題目一

請寫出一條查詢語句 (SQL)，列出在 2023 年 5 月下訂的訂單，使用台幣付款且5月總金額最 多的前 10 筆的旅宿 ID (bnb_id), 旅宿名稱 (bnb_name), 5 月總金額 (may_amount)

```sql
SELECT
    bnbs.id AS bnb_id,
    bnbs.name AS bnb_name,
    SUM(amount) AS may_amount
FROM
    orders
JOIN
    rooms
ON
    orders.room_id = rooms.id
JOIN
    bnbs
ON
    orders.bnb_id = bnbs.id
WHERE
    currency = 'TWD'
    AND DATE_FORMAT(created_at, '%Y-%m') = '2023-05'
GROUP BY
    bnbs.id
ORDER BY
    may_amount DESC
LIMIT 10;
```

題目二

在題目一的執行下，我們發現 SQL 執行速度很慢(undefined)，您會怎麼去優化?請列出您怎麼判斷與優化的方式

定義:

show variables like '%long_query_time%' // 預設OFF

set global slow_query_log = 1; // 開啟

set global long_query_time = 2 // 默認10 秒, 假設我們定義"很慢"是2秒, 這裡就設定為2秒

判斷0: 先用Prometheus + Grafana(或其他工具) 檢查連接

判斷1: slow query log

去/var/lib/mysql/hostname-slow.log檢查

判斷2: explain 檢查type(例如 變成all的話indexing沒作用), key, filtered column

---

優化0: 直接把數據mount到RAM裡面

sudo mount -t ramfs -o size=200000m ramfs /mnt/memory/asiayo

優化1：WHERE 中使用 SARGABLE query
DATE_FORMAT函數讓DB沒辦法使用created_at上的index

```
15 WHERE
16     currency = 'TWD'
17     AND created_at >= '2023-05-01 00:00:00'
18     AND created_at < '2023-06-01 00:00:00'
```

優化2：denormalize
把bobs.name 合併到orders, 省去join

優化3: cache result
如果操作常常使用可以暫存起來
Indexing 可能沒用？因為有SUM函數，還是要把數據sequential read一遍

API 實作測驗 題目一

請實作一個提供匯率轉換的 API，轉換金額請四捨五入到小數點第二位，且轉換後的金額顯 示格式請增加逗點分隔做為千分位表示，如 `123,456.78`，不限程式語言。(請附上 github / gitlab / bitbucket 連結)

GitHub: https://github.com/huaiche94/currency-converter.git
README.md ✅
Unit tests ✅

實作要求:
- 請附上如何執行的說明文件如 README
- 請針對您的 API 撰寫單元測試
- Method 請用 GET
- 輸入範例:

```
?source=USD&target=JPY&amount=$1,525
```

- 輸出範例:

```
{
  "msg": "success"
  "amount": "$170496.53"
}
```

- 請使用我們提供的匯率資料

```
{
  "currencies": {
    "TWD": {
      "TWD": 1,
      "JPY": 3.669,
      "USD": 0.03281
    },
    "JPY": {
      "TWD": 0.26956,
      "JPY": 1,
      "USD": 0.00885
    },
    "USD": {
      "TWD": 30.444,
      "JPY": 111.801,
      "USD": 1
    }
  }
}
```