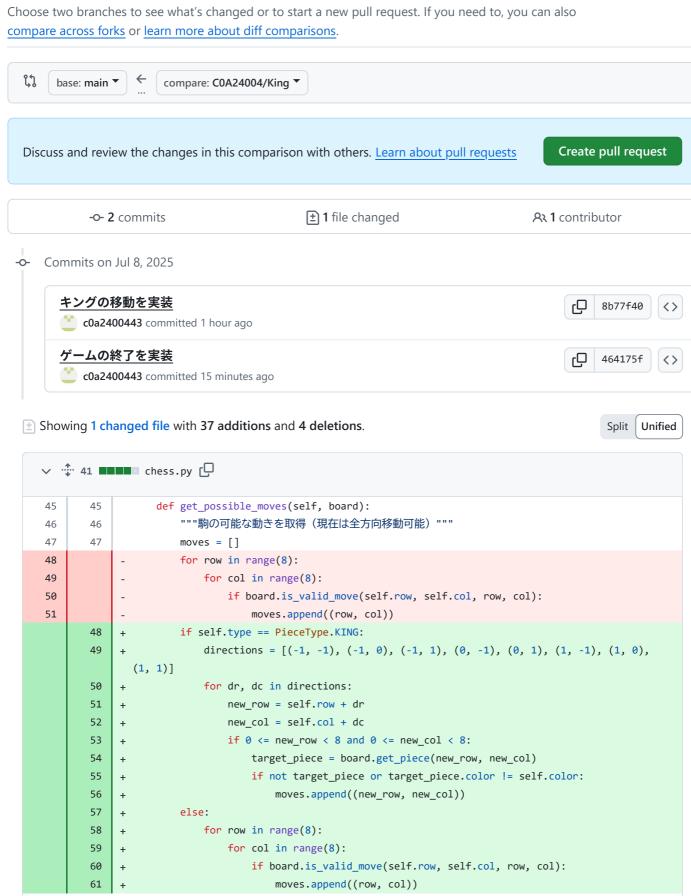


## Comparing changes



```
52
                       return moves
        62
53
        63
 54
        64
                   def move(self, new_row, new_col):
        88
                       self.current_turn = PieceColor.WHITE
78
 79
        89
                       self.selected_piece = None
        90
                       self.selected_pos = None
80
        91
                       self.winner = None
81
        92
                       self.possible_moves = []
                       self.setup_initial_position()
82
        93
83
        94
                       # 移動先が有効かチェック
137
       148
138
       149
                       if not self.is_valid_move(from_row, from_col, to_row, to_col):
139
       150
                           return False
       151
                       #target_pieceを定義
       152
                       target_piece = self.get_piece(to_row, to_col)
       153
       154
                       # キングが取られたら勝敗を設定
       155
                       if target_piece and target_piece.type == PieceType.KING:
       156
       157
                           self.set piece(to row, to col, piece)
       158
                           self.set_piece(from_row, from_col, None)
                           piece.move(to_row, to_col)
       159
                           self.winner = piece.color # 勝った側の色
       160
       161
                           return True
       162
140
       163
                       # 移動実行
141
       164
142
       165
                       self.set_piece(to_row, to_col, piece)
233
       256
                   def draw info(self):
                       """ゲーム情報を描画"""
234
       257
235
       258
                       info_y = BOARD_SIZE * SQUARE_SIZE + 10
       259
                       if self.board.winner != None:
       260
                           winner color = "White" if self.board.winner == PieceColor.WHITE else "Black"
       261
       262
                           result_text = f"{winner_color} wins!"
       263
                           text = self.font.render(result_text, True, RED)
                           self.screen.blit(text, (10, info_y))
       264
                           return # 勝敗が決まったら他の表示は不要
       265
236
       266
237
       267
                       # 現在のターン(英語で表示)
                       turn text = f"Current Turn: {'White' if self.board.current turn ==
238
       268
               PieceColor.WHITE else 'Black'}"
249
       279
                   def handle_click(self, mouse_pos):
250
       280
                       """マウスクリックを処理"""
251
       281
       282
                       if self.board.winner:
                           return # 勝敗が決まったらクリック操作無効
       283
             +
       284
252
                       row, col = self.get_board_pos(mouse_pos)
       285
253
       286
                       if row is not None and col is not None:
                           if self.board.selected piece:
254
       287
```