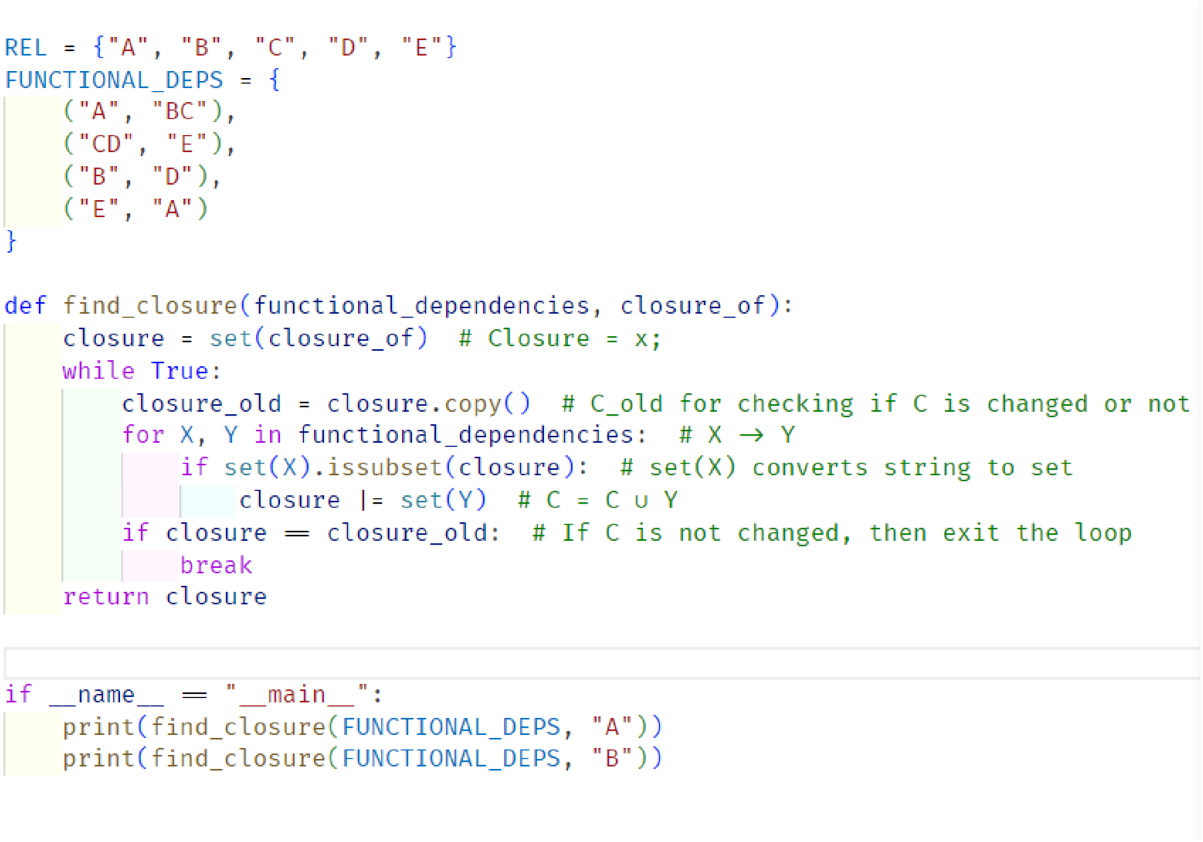
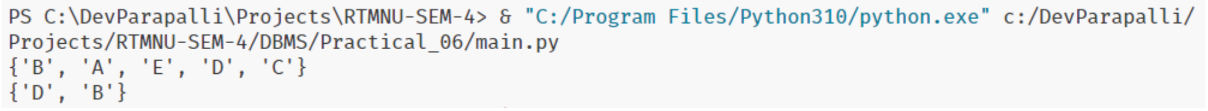


```python
REL = {"A", "B", "C", "D", "E"}
FUNCTIONAL_DEPS = {
    ("A", "BC"),
    ("CD", "E"),
    ("B", "D"),
    ("E", "A")
}

def find_closure(functional_dependencies, closure_of):
    closure = set(closure_of)  # Closure = x;
    while True:
        closure_old = closure.copy()  # C_old for checking if C is changed or not
        for X, Y in functional_dependencies:  # X -> Y
            if set(X).issubset(closure):  # set(X) converts string to set
                closure |= set(Y)  # C = C ∪ Y
        if closure == closure_old:  # If C is not changed, then exit the loop
            break
    return closure


if __name__ == "__main__":
    print(find_closure(FUNCTIONAL_DEPS, "A"))
    print(find_closure(FUNCTIONAL_DEPS, "B"))
```

OUTPUT



```
PS C:\DevParapalli\Projects\RTMNU-SEM-4> & "C:/Program Files/Python310/python.exe" c:/DevParapalli/
Projects/RTMNU-SEM-4/DBMS/Practical_06/main.py
{'B', 'A', 'E', 'D', 'C'}
{'D', 'B'}
```