Here's the list of AWS Icons organized in markdown format:

```markdown
# AWS Icons List

## Analytics
- `aws-athena`
- `aws-cloudsearch`
- `aws-emr`
- `aws-finspace`
- `aws-kinesis`
- `aws-kinesis-data-analytics`
- `aws-kinesis-data-streams`
- `aws-kinesis-firehose`
- `aws-kinesis-video-streams`
- `aws-managed-streaming-for-apache-kafka`
- `aws-opensearch-service`
- `aws-quicksight`
- `aws-redshift`
- `aws-data-exchange`
- `aws-data-pipeline`
- `aws-glue`
- `aws-glue-databrew`
- `aws-glue-elastic-views`
- `aws-lake-formation`

## Application Integration
- `aws-api-gateway`
- `aws-appflow`
- `aws-eventbridge`
- `aws-managed-workflows-for-apache-airflow`
- `aws-mq`
- `aws-simple-notification-service`
- `aws-simple-queue-service`
- `aws-appsync`

## Business Applications
- `aws-alexa-for-business`
- `aws-chime`
- `aws-chime-sdk`
- `aws-chime-voice-connector`
- `aws-connect`
- `aws-honeycode`
- `aws-pinpoint`
- `aws-pinpoint-apis`
- `aws-simple-email-service`
- `aws-workdocs`
- `aws-workdocs-sdk`
- `aws-workmail`

## Compute
- `aws-ec2`
- `aws-ec2-auto-scaling`
```

- `aws-ec2-image-builder`
- `aws-ec2-m5n`
- `aws-ec2-r5n`
- `aws-elastic-container-kubernetes`
- `aws-elastic-container-registry`
- `aws-elastic-container-service`
- `aws-genomics-cli`
- `aws-lightsail`
- `aws-app-runner`
- `aws-batch`
- `aws-compute-optimizer`
- `aws-elastic-beanstalk`
- `aws-fargate`
- `aws-lambda`
- `aws-local-zones`
- `aws-nitro-enclaves`
- `aws-outposts-family`
- `aws-outposts-rack`
- `aws-outposts-servers`
- `aws-parallelcluster`
- `aws-serverless-application-repository`
- `aws-thinkbox-deadline`
- `aws-thinkbox-frost`
- `aws-thinkbox-krakatoa`
- `aws-thinkbox-sequoia`
- `aws-thinkbox-stoke`
- `aws-thinkbox-xmesh`
- `aws-wavelength`
- `aws-bottlerocket`
- `aws-elastic-fabric-adapter`
- `aws-nice-dcv`
- `aws-nice-enginframe`
- `aws-vmware-cloud-on-aws`
- `aws-ecs-anywhere`
- `aws-eks-anywhere`
- `aws-eks-cloud`
- `aws-eks-distro`
- `aws-elastic-kubernetes-service`
- `aws-red-hat-openshift`

Database
- `aws-aurora`
- `aws-documentdb`
- `aws-dynamodb`
- `aws-elasticache`
- `aws-keyspaces`
- `aws-memorydb-for-redis`
- `aws-neptune`
- `aws-rds`
- `aws-rds-on-vmware`
- `aws-timestream`
- `aws-database-migration-service`

Developer Tools

- `aws-codeartifact`
- `aws-codebuild`
- `aws-codecommit`
- `aws-codedeploy`
- `aws-codepipeline`
- `aws-codestar`
- `aws-command-line-interface`
- `aws-tools-and-sdks`
- `aws-x-ray`

End User Computing
- `aws-appstream`
- `aws-worklink`
- `aws-workspaces`
- `aws-workspaces-web`

Front-End Web & Mobile
- `aws-amplify`
- `aws-device-farm`

Game Tech
- `aws-gamelift`
- `aws-gamesparks`
- `aws-gamekit`
- `aws-open-3d-engine`

IoT
- `aws-iot-1-click`
- `aws-iot-analytics`
- `aws-iot-button`
- `aws-iot-core`
- `aws-iot-device-defender`
- `aws-iot-device-management`
- `aws-iot-edukit`
- `aws-iot-events`
- `aws-iot-expresslink`
- `aws-iot-fleetwise`
- `aws-iot-greengrass`
- `aws-iot-roborunner`
- `aws-iot-sitewise`
- `aws-iot-things-graph`
- `aws-iot-twinmaker`
- `aws-freertos`

Machine Learning
- `aws-augmented-ai-a2i`
- `aws-codeguru`
- `aws-codewhisperer`
- `aws-comprehend`
- `aws-comprehend-medical`
- `aws-devops-guru`
- `aws-elastic-inference`
- `aws-forecast`
- `aws-fraud-detector`

