Exporter:

我被要求exporter要被2個普米scrape，2個普米拿到的值要一樣

我要collect()裡只做metric計算，檔案複製 rename 刪除移到 以繁體中文詳細說明:

我想要理解這段程式碼用法，為了加到我到exporter中做實驗時，為什麼start\_http\_server(8000)需換成server = ThreadingSimpleServer(("0.0.0.0", 8000), CustomMetricsHandler)，我對python的物件導向不熟，你能教我嗎，或是你能給更簡單的範例:

我想要理解這段程式碼用法，為了加到我到exporter中做實驗時，為什麼這樣修改就可以，我對python的動態新增label、計算metric不熟，你能教我嗎，或是你能給更簡單的範例:

我想要理解這段程式碼用法，為了加到我到exporter中做實驗時，為什麼這樣修改可以動態新增 labels、如何計算 metric?、如何動態更新 Prometheus 指標的 labels？，我對這些code不理解，你能教我嗎，你能多print出metric的計算過嗎，或是你能給更簡單的範例:

我想要理解這段程式碼用法，為了加到我到exporter中做實驗時，為什麼這樣修改可以動態新增 labels、如何計算 metric?、如何動態更新 Prometheus 指標的 labels？，我對這些code不理解，你能教我嗎，你能多print出metric的計算過嗎，或是你能給更簡單的範例:

以繁體中文詳細說明分析:這2段程式碼差異，

1:

2:

以繁體中文詳細說明分析:第一段程式碼要怎麼改為使用怎麼第2段的動態label機制

以繁體中文詳細說明:把解析來源換成讀data.collect.csv，並給我data.collect.csv內容的範例，像是with open(tmp\_log\_file, 'r') as f:

                reader = csv.reader(f)

新需求，要能依log，動態新增label的能力

Log.csv:

host\_1,job\_A, {service\_name=”aaa”, container\_name=”bbbb”}

host\_1,job\_A

host\_1,job\_B

host\_2,job\_A

host\_2,job\_C

host\_3,job\_B, {module\_name=”cbbb”}

host\_3,job\_B

host\_3,job\_B

metrics:

# HELP log\_host\_job\_count Count of occurrences of host and job\_name in log

# TYPE log\_host\_job\_count gauge

log\_host\_job\_count{host="host\_1", job\_name="job\_A"} 1

log\_host\_job\_count{host="host\_1", job\_name="job\_A", service\_name=”aaa”, container\_name=”bbbb”} 1

log\_host\_job\_count{host="host\_1", job\_name="job\_B"} 1

log\_host\_job\_count{host="host\_2", job\_name="job\_A"} 1

log\_host\_job\_count{host="host\_2", job\_name="job\_C"} 1

log\_host\_job\_count{host="host\_3", job\_name="job\_B"} 2

log\_host\_job\_count{host="host\_3", job\_name="job\_B", module\_name="cbbb"} 1

metric可不可以有多個label set

以CodeBlock格式給我這3張圖片的程式碼內容

我想要理解這段程式碼用法，為了加到我的exporter中做實驗時，為什麼這樣可以不顯示空值的labels，我對這些code不理解，你能教我嗎，或是你能給更簡單的範例:

“””

from prometheus\_client import start\_http\_server, Gauge, REGISTRY

from prometheus\_client.core import GaugeMetricFamily

import time

class CustomGauge:

"""

Custom Gauge to allow dynamic label keys (omit empty labels).

