

github : https://github.com/donghn5/ai_crypto < 이동현, 이건 >

#CODE

```
import pandas as pd
```

```
def cal_bookdelta_bookimbalance(bid_level, ask_level):
    book_delta = bid_level['quantity'].sum() - ask_level['quantity'].sum()
    total_volume = bid_level['quantity'].sum() + ask_level['quantity'].sum()
    book_imbalance = book_delta / total_volume if total_volume != 0 else 0
    return book_delta, book_imbalance
```

```
def cal_mid_price(bid_level, ask_level):
    if len(bid_level) > 0 and len(ask_level) > 0:
        bid_top_price = bid_level.iloc[0]['price']
        ask_top_price = ask_level.iloc[0]['price']
        return (bid_top_price + ask_top_price) * 0.5
    else:
        return None
```

```
def cal_spread(bid_level, ask_level):
    if len(bid_level) > 0 and len(ask_level) > 0:
        bid_top_price = bid_level.iloc[0]['price']
        ask_top_price = ask_level.iloc[0]['price']
        return ask_top_price - bid_top_price
    else:
        return None
```

```
def cal_market_depth(bid_level, ask_level, depth_level=5):
    bid_depth = bid_level['quantity'].head(depth_level).sum()
    ask_depth = ask_level['quantity'].head(depth_level).sum()
    return bid_depth, ask_depth
```

```
file_names = [
    'book-2024-04-27-ETH-bithumb.csv',
    'book-2024-04-28-ETH-bithumb.csv'
]
```

```
all_results = []
```

```
for file_name in file_names:
    df = pd.read_csv(file_name)
    df['timestamp'] = pd.to_datetime(df['timestamp'])
```

```

groups = df.groupby('timestamp')

results = []

for timestamp, group in groups:
    gr_bid_level = group[group['type'] == 0]
    gr_ask_level = group[group['type'] == 1]

    mid_price = cal_mid_price(gr_bid_level, gr_ask_level)
    book_delta, book_imbalance = cal_bookdelta_bookimbalance(gr_bid_level,
gr_ask_level)
    spread = cal_spread(gr_bid_level, gr_ask_level)
    bid_depth, ask_depth = cal_market_depth(gr_bid_level, gr_ask_level)

    results.append({
        'timestamp': timestamp,
        'mid_price': mid_price,
        'book_delta': book_delta,
        'book_imbalance': book_imbalance,
        'spread': spread,
        'bid_depth': bid_depth,
        'ask_depth': ask_depth,
    })

all_results.extend(results)

all_results_df = pd.DataFrame(all_results)
all_results_df.to_csv('2024-05-25-ETHtest1.csv', index=False)

```