

最新Linux Foundation PCA | 100%合格率のPCA日本語対策試験 | 試験の準備方法Prometheus Certified Associate Exam 実際試験



P.S.JPNTTestがGoogle Driveで共有している無料の2025 Linux Foundation PCAダンプ: <https://drive.google.com/open?id=1Th9aDFxIhiOpaPV2abSOx2My3oOzLaCz>

JPNTTestはLinux FoundationのPCA認定試験についてすべて資料を提供するの唯一サイトでございます。受験者はJPNTTestが提供した資料を利用してPCA認証試験は問題にならないだけでなく、高い点数も合格することができます。

Linux Foundation PCA 認定試験の出題範囲:

トピック	出題範囲
トピック 1	• PromQL: この試験セクションでは、モニタリングスペシャリストのスキルを測定し、Prometheusクエリ言語 (PromQL) の概念に焦点を当てます。データの選択、レートとデリバティブの計算、そして時間やディメンションをまたいだ集計の実行について学びます。また、バイナリ演算子、ヒストグラム、タイムスタンプメトリックの使用法についても学習し、モニタリングデータを効果的に分析することで、システムのパフォーマンスと傾向を正確に解釈できるようにします。
トピック 2	• Prometheusの基礎: このドメインでは、DevOpsエンジニアの知識を評価し、Prometheusのコアアーキテクチャとコンポーネントに重点を置きます。設定とスクレイピングの手法、Prometheusシステムの制限、データモデルとラベル、データ収集に使用される表示形式などのトピックが含まれます。このセクションでは、分散環境における監視およびアラートツールキットとしてのPrometheusの仕組みをしっかりと理解できるようにします。
トピック 3	• アラートとダッシュボード: このセクションでは、クラウド運用エンジニアの能力を評価し、監視の可視化とアラート管理に焦点を当てます。ダッシュボードの基本、アラートルールの設定、そして通知を処理するためのAlertmanagerの使用について学びます。受験者はまた、いつ、何を、なぜアラートをトリガーするかという基本原則を習得し、信頼性の高い監視ダッシュボードとプロアクティブなアラートシステムを構築してシステムの安定性を維持できるようにします。
トピック 4	• インストールメンテーションとエクスポーター: このドメインでは、ソフトウェアエンジニアの能力を評価し、Prometheusをアプリケーションに統合する手法について扱います。クライアントライブラリの使用、コードのインストールメンテーションプロセス、メトリクスの適切な構造化と命名などが含まれます。また、このセクションでは、Prometheusが様々なシステムからメトリクスを収集し、効率的で標準化された監視実装を実現するエクスポーターについても紹介します。

- 可観測性の概念: このセクションでは、サイト信頼性エンジニア (SRE) のスキルを評価し、現代のシステムで使用されている可観測性の基本原則を網羅します。メトリクス、ログ、スパンなどのトレースメカニズムの理解、そしてプッシュ型とプル型のデータ収集方法の違いに焦点を当てます。また、サービス検出プロセス、そしてパフォーマンスと信頼性を監視するためのSLO、SLA、SLIの定義と維持の基礎についても学習します。

>> PCA日本語対策 <<

PCA実際試験、PCA受験対策書

テストが来るのを静かに待っている場合は、目を覚まして、別の方法でPCA試験を受ける準備ができている必要があります。最近のPCAガイド急流の効果が資格試験を通じて受験者の秘密兵器になったことを示した後、PCAトレーニング資料を勉強して「テストデータ」を書くことがあなたの選択に最適です。PCAガイドトレントのユーザーは、PCA試験で予期しない結果を得ることができます。

Linux Foundation Prometheus Certified Associate Exam 認定 PCA 試験問題 (Q21-Q26):

質問 # 21

Given the metric `prometheus_tsdb_lowest_timestamp_seconds`, how do you know in which month the lowest timestamp of your Prometheus TSDB belongs?

- A. `format_date(prometheus_tsdb_lowest_timestamp_seconds,"%M")`
- B. `prometheus_tsdb_lowest_timestamp_seconds % month`
- C. `month(prometheus_tsdb_lowest_timestamp_seconds)`
- D. `(time() - prometheus_tsdb_lowest_timestamp_seconds) / 86400`

正解: D

解説:

The metric `prometheus_tsdb_lowest_timestamp_seconds` provides the oldest stored sample timestamp in Prometheus's local TSDB (in Unix epoch seconds). To determine the age or approximate date of this timestamp, you compare it with the current time (using `time()` in PromQL).