"""

def \_\_init\_\_(self, name, documentation):

self.name = name

self.documentation = documentation

self.metrics = {}

def set(self, labels, value):

# Filter out labels with empty string values

filtered\_labels = {k: v for k, v in labels.items() if v}

# Use a tuple of sorted (k,v) pairs as key for uniqueness

key = tuple(sorted(filtered\_labels.items()))

self.metrics[key] = (filtered\_labels, value)

def collect(self):

# Group metrics by label keys for correct GaugeMetricFamily grouping

group = {}

for label\_tuple, (labels, value) in self.metrics.items():

label\_keys = tuple(labels.keys())

if label\_keys not in group:

group[label\_keys] = []

group[label\_keys].append((labels, value))

for label\_keys, series in group.items():

gauge = GaugeMetricFamily(self.name, self.documentation, labels=label\_keys)

for labels, value in series:

gauge.add\_metric([labels[k] for k in label\_keys], value)

yield gauge

# Create a custom gauge instance

custom\_gauge = CustomGauge('my\_test\_metric', 'test')

# Register the collector

REGISTRY.register(custom\_gauge)

# Start HTTP server

start\_http\_server(8000)

while True:

# Only non-empty labels will appear

custom\_gauge.set({'a': 'foo', 'b': '', 'c': 'bar'}, 1.0)

custom\_gauge.set({'a': 'abc', 'b': 'def', 'c': ''}, 2.0)

time.sleep(10) “””

我想要理解的程式碼用法

“””

from prometheus\_client import start\_http\_server, Gauge, REGISTRY

from prometheus\_client.core import GaugeMetricFamily

import time

class CustomGauge:

"""

Custom Gauge to allow dynamic label keys (omit empty labels).

"""

def \_\_init\_\_(self, name, documentation):

self.name = name

self.documentation = documentation

self.metrics = {}

def set(self, labels, value):

# Filter out labels with empty string values

filtered\_labels = {k: v for k, v in labels.items() if v}

# Use a tuple of sorted (k,v) pairs as key for uniqueness

key = tuple(sorted(filtered\_labels.items()))

self.metrics[key] = (filtered\_labels, value)

def collect(self):

# Group metrics by label keys for correct GaugeMetricFamily grouping

group = {}

for label\_tuple, (labels, value) in self.metrics.items():

label\_keys = tuple(labels.keys())

if label\_keys not in group:

group[label\_keys] = []

group[label\_keys].append((labels, value))

for label\_keys, series in group.items():

gauge = GaugeMetricFamily(self.name, self.documentation, labels=label\_keys)

for labels, value in series:

gauge.add\_metric([labels[k] for k in label\_keys], value)

yield gauge

# Create a custom gauge instance

custom\_gauge = CustomGauge('my\_test\_metric', 'test')

# Register the collector

REGISTRY.register(custom\_gauge)

# Start HTTP server

start\_http\_server(8000)

while True:

# Only non-empty labels will appear

custom\_gauge.set({'a': 'foo', 'b': '', 'c': 'bar'}, 1.0)

custom\_gauge.set({'a': 'abc', 'b': 'def', 'c': ''}, 2.0)

time.sleep(10)

“””，為了整合進到我的exporter中做實驗時，為什麼這樣可以不顯示空值的labels，我對這些code不理解，你能教我嗎，或是你能給更簡單的範例:

# 任務

依照”任務說明”教會我並給我範例程式碼

# 任務說明

1.將我給你的程式碼好好利用

2.說明並讓我理解整合的方法

3.給我範例程式碼

4.‘程式碼’一要怎麼整合進’程式碼二’呢

# 數據庫

## 程式碼1

“””

from prometheus\_client import start\_http\_server, Gauge, REGISTRY

from prometheus\_client.core import GaugeMetricFamily

import time

class CustomGauge:

"""

Custom Gauge to allow dynamic label keys (omit empty labels).