- `aws-healthlake`
- `aws-kendra`
- `aws-lex`
- `aws-lookout-for-equipment`
- `aws-lookout-for-metrics`
- `aws-lookout-for-vision`
- `aws-monitron`
- `aws-personalize`
- `aws-polly`
- `aws-rekognition`
- `aws-sagemaker`
- `aws-sagemaker-ground-truth`
- `aws-sagemaker-studio-lab`
- `aws-textract`
- `aws-transcribe`
- `aws-translate`
- `aws-apache-mxnet-on-aws`
- `aws-deep-learning-amis`
- `aws-deep-learning-containers`
- `aws-deepcomposer`
- `aws-deeplens`
- `aws-deepracer`
- `aws-neuron`
- `aws-panorama`
- `aws-tensorflow-on-aws`
- `aws-torchserve`

Management & Governance
- `aws-cloudwatch`
- `aws-managed-grafana`
- `aws-managed-service-for-prometheus`
- `aws-appconfig`
- `aws-application-auto-scaling`
- `aws-auto-scaling`
- `aws-backint-agent`
- `aws-chatbot`
- `aws-cloudformation`
- `aws-cloudtrail`
- `aws-config`
- `aws-control-tower`
- `aws-distro-for-opentelemetry`
- `aws-fault-injection-simulator`
- `aws-launch-wizard`
- `aws-license-manager`
- `aws-management-console`
- `aws-opsworks`
- `aws-organizations`
- `aws-personal-health-dashboard`
- `aws-proton`
- `aws-resilience-hub`
- `aws-service-catalog`
- `aws-systems-manager`
- `aws-systems-manager-incident-manager`
- `aws-trusted-advisor`

- `aws-well-architected-tool`

Media Services
- `aws-elastic-transcoder`
- `aws-interactive-video-service`
- `aws-nimble-studio`
- `aws-elemental-appliances-&-software`
- `aws-elemental-conductor`
- `aws-elemental-delta`
- `aws-elemental-link`
- `aws-elemental-live`
- `aws-elemental-mediaconnect`
- `aws-elemental-mediaconvert`
- `aws-elemental-medialive`
- `aws-elemental-mediapackage`
- `aws-elemental-mediastore`
- `aws-elemental-mediatailor`
- `aws-elemental-server`

Migration & Transfer
- `aws-application-discovery-service`
- `aws-application-migration-service`
- `aws-datasync`
- `aws-mainframe-modernization`
- `aws-migration-evaluator`
- `aws-migration-hub`
- `aws-server-migration-service`
- `aws-transfer-family`

Networking & Content Delivery
- `aws-cloud-directory`
- `aws-cloudfront`
- `aws-route-53`
- `aws-virtual-private-cloud`
- `aws-app-mesh`
- `aws-client-vpn`
- `aws-cloud-map`
- `aws-cloud-wan`
- `aws-direct-connect`
- `aws-global-accelerator`
- `aws-private-5g`
- `aws-privatelink`
- `aws-site-to-site-vpn`
- `aws-transit-gateway`
- `aws-elastic-load-balancing`

Quantum Technologies
- `aws-braket`

Robotics
- `aws-robomaker`

Satellite
- `aws-ground-station`

Security, Identity, & Compliance
- `aws-cognito`
- `aws-detective`
- `aws-guardduty`
- `aws-inspector`
- `aws-macie`
- `aws-artifact`
- `aws-audit-manager`
- `aws-certificate-manager`
- `aws-cloudhsm`
- `aws-directory-service`
- `aws-firewall-manager`
- `aws-iam-identity-center`
- `aws-identity-and-access-management`
- `aws-key-management-service`
- `aws-network-firewall`
- `aws-resource-access-manager`
- `aws-secrets-manager`
- `aws-security-hub`
- `aws-shield`
- `aws-signer`
- `aws-waf`

Storage
- `aws-efs`
- `aws-elastic-block-store`
- `aws-fsx`
- `aws-fsx-for-lustre`
- `aws-fsx-for-netapp-ontap`
- `aws-fsx-for-openzfs`
- `aws-fsx-for-wfs`
- `aws-s3-on-outposts`
- `aws-simple-storage-service`
- `aws-simple-storage-service-glacier`
- `aws-backup`
- `aws-snowball`
- `aws-snowball-edge`
- `aws-snowcone`
- `aws-snowmobile`
- `aws-storage-gateway`
- `aws-cloudendure-disaster-recovery`

VR & AR
- `aws-sumerian`

Miscellaneous
- `aws-analytics`
- `aws-application-integration`
- `aws-blockchain`
- `aws-business-applications`
- `aws-cloud-financial-management`
- `aws-compute`
- `aws-containers`

- `aws-customer-enablement`
- `aws-database`
- `aws-developer-tools`
- `aws-end-user-computing`
- `aws-front-end-web-mobile`
- `aws-game-tech`
- `aws-internet-of-things`
- `aws-machine-learning`
- `aws-management-governance`
- `aws-media-services`
- `aws-migration-transfer`
- `aws-networking-content-delivery`
- `aws-quantum-technologies`
- `aws-robotics`
- `aws-satellite`
- `aws-security-identity-compliance`
- `aws-serverless`
- `aws-storage`
- `aws-vr-ar`

