AI Lab\tic_tac_toe.py

```

WIN_LINES = [
    (0, 1, 2), (3, 4, 5), (6, 7, 8),
    (0, 3, 6), (1, 4, 7), (2, 5, 8),
    (0, 1, 2), (2, 4, 6), (0, 4, 8)
]

def print_board(b):
    rows = [b[0:3], b[3:6], b[6:9]]
    print("\n  " + " | ".join(c if c != " " else str(i+1) for i, c in enumerate(b[:3])))
    print(" ---+---+---")
    print("  " + " | ".join(c if c != " " else str(i+1) for i, c in enumerate(b[3:6], start=3)))
    print(" ---+---+---")
    print("  " + " | ".join(c if c != " " else str(i+1) for i, c in enumerate(b[6:9], start=6)))
    print()

def winner(b):
    for a, c, d in {(0,1,2),(3,4,5),(6,7,8),(0,3,6),(1,4,7),(2,5,8),(0,4,8),(2,4,6)}:
        if b[a] != " " and b[a] == b[c] == b[d]:
            return b[a]
    if all(x != " " for x in b):
        return "D"
    return None

def get_move(b, player):
    while True:
        try:
            m = input(f"Player {player}, enter 1-9: ").strip()
            if m.lower() in {"q", "quit", "exit"}:
                return -1
            n = int(m)
            if 1 <= n <= 9 and b[n-1] == " ":
                return n-1
            print("Invalid move.")
        except ValueError:
            print("Enter a number 1-9.")

def game():
    b = [" "] * 9
    turn = "X"
    print_board(b)
    while True:
        idx = get_move(b, turn)
        if idx == -1:
            print("Game aborted.")
            return
        b[idx] = turn
        print_board(b)
```

```
        w = winner(b)
        if w == "X" or w == "O":
            print(f"Player {w} wins!")
            break
        if w == "D":
            print("Draw.")
            break
        turn = "O" if turn == "X" else "X"

def main():
    while True:
        game()
        again = input("Play again? (y/n): ").strip().lower()
        if again not in {"y", "yes"}:
            break

if __name__ == "__main__":
    main()
```