"""

def \_\_init\_\_(self, name, documentation):

self.name = name

self.documentation = documentation

self.metrics = {}

def set(self, labels, value):

# Filter out labels with empty string values

filtered\_labels = {k: v for k, v in labels.items() if v}

# Use a tuple of sorted (k,v) pairs as key for uniqueness

key = tuple(sorted(filtered\_labels.items()))

self.metrics[key] = (filtered\_labels, value)

def collect(self):

# Group metrics by label keys for correct GaugeMetricFamily grouping

group = {}

for label\_tuple, (labels, value) in self.metrics.items():

label\_keys = tuple(labels.keys())

if label\_keys not in group:

group[label\_keys] = []

group[label\_keys].append((labels, value))

for label\_keys, series in group.items():

gauge = GaugeMetricFamily(self.name, self.documentation, labels=label\_keys)

for labels, value in series:

gauge.add\_metric([labels[k] for k in label\_keys], value)

yield gauge

# Create a custom gauge instance

custom\_gauge = CustomGauge('my\_test\_metric', 'test')

# Register the collector

REGISTRY.register(custom\_gauge)

# Start HTTP server

start\_http\_server(8000)

while True:

# Only non-empty labels will appear

custom\_gauge.set({'a': 'foo', 'b': '', 'c': 'bar'}, 1.0)

custom\_gauge.set({'a': 'abc', 'b': 'def', 'c': ''}, 2.0)

time.sleep(10)

“””

## 程式碼2

“””

import csv

import os

import time

import logging

from prometheus\_client import Gauge, start\_http\_server

# === [1] 設定 log 紀錄格式，方便我們觀察 metrics 的更新狀況 ===

logging.basicConfig(

    level=logging.INFO,

    format="%(asctime)s - %(levelname)s - %(message)s",

)

# === [2] 設定要讀取的 CSV 檔案 ===

CSV\_FILE = "bak-data\_collect.csv"

# === [3] 初始化 Prometheus 的 Gauge（我們稍後會根據 labels 來建立） ===

log\_host\_job\_count = None  # 先留空，等 labels 決定好再建立

def parse\_csv():

    """解析 CSV，動態收集所有出現過的 labels，並統計每組 labels 的出現次數"""

    counts = {}  # key = frozenset(labels)，value = 次數

    dynamic\_labels = {"host", "job\_name"}  # 初始固定兩個 labels

    if not os.path.exists(CSV\_FILE):

        logging.error(f"CSV 檔案 `{CSV\_FILE}` 不存在！")

        return counts, sorted(list(dynamic\_labels))

    with open(CSV\_FILE, 'r', encoding='utf-8') as f:

        reader = csv.reader(f)

        for row in reader:

            if len(row) < 2:

                continue  # 至少要有 host 和 job\_name

            host = row[0].strip()

            job\_name = row[1].strip()

            extra\_labels = {}  # 額外的 labels

            # === 處理第三欄之後的欄位：包含額外 labels ===

            for col in row[2:]:

                col = col.strip()

                if col.startswith("{") and col.endswith("}"):

                    col = col[1:-1].strip()  # 拿掉大括號

                # 拆解 key=value

                key\_value\_pairs = col.split(",")

                for pair in key\_value\_pairs:

                    pair = pair.strip()

                    if "=" in pair:

                        key, value = map(str.strip, pair.split("=", 1))

                        key = key.replace("{", "").replace("}", "").strip()

                        value = value.replace("”", "").replace("“", "").strip("\"")

                        if key and value:

                            extra\_labels[key] = value  # 收集有效 label

            # 更新動態 labels（只加入 key，不重複）

            dynamic\_labels.update(extra\_labels.keys())

            # 組合完整的 labels

            full\_label\_dict = {\*\*extra\_labels, "host": host, "job\_name": job\_name}

            key = frozenset(full\_label\_dict.items())

            counts[key] = counts.get(key, 0) + 1

    logging.info(f"[parse\_csv] 最終 labels: {sorted(dynamic\_labels)}")

    return counts, sorted(dynamic\_labels)

def update\_metrics():

    """每次更新 metrics（Prometheus 會來抓），我們就重建一次"""

    global log\_host\_job\_count

    counts, labels\_list = parse\_csv()