Here's the list of Azure Icons organized in markdown format:

```markdown
# Azure Icons List

## AI & Machine Learning
- `azure-batch-ai`
- `azure-machine-learning-studio-classic-web-services`
- `azure-genomics`
- `azure-translator-text`
- `azure-experimentation-studio`
- `azure-object-understanding`
- `azure-cognitive-services`
- `azure-genomics-accounts`
- `azure-machine-learning`
- `azure-machine-learning-studio-workspaces`
- `azure-machine-learning-studio-web-service-plans`
- `azure-applied-ai`
- `azure-language-services`
- `azure-log-analytics-workspaces`
- `azure-synapse-analytics`
- `azure-metrics-advisor`

## Application Development
- `azure-app-service-plans`
- `azure-app-service-certificates`
- `azure-app-service-domains`
- `azure-cdn-profiles`
- `azure-app-services`
- `azure-api-management-services`
- `azure-search-services`
- `azure-notification-hubs`
- `azure-app-service-environments`
- `azure-applens`
- `azure-azure-api-proxy`

## Compute
- `azure-virtual-machine`
- `azure-kubernetes-services`
- `azure-mesh-applications`
- `azure-availability-sets`
- `azure-disks-snapshots`
- `azure-virtual-machines-classic`
- `azure-function-apps`
- `azure-cloud-services-classic`
- `azure-batch-accounts`
- `azure-disks`
- `azure-images`
- `azure-vm-scale-sets`
- `azure-service-fabric-clusters`
- `azure-image-definitions`
- `azure-shared-image-galleries`
- `azure-container-instances`
```

- `azure-container-registries`
- `azure-spring-cloud`
- `azure-vm-images-classic`

Database
- `azure-sql-data-warehouses`
- `azure-sql`
- `azure-ssis-lift-and-shift-ir`
- `azure-purview-accounts`
- `azure-sql-edge`
- `azure-database-postgresql-server-group`
- `azure-cosmos-db`
- `azure-database-mysql-server`
- `azure-database-mariadb-server`
- `azure-sql-vm`
- `azure-data-factory`
- `azure-virtual-clusters`
- `azure-elastic-job-agents`
- `azure-sql-database`
- `azure-sql-server`
- `azure-database-migration-services`
- `azure-sql-elastic-pools`
- `azure-managed-database`
- `azure-sql-managed-instance`
- `azure-sql-server-stretch-databases`
- `azure-cache-redis`
- `azure-instance-pools`
- `azure-data-explorer-clusters`

DevOps & Monitoring
- `azure-devops`
- `azure-devtest-labs`
- `azure-lab-services`
- `azure-cost-management-and-billing`
- `azure-preview-features`
- `azure-subscriptions`
- `azure-service-health`
- `azure-log-streaming`
- `azure-application-insights`
- `azure-cloudtest`
- `azure-monitor`
- `azure-alerts`
- `azure-advisor`
- `azure-automation-accounts`
- `azure-activity-log`
- `azure-diagnostics-settings`
- `azure-troubleshoot`
- `azure-metrics`
- `azure-network-watcher`
- `azure-resource-explorer`
- `azure-operation-log-classic`
- `azure-dashboard-hub`
- `azure-scheduler-job-collections`

Identity & Security
- `azure-active-directory`
- `azure-ad-domain-services`
- `azure-groups`
- `azure-active-directory-connect-health`
- `azure-enterprise-applications`
- `azure-managed-identities`
- `azure-ad-b2c`
- `azure-information-protection`
- `azure-users`
- `azure-ad-identity-protection`
- `azure-app-registrations`
- `azure-ad-privilege-identity-management`
- `azure-identity-governance`
- `azure-tenant-properties`
- `azure-custom-azure-ad-roles`
- `azure-aad-licenses`
- `azure-identity-governance`

Networking
- `azure-virtual-networks`
- `azure-load-balancers`
- `azure-virtual-network-gateways`
- `azure-dns-zones`
- `azure-traffic-manager-profiles`
- `azure-network-watcher`
- `azure-network-security-groups`
- `azure-public-ip-addresses`
- `azure-route-filters`
- `azure-ddos-protection-plans`
- `azure-front-doors`
- `azure-application-gateways`
- `azure-local-network-gateways`
- `azure-expressroute-circuits`
- `azure-network-interfaces`
- `azure-connections`
- `azure-route-tables`
- `azure-firewalls`
- `azure-service-endpoint-policies`
- `azure-nat`
- `azure-virtual-wans`
- `azure-web-application-firewall-policies-waf`
- `azure-private-link`
- `azure-public-ip-prefixes`
- `azure-network-manager`
- `azure-private-link-service`
- `azure-private-link-hub`

Storage
- `azure-storage-accounts`
- `azure-storage-accounts-classic`
- `azure-blob-block`
- `azure-blob-page`
- `azure-storage-container`

- `azure-storage-queue`
- `azure-storsimple-device-managers`
- `azure-data-lake-storage-gen1`
- `azure-netapp-files`
- `azure-data-share-invitations`
- `azure-fileshare`
- `azure-storage-azure-files`

