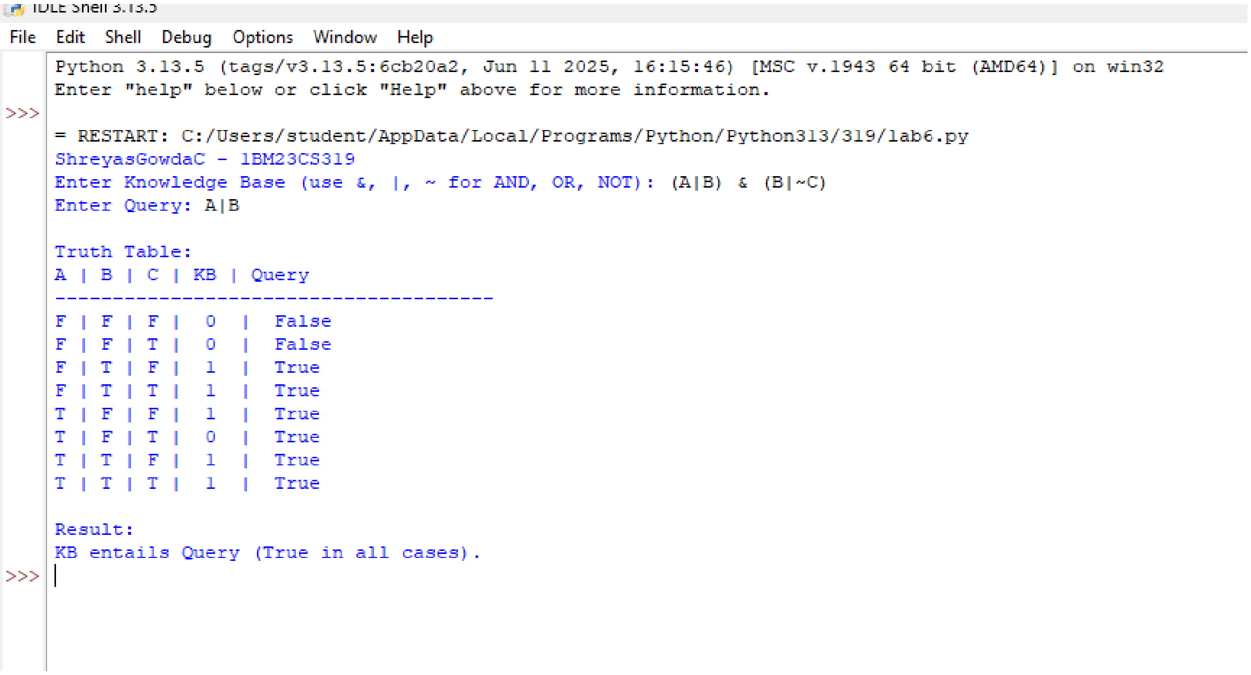
Propositional Logic

CODE:-

```
print("ShreyasGowdaC - 1BM23CS319")
import itertools

def eval_expr(expr, model):
    try:
        return eval(expr, {}, model)
    except:
        return False

def tt_entails(KB, query):
    symbols = sorted(set([ch for ch in KB + query if ch.isalpha()]))

    print("\nTruth Table:")
    print(" | ".join(symbols) + " | KB | Query")
    print("-" * (6 * len(symbols) + 20))

    entails = True
    for values in itertools.product([False, True], repeat=len(symbols)):
        model = dict(zip(symbols, values))
        kb_val = eval_expr(KB, model)
        query_val = eval_expr(query, model)

        row = " | ".join(["T" if model[s] else "F" for s in symbols])
        print(f"{row} |  {kb_val}  |  {query_val}")

        if kb_val and not query_val:
            entails = False

    return entails

KB = input("Enter Knowledge Base (use &, |, ~ for AND, OR, NOT): ")
query = input("Enter Query: ")

result = tt_entails(KB, query)

print("\nResult:")
if result:
    print("KB entails Query (True in all cases).")
else:
    print("KB does NOT entail Query.")
```

OUTPUT:-



```
    Python 3.13.5 (tags/v3.13.5:6cb20a2, Jun 11 2025, 16:15:46) [MSC v.1943 64 bit (AMD64)] on win32
    Enter "help" below or click "Help" above for more information.
>>>
    = RESTART: C:/Users/student/AppData/Local/Programs/Python/Python313/319/lab6.py
    ShreyasGowdaC - 1BM23CS319
    Enter Knowledge Base (use &, |, ~ for AND, OR, NOT): (A|B) & (B|~C)
    Enter Query: A|B

    Truth Table:
    A | B | C | KB | Query
    --------------------------------------
    F | F | F |  0  |  False
    F | F | T |  0  |  False
    F | T | F |  1  |  True
    F | T | T |  1  |  True
    T | F | F |  1  |  True
    T | F | T |  0  |  True
    T | T | F |  1  |  True
    T | T | T |  1  |  True

    Result:
    KB entails Query (True in all cases).
>>> |
```