    # 初始化或重新初始化 Gauge

    log\_host\_job\_count = Gauge("log\_host\_job\_count", "Count of host/job\_name with optional labels", labels\_list)

    log\_host\_job\_count.\_metrics.clear()  # 清掉舊資料

    logging.info("[update\_metrics] 開始設定 metrics...")

    for key, count in counts.items():

        labels\_dict = dict(key)

        complete\_labels = {k: labels\_dict.get(k, "") for k in labels\_list}  # 確保所有 label key 都存在

        logging.info(f"[metric] {complete\_labels} => {count}")

        log\_host\_job\_count.labels(\*\*complete\_labels).set(count)

if \_\_name\_\_ == "\_\_main\_\_":

    start\_http\_server(8080)

    logging.info("Exporter 正在 http://localhost:8080/metrics 執行...")

    while True:

        update\_metrics()

        time.sleep(10)  # 每 10 秒更新一次

“””

## bak-data\_collect.csv

“””

host\_1,job\_A, {service\_name=”aaa”, container\_name=”bbbb”}

host\_1,job\_A

host\_1,job\_B

host\_2,job\_A

host\_2,job\_C

host\_3,job\_B, {module\_name=”cbbb”}

host\_3,job\_B

host\_3,job\_B

“””

fastapi:

我在寫fastapi，寫好的api要給react call，把值存到mariadb，為什麼我看mariadb存的值是{message: “hello”, FAIL}，在swagger的/doc看到的reponse是

”/”message/”:/”hello/””

我要存的格式是

{

"username": "佳恩",

" age ": 28,

" hobbies ": ["coding", "gaming"],

" content":

{

message: “hello”,

student: “allen”

}

}

，username、age、hobbies、content 都是mariadb schema的欄位

存整包 JSON資料存進mariadb的content欄位，資料格式LONGTEXT

我要用sessonmaker不要async

我想要理解json資料存進mariadb的用法(資料格式LONGTEXT)，為了做之後跟react串接的實驗，我對fastapi的寫法與mariadb的用法不熟，你能教我嗎，或是你能給更簡單的範例:

我要DeclaratineBase改寫(使用python typing 與SQLAlchemy 2.0寫法)

只確定content是一筆json資料，不確定會有哪一些key

我想要理解json資料存進mariadb的用法(資料格式LONGTEXT)，為了做之後跟react串接的實驗，我對fastapi的寫法與json存入mariadb的用法不熟，這樣寫會有甚麼問題嗎?，你能教我嗎，或是你能給更簡單的範例:

class UserData(Base):

\_\_tablename\_\_ = "user\_data"

id: Mapped[int] = mapped\_column(primary\_key=True, autoincrement=True)

username: Mapped[str] = mapped\_column(String(100))

age: Mapped[int] = mapped\_column(Integer)

hobbies: Mapped[str] = mapped\_column(Text) # 儲存 JSON string

content: Mapped[str] = mapped\_column(LONGTEXT) # 儲存 JSON string

class UserInput(BaseModel):

username: str

age: int

hobbies: List[str]

content: Any

store\_data能加上response\_model嗎

使用 DBeaver 或 Navicat 檢查資料庫欄位內容

Vm:

參考"Cluster version.pdf"，以繁體中文詳細說明用docker-compose建立cluster VictoriaMetrics

參考"Cluster version.pdf"，以繁體中文詳細說明如何debug此cluster VictoriaMetrics

參考"Cluster version.pdf"，以繁體中文詳細說明若此cluster VictoriaMetric失效

參考"Cluster version.pdf"，以繁體中文詳細說明若vmselect失效

參考"Cluster version.pdf"，以繁體中文詳細說明若2個vmselect 中的一個失效

參考"Cluster version.pdf"，以繁體中文詳細說明使用k8s建置

參考"Cluster version.pdf"，以繁體中文詳細說明使用k8s建置vmstorage、vminsert、vmselect

參考"Cluster version.pdf"，以繁體中文詳細說明不使用k8s能實現動態水平擴展、動態垂直擴展嗎

參考"Cluster version.pdf"，以繁體中文詳細說明:我用docker-compose 建立了vmstorage, vminsert, vmselect, vmagent, vmalert 各一個node，請問怎麼知道使用了多少cpu、mem

