```python
#SLOT MACHINE GAME

import time
import random
import datetime

MAX_LINES = 3
MAX_BET = 100
MIN_BET = 1
ROWS = 3
COLS = 3

symbol_count = {
    "A" : 3,
    "B" : 4,
    "C" : 4,
}

symbol_values = {
    "A" : 10,
    "B" : 40,
    "C" : 30,
}

def check_winnings(columns, lines, bet, values):
    winnings = 0
    winning_lines = []

    for line in range(lines):
        symbol = columns[0][line]
        for column in columns:
            symbol_to_check = column[line]
            if symbol != symbol_to_check:
                break
```

```
        else:
            winnings += values[symbol] * bet
            winning_lines.append(line + 1)

    return winnings, winning_lines

def get_slot_machine_spin(rows, cols, symbols):
    all_symbols = []

    for symbol, symbol_count in symbols.items():
        for _ in range(symbol_count):
            all_symbols.append(symbol)

    columns = []

    for _ in range(cols):
        column = []
        current_symbols = all_symbols[:]
        for _ in range(rows):
            value = random.choice(current_symbols)
            current_symbols.remove(value)
            column.append(value)

        columns.append(column)

    return columns

def print_slot_machine(columns):
    print("\n")
    for row in range(len(columns[0])):
        for i, column in enumerate(columns):
            if i != len(columns) - 1:
                print(column[row], end =" | ")
            else:
```

```
                print(column[row], end = "")
        print()

    time.sleep(1)


def deposit():
    global smount #global bana de isko
    while True:
        amount = input("What would you like to
deposit?: $")

        if amount.isdigit():
            smount = int(amount)  # Assigning the
input value to smount
            if smount > 0:
                break
            else:
                print("Amount must be greater than
0.")
        else:
            print("Please enter a number.")

    return smount

def get_number_of_lines():
    while True:
        lines = input("Enter the number of lines you
want to bet on: (1-" + str(MAX_LINES) + ")?: ")

        if lines.isdigit():
            lines = int(lines)
            if 1 <= lines <= MAX_LINES:
                break
```

```
            else:
                print("Enter a valid number of lines")
        else:
            print("Please enter a number.")

    return lines

def get_bet():
    global betamount
    while True:
        betamount = input("What would you like to bet
on each line?: $")

        if betamount.isdigit():
            betamount = int(betamount)
            if MIN_BET <= betamount <= MAX_BET:
                break
            else:
                print(f"Amount must be between
${MIN_BET} - ${MAX_BET}.")
        else:
            print("Please enter a number.")

    return betamount

def spin(balance):
    lines = get_number_of_lines()

    while True:
        bet = get_bet()
        total_bet = bet * lines
        if total_bet > balance:
            print(f"Current balance is ${balance}. Not
enough to bet this amount.")
```

```
        else:
            break

    print(f"You are betting ${bet} on {lines} lines.
Total bet is equal to: ${total_bet}")

    slots = get_slot_machine_spin(ROWS, COLS,
symbol_count)
    print_slot_machine(slots)

    winnings, winning_lines = check_winnings(slots,
lines, bet, symbol_values)

    print(f"You won ${winnings}.")
    print(f"You won on lines: ", *winning_lines)

    return winnings - total_bet

def write_game_history(balance_history):
    current_time =
datetime.datetime.now().strftime("%Y-%m-%d %H:%M:%S")



    with open("bill.txt", "w") as file:
        file.write("Game History:\n")
        file.write("Original Balance: $%d\n" %
balance_history[0][0])
        file.write("Number of spins: %d\n" %
(len(balance_history) - 1))
        file.write("Time: %s\n" % current_time)  # Add
current time
        file.write("----------------------------------
------------------------------\n")
```

```
        file.write("Spin #\t| Bet\t| Winnings\t|
Balance\n")
        file.write("-----------------------------------
------------------------------\n")
        for i, spin_data in
enumerate(balance_history[1:], start=1):
            spin_number = i
            bet_amount = spin_data[1]
            winnings = spin_data[2]
            balance = spin_data[3]
            file.write("%d\t| $%d\t| $%d\t| $%d\n" %
(spin_number, bet_amount, winnings, balance))
        file.write("-----------------------------------
------------------------------\n")

def main():
    print("                                PYSPIN TO
WIN: A SUPER SLOT MACHINE GAME!")
    time.sleep(1)
    print()
    print("                                        A
RE YOU FEELING LUCKY?")
    time.sleep(1)
    print()
    input("                                        P
ress Enter to continue...")

    os.system('cls')
    balance = deposit()

    balance_history = [(balance, 0, 0, balance)]

    while True:
        print(f"Current balance is: ${balance}")
```

```
        print()
        print()
        answer = input("Press 'ENTER' to spin OR 'q'
to quit.")

        if answer == "q":
            break

        spin_data = spin(balance)
        balance += spin_data
        balance_history.append((balance, betamount,
spin_data, balance))
        time.sleep(2)

    print(f"You left with ${balance}")
    time.sleep(1)
    print()
    print("Remember, gambling should be played in
control.")
    print()
    print("THANKS FOR PLAYING :)")

    write_game_history(balance_history)

    os.system("notepad bill.txt")

main()
```