Web & Mobile
- `azure-web-app-+-database`
- `azure-windows-virtual-desktop`
- `azure-mobile`
- `azure-mobile-engagement`
- `azure-web-jobs`
- `azure-static-apps`
- `azure-notification-hub-namespaces`
- `azure-media-service`

Miscellaneous
- `azure-marketplace`
- `azure-free-services`
- `azure-resource-groups`
- `azure-tags`
- `azure-biz-talk`
- `azure-branch`
- `azure-error`
- `azure-file`
- `azure-folder-blank`
- `azure-folder-website`
- `azure-globe-error`
- `azure-globe-success`
- `azure-globe-warning`
- `azure-heart`
- `azure-location`
- `azure-media-file`
- `azure-power`
- `azure-signalr`
- `azure-toolbox`
- `azure-verifiable-credentials`
- `azure-intune`
- `azure-ebooks`
- `azure-client-apps`
- `azure-intune-for-education`
- `azure-intune-app-protection`
- `azure-device-security-apple`
- `azure-device-security-google`
- `azure-device-security-windows`
- `azure-iot-hub`
- `azure-iot-central-applications`
- `azure-maps-accounts`
- `azure-iot-edge`
- `azure-time-series-insights-event-sources`
- `azure-time-series-data-sets`
- `azure-device-compliance`

- `azure-software-updates`
- `azure-device-enrollment`
- `azure-exchange-access`
- `azure-device-configuration`
- `azure-security-baselines`
- `azure-device-provisioning-services`
- `azure-api-connections`
- `azure-spring-cloud`
- `azure-api-for-fhir`
- `azure-arc`
- `azure-application-gateways`
- `azure-virtual-networks-classic`
- `azure-application-security-groups`
- `azure-key-vaults`
- `azure-sentinel`
- `azure-extendedsecurityupdates`
- `azure-backup-center`
- `azure-disk-encryption-sets`
- `azure-disk-pool`
- `azure-service-providers`
- `azure-biz-talk`
- `azure-resource-graph-explorer`
- `azure-machinesazurearc`
- `azure-cloud-services-extended-support`
- `azure-migration-services`
- `azure-service-bus`
- `azure-stream-analytics-jobs`
- `azure-monitor-dashboard`
- `azure-ssh-keys`
- `azure-log-analytics-workspaces`
- `azure-operations-management`
- `azure-load-testing`
- `azure-container-app-environments`
- `azure-marketplace-management`
- `azure-connected-cache`
- `azure-connected-vehicle-platform`
- `azure-reserved-capacity`
- `azure-ceres`
- `azure-azurite`
- `azure-update-center`
- `azure-savings-plan`
- `azure-load-balancer-hub`
- `azure-modular-data-center`
- `azure-custom-ip-prefix`
- `azure-image-definition`
- `azure-image-version`
- `azure-vm-application-definition`
- `azure-vm-application-version`
- `azure-application-discovery-service`
- `azure-storage-explorer`
- `azure-data-box`
- `azure-data-box-edge`
- `azure-migrate`
- `azure-backup`

- `azure-app-configuration`
- `azure-partner-namespace`
- `azure-logic-apps`
- `azure-event-grid-topics`
- `azure-logic-apps-custom-connector`
- `azure-service-bus`
- `azure-time-series-insights-access-policies`
- `azure-logic-apps`
- `azure-partner-registration`
- `azure-event-grid-domains`
- `azure-integration-accounts`
- `azure-event-grid-subscriptions`
- `azure-consortium`
- `azure-system-topic`
- `azure-partner-topic

```
User [icon: user] {
  id Int pk
  username String
  email String
  avatar String
  createdDate DateTime
}

Booking [icon: clock] {
  id Int pk
  userId Int
  title String
  startTime DateTime
  endTime DateTime
  location String
  eventTypeId Int
  destinationCalendarId Int
}

EventType [icon: list] {
  id Int pk
  userId Int
  teamId Int
  hidden Boolean
  length Int
}

ApiKey [icon: key]{
  id String pk
  userId Int
  appId String
  hashedKey String
}

App [icon: grid] {
  slug String
  dirName String
  keys Json
  createdAt DateTime
}

Webhook [icon: link] {
  id String pk
  userId Int
  appId String
  active Boolean
}

DestinationCalender [icon: calendar] {
  id Int pk
  userId Int
  integration String
  eventTypeId Int
```

```
}

// Booking.eventtType < EventType.id
Webhook.appId > App.slug
Webhook.userId > User.id
// Webhook.eventTypeId > EventType.id
App.slug > ApiKey.appId
User.id < Booking.userId
EventType.userId <> User.id
User.id > ApiKey.userId
DestinationCalender.id > Booking.destinationCalendarId
DestinationCalender.userId < User.id
DestinationCalender.eventTypeId < EventType.id
```

```
// Define groups and nodes
Input data sources {
  Oracle [icon: oracle]
  Twitter [icon: twitter]
  Facebook [icon: facebook]
}
ETL pipeline [color: silver]{
  User survey data [icon: kafka]
  Data load [icon: aws-s3]
  Data transformation [icon: databricks]
  Data store [icon: snowflake]
}
Data destinations {
  Notification [icon: slack]
  Experimentation [icon: tensorflow]
  BI dashboard [icon: tableau]
}

// Define connections
Oracle, Twitter, Facebook > User survey data
User survey data > Data load > Data transformation > Data store
Data store > Notification, Experimentation, BI dashboard
```

```
User [icon: user] {
  id Int
  firstname String
  lastname String
  email String
  emailVerified DateTime
}

Form [icon: check-square] {
  id String
  ownerId Int
  name String
  formType FormType
  createdAt DateTime
}

Pipeline [icon: filter] {
  id String
  name String
  formId String
  events PipelineEvent
  createdAt DateTime
}

SessionEvent [icon: zap]{
  id String
  submissionSessionId String
  type String
  createdAt DateTime
}

SubmissionSession [icon: clock]{
  id String
  formId String
  createdAt DateTime
}

NoCodeForm [icon: check-square] {
  id String
  published Boolean
  closed Boolean
  formId String
}

User.id < Form.ownerId
Form.id < Pipeline.formId
Form.id < SubmissionSession.formId
Form.id < NoCodeForm.formId
SubmissionSession.id < SessionEvent.submissionSessionId
```