以繁體中文詳細說明:我用docker-compose 建立了vmstorage, vminsert, vmselect, vmagent, vmalert 各一個node，請問怎麼知道使用了多少cpu、mem

公司使用gitlab pipeline，用statefulset把vminsert佈到k8s，要求我改為用Deployment，但我對這statefulset、Deployment都不熟，你能教我嗎

以繁體中文詳細說明:我可以把statefulset裡的resorceversion、uid、podmanagmentpolicy、revisionhistorylimit、servicename、creationtimestamp、apiversion、terminationmessagepath、terminationmessagepolicy、volumemounts、restartpolicy、schedulername、sercuritycontentext、terminationgraceperiodseconds、updatestrategy、voluneclaimtemplates、status拿掉嗎

以繁體中文詳細說明:我可以把statefulset裡的metadata、dnsconfig、dnspolicy、tolarations拿掉嗎

以繁體中文詳細說明:我改成deployment後，argocd的ui的deployment資訊顯示” OutOfSync (This resource is not present in the application's source. It will be deleted from Kubernetes if the prune option is enabled during sync.)”

以繁體中文詳細說明:argocd的ui的StatefulSet資訊顯示” Resource is not found in cluster: app/v1/StatefulSet:dev-victoriametricsmonitoring-v3”

詳細說明: k8s上部屬vminsert，應該使用deployment還是statefulse

怎麼改成k8s上，透過statefulset部屬vmagent

公司的pipeline都是使用statefuleset來部屬，請問我要怎麼改成用deployment

vminsert 需要 StatefulSet 嗎

詳細說明:我被要求vminsert可用的cpu、mem分配要合理，但我查不到建議值

我改成deployment後，argocd的ui的deployment資訊顯示” OutOfSync (This resource is not present in the application's source. It will be deleted from Kubernetes if the prune option is enabled during sync.)”

Vmalert ha設定

以繁體中文詳細說明:我用dockercompose在虛擬機建好vminsert、vmselect、vmstorage、vmagent、vmalert了，請問怎麼用vmalert + alertmanager設定alert

我想要理解vmalert + alertmanager用法，為了做vminsert、vmselect、vmstorage、vmagent、vmaler cpu使用率超過10%發出mail通知我的實驗，我對vmalert與alertmanager用法不熟，你能教我嗎，或是你能給更簡單的範例:

以繁體中文詳細說明:

Vmalert之後要用gitlab pipeline佈到k8s環境，應該使用deployment還是statefulset?

參考vmalert-ha.pdf，以繁體中文詳細說明:我需要建立vmalert ha到k8s，我對vmalert ha用法不熟，你能教我嗎，或是你能給更簡單的範例

我用dockercompose在虛擬機建好vminsert、vmselect、vmstorage、vmagent、vmalert了，我想要理解vmalert + alertmanager用法，為了做vminsert、vmselect、vmstorage、vmagent、vmaler cpu使用率超過10%發出mail通知我的實驗，我對vmalert與alertmanager用法不熟，你能教我嗎，或是你能給更簡單的範例:

以繁體中文詳細說明:

如果Vmalert 連線不到vmagent，但vmselect裡有vmagent，這樣是不是還是能監控，當vmagent的cpu使用率超過10%時，發出vmalert呢

以繁體中文詳細說明:

我想要理解vmalert + alertmanager用法，為了做vminsert、vmselect、vmstorage、vmagent、vmaler cpu使用率超過10%發出mail通知我的實驗，我對alertmanager group如何去重多個vmalert發出警報的用法不熟，你能教我alertmanager的config設定嗎，例如怎麼寄信，或是你能給更簡單的範例:

以繁體中文詳細說明:

我想要理解vmalert + alertmanager用法，為了做vminsert、vmselect、vmstorage、vmagent、vmaler cpu使用率超過10%發出mail通知我的實驗，我對alertmanager cluster如何去重多個vmalert發出警報的用法不熟，你能教我alertmanager cluster的config設定嗎，例如怎麼寄信，或是你能給更簡單的範例:

以繁體中文詳細說明:

我想要理解vmalert + alertmanager用法，為了做vminsert、vmselect、vmstorage、vmagent、vmaler cpu使用率超過10%發出mail通知我的實驗，我對alertmanager cluster如何去重多個vmalert發出警報的用法不熟，你能教我alertmanager cluster的config設定嗎，例如怎麼寄信，或是你能給更簡單的範例:

以繁體中文詳細說明:

我想要理解vmalert ha架構去重+ alertmanager用法，為了做vminsert、vmselect、vmstorage、vmagent、vmaler cpu使用率超過10%發出mail通知我的實驗，我對vmalert ha架構如何去重多個vmalert發出警報的用法不熟，你能教我vmalert ha架構的config設定嗎，例如怎麼去重同時發出的alert，怎麼去重VictoriaMetrics裡儲存的alert紀錄，或是你能給更簡單的範例:

以繁體中文詳細說明:

我想要理解”3. VictoriaMetrics 儲存層去重

只要你的 vmalert rules 和 -dedup.minScrapeInterval 設對，VictoriaMetrics 自動幫你合併重複數據:只會留下同一組 labels、同一時間點的唯一一份結果

查詢歷史警報或錄製資料時不會看到重複數據

查詢歷史警報或錄製資料時不會看到重複數據”

這一段意義是每個vmalert都寫入一份alert記錄到VictoriaMetrics裡儲存，那VictoriaMetrics是透過vmalert還是vmselect設定的去重interval做去重呢，alert記錄只會保留一份，還是會因為vminsert repication=3，而有3份呢?或是你能給更簡單的範例:

Ui開發:

以繁體中文詳細說明:

我是台積電it，想要開發一個gui服務，方便從原先輸入json來使用api，變為使用gui來做到，必須要有ha高可用，之後要部署到k8s

主管要我跟他分析應使用哪一種軟體工具或框架來實作

你能教我嗎

你能給我簡單的範例嗎

你能給我目前對話內容的投影片大綱嗎，要有React JSON Schema Form、Ant Design + React Hook Form、Formik / React Final Form，三者的比較

以繁體中文詳細說明:

我想要理解React JSON Schema Form用法，為了做React JSON Schema Form開發Web GUI輸入json來使用api的實驗，我對React & React JSON Schema Form的用法不熟，你能教我嗎，或是你能給更簡單的範例:

以繁體中文詳細說明:

我想要理解Ant Design + React Hook Form用法，為了做Ant Design + React Hook Form開發Web GUI輸入json來使用api的實驗，我對React & Ant Design + React Hook Form的用法不熟，你能教我嗎，或是你能給更簡單的範例:

以繁體中文詳細說明:

我想要理解Formik / React Final Form用法，為了做Formik / React Final Form開發Web GUI輸入json來使用api的實驗，我對React & Formik / React Final Form的用法不熟，你能教我嗎，或是你能給更簡單的範例:

Config移出args:

以繁體中文詳細說明:我的公司使用Helm，比較這3種方式:

方法 適用場景 優點 缺點

方法 1：直接寫在 args vmstorage 節點固定，簡單環境 易於理解，適合小型集群 變更 vmstorage 需更新 Deployment

方法 2：使用 ConfigMap vmstorage 節點可能變更，需集中管理 易於維護，可即時修改 需要多管理 ConfigMap

方法 3：使用環境變數 env 配合 CI/CD，vmstorage 節點不固定 可動態管理變數，適合自動化 變數長時不易閱讀

以繁體中文詳細說明:我目前將vmagent的srrape job的 yml放在gitlab ci階段，被要求移到cd階段，怎麼處理

這個Comfig map可以給不同的vmagent共同使用嗎

Capacity說明 + Grafana 圖表

我想要理解rate()、irate()、increase()、sum()、avg()、min()、max()、count()、stddev()用法，為了用Grafana做VictoriaMetric- Cluster version的Capacity planning實驗，你能教我嗎，或是你能給更簡單的範例:  
  
  
我想要理解by (job)、by (instance)用法，為了用Grafana做VictoriaMetric- Cluster version的Capacity planning實驗，你能教我嗎，或是你能給更簡單的範例:

我想要理解折線圖、長條圖、stat圖用法，為了用Grafana做VictoriaMetric- Cluster version的Capacity planning實驗，你能教我嗎，或是你能給更簡單的範例:

以繁體中文詳細說明:Prometheus scrape intelval是1、garafana data source 設定是15秒，請問這樣我panel的rate()[時間要填多少]，15s、1m、還是4m

以繁體中文詳細說明:Min step vs Min time interval 差別

以繁體中文詳細說明: Grafana $\_\_interval、$\_\_rate\_interval、Min step的關係

以繁體中文詳細說明: panel的md、interval

以繁體中文詳細說明:要怎麼各panel 設定不同的時間範圍呢?

以繁體中文詳細說明:要怎麼各panel 的ministep 都設定成1m，而不用分開設定呢?

以繁體中文詳細說明:grafana的alert功能、跟prometheus的alert 有什麼不同?

以繁體中文詳細說明:grafana的alert功能、跟prometheus的alert 有什麼不同?

以繁體中文詳細說明:為什麼panel的某個時間點會出現黃色縱向虛線呢?

Vmalert ha設定:

以繁體中文詳細說明: 我

修改gitlab pipeline，要求有一個json，裡面放密碼，密碼的值是gitlab ci/cd variables，有3個分別是ap:111,bp:222,cp:333

要有一個job透過json將檔案產生出來放入新建的cre\_files裡，job怎麼寫

aa.cre裡面放111

bb.cre裡面放222

cc.cre裡面放333

以繁體中文詳細說明:我在修改gitlab pipeline，要求有一個json，裡面放密碼，密碼的值是gitlab ci/cd variables，有3個分別是ap:111,bp:222,cp:333

要有一個job透過json將檔案產生出來放入新建的cre\_files裡，job怎麼寫

aa.cre裡面放111

bb.cre裡面放222

cc.cre裡面放333

Json內容:

{

aa: “$ap”,

bb:”$bp”,

cc:”$cp”

}

我的意思是給我手動建passwords.json，要依內容自動產生，不要寫死

- echo -n "$ap" > cre\_files/aa.cre

- echo -n "$bp" > cre\_files/bb.cre

- echo -n "$cp" > cre\_files/cc.cre

我想要理解gitlab pipeline job script->、|用法，為了實驗修改gitlab pipeline

，你能教我嗎，或是你能給更簡單的範例:

Pipeline修改

gitlab.yml，我需要能設置白名單控制job能不能執行，例如白名單: white-1/white-2/apple，代表只有project\_namespace=white-1或white-2或apple才可以執行，"white"沒有完全匹配就不能執行，我的白名單很多，所以給我通用的設置白名單方式

WHITELIST: "white-1/white-2/apple"可以嗎

如果要改成用job的rules:來控制執行條件，而不是script可以嗎

我的check\_whitelist是由qqq這個job trigger的，CONTAINER\_NAME是qqq裡宣告的，我為什麼不能在check\_whitelist做export WHITELIST\_REGEX="($(echo $WHITELIST | tr '/' '|'))"，rules: - if: '$CONTAINER\_NAME =~ $WHITELIST\_REGEX'