	Patrick	50.0

activity

id	account_id	action	amount
1	1	DEP	110.0
2	2	DEP	50.0

- Given these two tables, in psuedo-code-y hand-waving English, what *mutations* would be required for SpongeBob to transfer Patrick \$10
- Write down the sequence of operations in terms of (VERB T#PK) DATA
 - VERB T#PK is "INSERT" (row) or "UPDATE" (row) with Table#Primary KeyID
 - DATA contains relevant columns/values
 - Eg: (INSERT account#3) account_id: 2, action: DEP, amount: 10.0
- Hint: It should take 4x Operations

TRANSACTIONS

A - atomicity
- "all or nothing"

C - consistency

I - isolation

D - Durability

What concerns exist when going between DB system and our Application Server (an OOP system)?