Here's the list of Google Cloud Icons organized in markdown format:

```markdown
# Google Cloud Icons List

## AI & Machine Learning
- `gcp-ai-hub`
- `gcp-ai-platform-unified`
- `gcp-ai-platform`
- `gcp-automl-natural-language`
- `gcp-automl-tables`
- `gcp-automl-translation`
- `gcp-automl-video-intelligence`
- `gcp-automl-vision`
- `gcp-automl`
- `gcp-document-ai`
- `gcp-vertexai`
- `gcp-video-intelligence-api`
- `gcp-cloud-inference-api`
- `gcp-healthcare-nlp-api`
- `gcp-tensorflow-enterprise`
- `gcp-text-to-speech`
- `gcp-speech-to-text`
- `gcp-recommendations-ai`

## API Management
- `gcp-api`
- `gcp-api-analytics`
- `gcp-api-monetization`
- `gcp-apigee-api-platform`
- `gcp-apigee-sense`
- `gcp-cloud-api-gateway`
- `gcp-cloud-apis`

## Application Development
- `gcp-app-engine`
- `gcp-cloud-build`
- `gcp-cloud-composer`
- `gcp-cloud-functions`
- `gcp-cloud-run-for-anthos`
- `gcp-cloud-run`
- `gcp-cloud-tasks`
- `gcp-cloud-test-lab`
- `gcp-cloud-shell`
- `gcp-cloud-code`
- `gcp-artifact-registry`

## Compute
- `gcp-compute-engine`
- `gcp-container-optimized-os`
- `gcp-container-registry`
- `gcp-cloud-gpu`
- `gcp-cloud-tpu`
```

- `gcp-local-ssd`
- `gcp-persistent-disk`

Data Analytics
- `gcp-analytics-hub`
- `gcp-bigquery`
- `gcp-bigtable`
- `gcp-data-catalog`
- `gcp-data-labeling`
- `gcp-data-layers`
- `gcp-data-loss-prevention-api`
- `gcp-data-qna`
- `gcp-data-studio`
- `gcp-data-transfer`
- `gcp-dataflow`
- `gcp-datalab`
- `gcp-dataplex`
- `gcp-datapol`
- `gcp-dataprep`
- `gcp-dataproc-metastore`
- `gcp-dataproc`
- `gcp-datashare`
- `gcp-datastore`
- `gcp-datastream`
- `gcp-genomics`
- `gcp-google-cloud-marketplace`

Database
- `gcp-cloud-spanner`
- `gcp-cloud-sql`
- `gcp-cloud-firestore`
- `gcp-cloud-datastore`
- `gcp-cloud-bigtable`
- `gcp-database-migration-service`
- `gcp-managed-service-for-microsoft-active-directory`

DevOps
- `gcp-cloud-deploy`
- `gcp-cloud-deployment-manager`
- `gcp-stackdriver`
- `gcp-profiler`
- `gcp-cloud-logging`
- `gcp-cloud-monitoring`
- `gcp-error-reporting`
- `gcp-cloud-trace`
- `gcp-cloud-audit-logs`

Identity & Security
- `gcp-identity-and-access-management`
- `gcp-identity-platform`
- `gcp-identity-aware-proxy`
- `gcp-key-management-service`
- `gcp-key-access-justifications`
- `gcp-binary-authorization`

- `gcp-security-command-center`
- `gcp-security-health-advisor`
- `gcp-security-key-enforcement`
- `gcp-security`
- `gcp-security-scanner`
- `gcp-cloud-ids`
- `gcp-cloud-armor`
- `gcp-cloud-hsm`
- `gcp-cloud-firewall-rules`
- `gcp-phishing-protection`

