PCA   
Principal Component Analysis

**is an algorithm to transform the columns of a dataset into a new set of features called Principal Components**

The first principal component captures the most variation in the data, but the second principal component captures the maximum variance

**Steps of PCA**

* Standardization

First, we need to standardize our dataset to ensure that each variable has a mean of 0 and a standard deviation of 1.

* Covariance Matrix Computation

Covariance measures the strength of joint variability between two or more variables, indicating how much they change in relation to each other

* Compute Eigenvalues and Eigenvectors of Covariance Matrix to Identify Principal Components

## How Principal Component Analysis(PCA) works?

Hence, PCA employs a linear transformation that is based on preserving the most variance in the data using the least number of dimensions

<https://www.machinelearningplus.com/machine-learning/principal-components-analysis-pca-better-explained/#google_vignette>

<https://www.geeksforgeeks.org/principal-component-analysis-pca/>

**devops**

DevOps is typically cross-functional (people from different IT-related business units) and uses different software tools.

* Coding – code development and review, source code management tools, code merging
* Building – continuous integration tools (like Jenkins), build status
* Testing – continuous testing tools (like Query surge , Selenium, Cucumber, JMeter) that provide feedback on business risks
* Packaging – artifact repository, application pre-deployment staging
* Releasing – change management, release approvals, release automation
* Configuring – infrastructure configuration and management, infrastructure as code tools
* Monitoring – applications performance monitoring, end-user experience

# The goals of DevOps

* Increase**speed** of development and release processes
* Make builds more **reliable**
* **Shorter turnaround** for new features and bug fixes
* Greater**Scalability**of applications and infrastructure
* Increased**security** by automating compliance practices
* Improved **collaboration**throughout the development lifecycle

<https://www.querysurge.com/solutions/devops-for-data?utm_source=bing&utm_medium=ad&utm_campaign=DevOp-for-Data-article&msclkid=75275054377c1f05e44e369016855798>

# the tools of devops

* **DevOps Tools For Configuration Management**

1. Chef
2. Puppet
3. Ansible
4. SaltStack
5. AWS Systems Manager

* **DevOps Tools For CI/CD**

Continuous integration (CI)

1. CircleCI
2. GitLab
3. Jenkins
4. Semaphore
5. CloudBees
6. AWS CodePipeline
7. AWS CodeDeploy
8. GitHub

* **DevOps Tools For Log Management**

1. AWS CloudWatch Logs
2. Loggly
3. Logz.io
4. ChaosSearch
5. Splunk Log Observer
6. Sumo Logic

<https://www.cloudzero.com/blog/devops-tools/>

# Data Warehouse

# data warehouse, or enterprise data warehouse (EDW), is a system that aggregates data from different sources into a single, central, consistent data store to support data analysis, data mining, artificial intelligence (AI) and machine learning.

data warehouse system enables an organization to run powerful analytics on large amounts of data (petabytes and petabytes) in ways that a standard database cannot.

Data warehousing systems have been a part of business intelligence (BI)

**ETL**

When database analysts want to move data from a data source into their data warehouse, this is the process they use

**Metadata**

Metadata is data about data. Basically, it describes all of the data that’s stored in a system to make it searchable. Some examples of metadata include authors, dates or locations of an article, create date of a file, the size of a file, etc.

**SQL query processing**

SQL is the de facto standard language for querying your data. This is the language that analysts use to pull out insights from their data stored in the data warehouse. Typically data warehouses have proprietary SQL query processing technologies tightly coupled with the compute. This allows for very high performance when it comes to your analytics.

**Data layer**

The data layer is the access layer that allows users to actually get to the data.

# Data warehouse vs. database

A database is built primarily for fast queries and transaction processing, not analytics. A database typically serves as the focused data store for a specific application, whereas a data warehouse stores data from any number (or even all) of the applications in your organization.

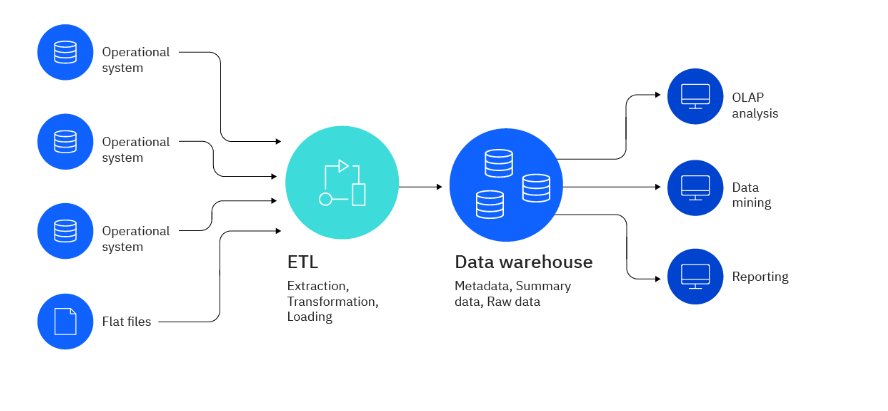
A database focuses on updating real-time data while a data warehouse has a broader scope, capturing current and historical data for predictive analytics, machine learning, and other advanced types of analysis.

Data base is fast in

* search
* add
* delete
* update

Data warehouse is fast in

* select



<https://www.ibm.com/topics/data-warehouse>

# task4 ways show the missing value