The expression:

`(time() - prometheus_tsdb_lowest_timestamp_seconds) / 86400`

converts the difference between the current time and the oldest timestamp from seconds into days (1 day = 86,400 seconds). This gives the number of days since the earliest sample was stored, allowing you to infer the time range and approximate month manually. The other options are invalid because PromQL does not support direct date formatting (`format_date`) or `month()` extraction functions.

Reference:

Extracted and verified from Prometheus documentation - TSDB Internal Metrics, Time Functions in PromQL, and Using `time()` for Relative Calculations.

質問 # 22

The following is a list of metrics exposed by an application:

`http_requests_total{code="500"} 10`

`http_requests_total{code="200"} 20`

`http_requests_total{code="400"} 30`

`http_requests_total{verb="POST"} 30`

`http_requests_total{verb="GET"} 30`

What is the issue with the metric family?

- A. Metric names are missing a prefix to indicate which application is exposing the query.
- B. The value represents two different things across the dimensions: code and verb.

正解: B

解説:

Prometheus requires that a single metric name represents one well-defined thing, and all time series in that metric share the same set of label keys so the value's meaning is consistent across dimensions. The official guidance states that metrics should not "mix different dimensions under the same name," and that a metric name should have a consistent label schema; otherwise, "the same metric name would represent different things," making queries ambiguous and aggregations error-prone. In the example, `http_requests_total{code="..."}` expresses per-status-code request counts, while `http_requests_total{verb="..."}` expresses per-HTTP-method request counts. Because some series have only code and others only verb, the value changes its meaning across label sets, violating the consistency principle for a metric family. The correct approach is to expose one metric with both labels present on every series, e.g., `http_requests_total{code="200", method="GET"}`, ensuring every time series has the same label keys and the value always means "count of requests," sliced by the same dimensions. A missing application prefix is optional and not the core issue here.

質問 # 23

Where does Prometheus store its time series data by default?

- A. In etcd.
- B. In-memory only.
- C. In an external database such as InfluxDB.
- **D. In an embedded TSDB on local disk.**

正解: D

解説:

By default, Prometheus stores its time series data in a local, embedded Time Series Database (TSDB) on disk. The data is organized in block files under the `data/` directory inside Prometheus's storage path.

Each block typically covers two hours of data, containing chunks, index, and metadata files. Older blocks are compacted and deleted based on retention settings.

質問 # 24

What are the four golden signals of monitoring as defined by Google's SRE principles?

- **A. Traffic, Errors, Latency, Saturation**
- B. Utilization, Load, Disk, Network
- C. Availability, Logging, Errors, Throughput
- D. Requests, CPU, Memory, Latency

正解: A

解説:

The Four Golden Signals-Traffic, Errors, Latency, and Saturation-are key service-level indicators defined by Google's Site Reliability Engineering (SRE) discipline.

Traffic: Demand placed on the system (e.g., requests per second).

Errors: Rate of failed requests.

Latency: Time taken to serve requests.

Saturation: How "full" the system resources are (CPU, memory, etc.).

Prometheus and its metrics-based model are ideal for capturing these signals.

質問 # 25

With the following metrics over the last 5 minutes:

`up{instance="localhost"} 1 1 1 1 1`

`up{instance="server1"} 1 0 0 0 0`

What does the following query return:

`min_over_time(up[5m])`

- **A. {instance="localhost"} 1 {instance="server1"} 0**
- B. {instance="server1"} 0

正解: A

The `min_over_time()` function in PromQL returns the minimum sample value observed within the specified time range for each time series.

In the given data:

For up {instance="localhost"}, all samples are 1. The minimum value over 5 minutes is therefore 1.

For `up {instance="server1"}`, the sequence is 1 0 0 0 0. The minimum observed value is 0.

Thus, the query `min_over_time(up[5m])` returns two series - one per instance:

```
{instance="localhost"} 1
```

```
{instance="server1"} 0
```

This query is commonly used to check uptime consistency. If the minimum value over the time window is 0, it indicates at least one scrape failure (target down).