IoT
- `gcp-iot-core`
- `gcp-iot-edge`
- `gcp-fleet-engine`
- `gcp-fleet-routing-api`
- `gcp-contact-center-ai`

Networking
- `gcp-cloud-network`
- `gcp-cloud-vpn`
- `gcp-cloud-nat`
- `gcp-cloud-interconnect`
- `gcp-cloud-load-balancing`
- `gcp-cloud-dns`
- `gcp-cloud-domains`
- `gcp-cloud-router`
- `gcp-cloud-routes`
- `gcp-network-connectivity-center`
- `gcp-network-intelligence-center`
- `gcp-network-security`
- `gcp-network-tiers`
- `gcp-network-topology`
- `gcp-private-connectivity`
- `gcp-private-service-connect`
- `gcp-traffic-director`
- `gcp-service-discovery`

Management Tools
- `gcp-cloud-ops`
- `gcp-cloud-healthcare-api`
- `gcp-cloud-healthcare-marketplace`
- `gcp-cloud-inference-api`
- `gcp-cloud-for-marketing`
- `gcp-cloud-scheduler`
- `gcp-cloud-sql`
- `gcp-cloud-storage`
- `gcp-cloud-tasks`
- `gcp-cloud-test-lab`
- `gcp-cloud-vision-api`
- `gcp-cloud-logging`
- `gcp-cloud-monitoring`

Operations

- `gcp-operations`
- `gcp-profiler`
- `gcp-cloud-debugger`

Security & Identity
- `gcp-access-context-manager`
- `gcp-advanced-agent-modeling`
- `gcp-advanced-solutions-lab`
- `gcp-agent-assist`
- `gcp-billing`
- `gcp-certificate-authority-service`
- `gcp-certificate-manager`
- `gcp-security-command-center`

Serverless
- `gcp-cloud-functions`
- `gcp-cloud-run`

Storage
- `gcp-cloud-storage`
- `gcp-filestore`
- `gcp-storage-transfer`
- `gcp-backup-and-dr`

Miscellaneous
- `gcp-cloud-endpoints`
- `gcp-early-access-center`
- `gcp-financial-services-marketplace`
- `gcp-game-servers`
- `gcp-genomics`
- `gcp-launcher`
- `gcp-looker`
- `gcp-recommendations-ai`
- `gcp-retail-api`
- `gcp-stream-suite`
- `gcp-tools-for-powershell`
- `gcp-user-preferences`
- `gcp-workflows`
```

This markdown format organizes the Google Cloud Icons into categories, similar to the AWS icons list.
```

General Icons List

A - C
- `activity`
- `airplay`
- `alert-circle`
- `alert-octagon`
- `alert-triangle`
- `align-center`
- `align-justify`
- `align-left`
- `align-right`
- `anchor`
- `aperture`
- `archive`
- `arrow-down-circle`
- `arrow-down-left`
- `arrow-down-right`
- `arrow-down`
- `arrow-left-circle`
- `arrow-left`
- `arrow-right-circle`
- `arrow-right`
- `arrow-up-circle`
- `arrow-up-left`
- `arrow-up-right`
- `arrow-up`
- `at-sign`
- `award`
- `bar-chart-2`
- `bar-chart`
- `battery-charging`
- `battery`
- `bell-off`
- `bell`
- `bold`
- `book-open`
- `book`
- `bookmark`
- `briefcase`
- `calendar`
- `camera-off`
- `camera`
- `cast`
- `check-circle`
- `check-square`
- `check`
- `chevron-down`
- `chevron-left`
- `chevron-right`
- `chevron-up`
- `chevrons-down`
- `chevrons-left`

- `chevrons-right`
- `chevrons-up`
- `circle`
- `clipboard`
- `clock`
- `cloud-drizzle`
- `cloud-lightning`
- `cloud-off`
- `cloud-rain`
- `cloud-snow`
- `cloud`
- `code`
- `coffee`
- `columns`
- `command`
- `compass`
- `copy`
- `corner-down-left`
- `corner-down-right`
- `corner-left-down`
- `corner-left-up`
- `corner-right-down`
- `corner-right-up`
- `corner-up-left`
- `corner-up-right`
- `cpu`
- `credit-card`
- `crop`
- `crosshair`

D - F
- `database`
- `delete`
- `disc`
- `divide-circle`
- `divide-square`
- `divide`
- `dollar-sign`
- `download-cloud`
- `download`
- `dribbble`
- `droplet`
- `edit-2`
- `edit-3`
- `edit`
- `external-link`
- `eye-off`
- `eye`
- `fast-forward`
- `feather`
- `file-minus`
- `file-plus`
- `file-text`
- `file`

- `film`
- `filter`
- `flag`
- `folder-minus`
- `folder-plus`
- `folder`
- `framer`
- `frown`

G - I
- `gift`
- `git-branch`
- `git-commit`
- `git-merge`
- `git-pull-request`
- `globe`
- `grid`
- `hard-drive`
- `hash`
- `headphones`
- `heart`
- `help-circle`
- `hexagon`
- `home`
- `image`
- `inbox`
- `info`
- `italic`