Reference:

Verified from Prometheus documentation - PromQL Range Vector Functions, `min_over_time()` definition, and up Metric Semantics sections.

質問 #26

• • • • •

弊社のPCA練習資料は、さまざまな学位の受験者に適しています。この受験者は、この分野の知識のレベルに関係なく互換性があります。これらのPrometheus Certified Associate Examトレーニング資料は、当社にとって名誉あるものであり、お客様の目標を達成するための最大の特権として扱っています。一方、理論と実践を離婚することはできませんが、心配する必要はありません。刺激テストの質問があり、学習と実践の両方を同時に行うことができます。私たちの知る限り、PCA試験準備は何百万人もの受験者に夢を追いかけて、より効率的に学習するように動機付けました。PCAの練習資料は、あなたを失望させません。

PCA 実際試験: <https://www.jpntest.com/shiken/PCA-mondaishu>

- 試験の準備方法-実用的なPCA日本語対策試験-ハイパスレートのPCA実際試験 □ URL □
www.japancert.com □ をコピーして開き、【 PCA 】を検索して無料でダウンロードしてくださいPCA学習教材
- コンプリートLinux Foundation PCA日本語対策 インタラクティブテストエンジンを使用して - ハイパスレートPCA実際試験 □ ➡ www.goshiken.com □ に移動し、《 PCA 》を検索して、無料でダウンロード可能な試験資料を探しますPCA関連資格試験対応
- PCA試験の準備方法 | 完璧なPCA日本語対策試験 | 素敵なPrometheus Certified Associate Exam実際試験 □ ☀
www.jpctestking.com □☀□ に移動し、“PCA”を検索して、無料でダウンロード可能な試験資料を探します
PCA合格体験記
- 便利なPCA日本語対策 - 合格スムーズPCA実際試験 | 人気PCA受験対策書 Prometheus Certified Associate Exam □ ▶ www.goshiken.com ◁ から（ PCA ）を検索して、試験資料を無料でダウンロードしてください
PCA合格記
- PCA認証資格 □ PCA受験対策書 □ PCA認証資格 □ □ www.passtest.jp □ から▶ PCA ◀を検索して、試験資料を無料でダウンロードしてくださいPCA受験対策書
- PCA試験の準備方法 | 完璧なPCA日本語対策試験 | 素敵なPrometheus Certified Associate Exam実際試験 □
ウェブサイト☀ www.goshiken.com □☀□を開き、➡ PCA □を検索して無料でダウンロードしてください
PCA関連資格試験対応
- 試験の準備方法-実用的なPCA日本語対策試験-ハイパスレートのPCA実際試験 □ （ www.jpexam.com ）は、□ PCA □を無料でダウンロードするのに最適なサイトですPCA全真模擬試験
- PCA勉強方法 □ PCA認定デベロッパー □ PCA試験関連情報 □ 検索するだけで➡ www.goshiken.com □から【 PCA 】を無料でダウンロードPCA試験問題解説集
- ハイパスレートのPCA日本語対策一回合格-便利なPCA実際試験 □ □ www.japancert.com □を入力して➡ PCA □を検索し、無料でダウンロードしてくださいPCAテスト対策書
- PCA合格体験記 □ PCA認定デベロッパー □ PCA復習資料 □ 「 www.goshiken.com 」から簡単に▷ PCA ◁を無料でダウンロードできますPCA関連合格問題
- 便利なPCA日本語対策 - 合格スムーズPCA実際試験 | 人気PCA受験対策書 Prometheus Certified Associate Exam □ ➡ www.it-passports.com □ サイトにて▶ PCA ◀問題集を無料で使おうPCA認定デベロッパー
- myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt,
myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, www.stes.tyc.edu.tw, www.stes.tyc.edu.tw,
shortcourses.russellcollege.edu.au, www.camcadexperts.com, neon.edu.sa, www.stes.tyc.edu.tw, www.stes.tyc.edu.tw,
study.stcs.edu.np, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt,
myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, Disposable vapes

さらに、JPNTest PCAダンプの一部が現在無料で提供されています：<https://drive.google.com/open?id=1Th9aDFxIhiOpaPV2abSOx2My3oOzLaCz>