J - L
- `key`
- `layers`
- `layout`
- `life-buoy`
- `link-2`
- `link`
- `list`
- `loader`
- `lock`
- `log-in`
- `log-out`
- `mail`
- `map-pin`
- `map`
- `maximize-2`
- `maximize`
- `meh`
- `menu`
- `message-circle`
- `message-square`
- `mic-off`
- `mic`
- `minimize-2`
- `minimize`

- `minus-circle`
- `minus-square`
- `minus`
- `monitor`
- `moon`
- `more-horizontal`
- `more-vertical`
- `mouse-pointer`
- `move`
- `music`

N - R
- `navigation-2`
- `navigation`
- `octagon`
- `package`
- `paperclip`
- `pause-circle`
- `pause`
- `pen-tool`
- `percent`
- `phone-call`
- `phone-forwarded`
- `phone-incoming`
- `phone-missed`
- `phone-off`
- `phone-outgoing`
- `phone`
- `pie-chart`
- `play-circle`
- `play`
- `plus-circle`
- `plus-square`
- `plus`
- `pocket`
- `power`
- `printer`
- `radio`
- `refresh-ccw`
- `refresh-cw`
- `repeat`
- `rewind`
- `rotate-ccw`
- `rotate-cw`

S - T
- `save`
- `scissors`
- `search`
- `send`
- `server`
- `settings`
- `share-2`
- `share`

- `shield-off`
- `shield`
- `shopping-bag`
- `shopping-cart`
- `shuffle`
- `sidebar`
- `skip-back`
- `skip-forward`
- `slash`
- `sliders`
- `smartphone`
- `smile`
- `speaker`
- `star`
- `stop-circle`
- `sun`
- `sunrise`
- `sunset`
- `tablet`
- `tag`
- `target`
- `terminal`
- `thermometer`
- `thumbs-down`
- `thumbs-up`
- `toggle-left`
- `toggle-right`
- `tool`
- `trash-2`
- `trash`
- `trello`
- `trending-down`
- `trending-up`
- `triangle`
- `truck`
- `tv`
- `twitch`
- `twitter`
- `type`

U - Z
- `umbrella`
- `underline`
- `unlock`
- `upload-cloud`
- `upload`
- `user-check`
- `user-minus`
- `user-plus`
- `user-x`
- `user`
- `users`
- `video-off`
- `video`

- `voicemail`
- `volume-1`
- `volume-2`
- `volume-x`
- `volume`
- `watch`
- `wifi-off`
- `wifi`
- `wind`
- `x-circle`
- `x-octagon`
- `x-square`
- `x`
- `zap-off`
- `zap`
- `zoom-in`
- `zoom-out`

```
// Define groups and nodes
Cloud Provider API [icon: settings]
AWS [icon: aws]
GCP [icon: google-cloud]
Azure [icon: azure]
Control Plane [icon: k8s-control-plane]{
  api [icon: k8s-api]
  sched [icon: k8s-sched]
  ccm [icon: k8s-c-c-m]
  cm [icon: k8s-c-m]
  etcd [icon: k8s-etcd]
}
Node1 [icon: k8s-node] {
  kubelet1 [icon: k8s-kubelet]
  kproxy1 [icon: k8s-k-proxy]
}
Node2 [icon: k8s-node] {
  kubelet2 [icon: k8s-kubelet]
  kproxy2 [icon: k8s-k-proxy]
}
Node3 [icon: k8s-node] {
  kubelet3 [icon: k8s-kubelet]
  kproxy3 [icon: k8s-k-proxy]
}

// Define connections
ccm > Cloud Provider API
Cloud Provider API > AWS, Azure, GCP
api > ccm, sched, etcd, cm
kubelet1, kproxy1, kubelet2, kproxy2, kubelet3, kproxy3 > api
```

Kubernetes Icons List

Core Components
- `k8s-kubernetes`: Kubernetes
- `k8s-api`: API server
- `k8s-c-c-m`: Cloud controller manager
- `k8s-c-m`: Controller manager
- `k8s-k-proxy`: Kubernetes proxy
- `k8s-kubelet`: Kubelet
- `k8s-sched`: Scheduler
- `k8s-control-plane`: Control plane
- `k8s-node`: Node
- `k8s-etcd`: etcd

Roles & Access Control
- `k8s-c-role`: ClusterRole
- `k8s-crb`: ClusterRoleBinding
- `k8s-rb`: RoleBinding
- `k8s-role`: Role
- `k8s-sa`: ServiceAccount
- `k8s-user`: User

Workloads
- `k8s-deploy`: Deployment
- `k8s-ds`: DaemonSet
- `k8s-cronjob`: CronJob
- `k8s-job`: Job
- `k8s-rs`: ReplicaSet
- `k8s-sts`: StatefulSet
- `k8s-pod`: Pod

Configuration & Storage
- `k8s-cm`: ConfigMap
- `k8s-secret`: Secret
- `k8s-pv`: PersistentVolume
- `k8s-pvc`: PersistentVolumeClaim
- `k8s-sc`: StorageClass
- `k8s-vol`: Volume

Networking & Security
- `k8s-ing`: Ingress
- `k8s-netpol`: Network policy
- `k8s-psp`: PodSecurityPolicy
- `k8s-svc`: Service

Autoscaling & Monitoring
- `k8s-hpa`: HorizontalPodAutoscaler
- `k8s-limits`: Limits
- `k8s-quota`: Quota

Custom Resources
- `k8s-crd`: CustomRoleDefinition

Namespaces & Groups
- `k8s-ns`: Namespace
- `k8s-group`: Group

Miscellaneous
- `k8s-ep`: Endpoint

```
// Define groups and nodes
AD tenant [icon: azure-active-directory]
Load Balancers [icon: azure-load-balancers]
Virtual Network [icon: azure-virtual-networks] {
  Web Tier [icon: azure-network-security-groups] {
    vm1 [icon: azure-virtual-machine]
    vm2 [icon: azure-virtual-machine]
    vm3 [icon: azure-virtual-machine]
  }
  Business Tier [icon: azure-network-security-groups] {
    lb2 [icon: azure-load-balancers]
    vm4 [icon: azure-virtual-machine]
    vm5 [icon: azure-virtual-machine]
    vm6 [icon: azure-virtual-machine]
  }
}

// Define connections
AD tenant > Load Balancers
Load Balancers > vm1, vm2, vm3
vm1, vm2, vm3 > lb2 > vm4, vm5, vm6
```

```
// Define groups and nodes
API gateway [icon: aws-api-gateway]
Lambda [icon: aws-lambda]
S3 [icon: aws-simple-storage-service]
VPC Subnet {
  Main Server {
    Server [icon: aws-ec2]
    Data [icon: aws-rds]
  }
  Queue [icon: aws-auto-scaling]
  Compute Nodes {
    Worker1 [icon: aws-ec2]
    Worker2 [icon: aws-ec2]
    Worker3 [icon: aws-ec2]
  }
}
Analytics [icon: aws-redshift]

// Define connections
API gateway > Lambda > Server > Data
Server > Queue
Queue > Worker1, Worker2, Worker3
S3 < Data
Compute Nodes > Analytics
```

```
users [icon: user, color: blue] {
  id string pk
  displayName string
  team_role string
  teams string
}

teams [icon: users, color: blue] {
  id string pk
  name string
}

workspaces [icon: home] {
  id string
  createdAt timestamp
  folderId string
  teamId string
}

folders [icon: folder] {
  id string
  name string
}

chat [icon: message-circle, color: green] {
  duration number
  startedAt timestamp
  endedAt timestamp
  workspaceId string
}

invite [icon: mail, color: green] {
  inviteId string
  type string
  workspaceId string
  inviterId string
}

users.teams <> teams.id
workspaces.folderId > folders.id
workspaces.teamId > teams.id
chat.workspaceId > workspaces.id
invite.workspaceId > workspaces.id
invite.inviterId > users.id
```

```
// Define groups and nodes
Stream [icon: kafka, color: grey]
Ingest {
  Pub/Sub [icon: gcp-pubsub]
  Logging [icon: gcp-cloud-logging]
}
Pipelines {
  Dataflow [icon: gcp-dataflow]
}
Storage [icon: gcp-cloud-storage] {
  Datastore [icon: gcp-datastore]
  Bigtable [icon: gcp-bigtable]
}
Analytics {
  BigQuery [icon: gcp-bigquery]
}
Application [icon: gcp-app-engine] {
  App Engine [icon: gcp-app-engine]
  Container Engine [icon: gcp-container-registry]
  Compute Engine [icon: gcp-compute-engine]
}

// Define connections
Stream > Ingest
Logging > Analytics > Application
Pub/Sub > Pipelines > Storage > Application
```

```
Web App [icon: layout] > DB [icon: database]: Start transaction
Web App > Cloud Fx [icon: function]: Call function
Cloud Fx > API [icon: cloud-cog]: Create session
API > Cloud Fx: Session info
Cloud Fx > DB: Create tx record
Cloud Fx > API: Request access token
API > Cloud Fx: Access token
Cloud Fx > Web App: Token and transaction info
Web App > API: Complete transaction
alt [label: If successful]{
  API > Web App: Transaction confirmation
}
else [label: If failed]{
  API > Web App: Transaction cancellation
}
Web App > DB: Create tx record
Web App > API: Subscribe to transaction changes
activate API
API > API: Ongoing events
API > Web App: Push events
deactivate API
```

Sequence Diagram Syntax Guide

Basic Structure
Each line in a sequence diagram consists of two columns (entities), an arrow (direction of flow), and a message. The two columns are separated by the `>` arrow, and the message is prepended with a `:`.

- **Example**:
  ```text
  Web App > DB: Start transaction
  ```

Arrows
Arrows indicate the direction of flow between entities.

Arrow	Syntax	Description
`>`	`>`	Left-to-right arrow
`<`	`<`	Right-to-left arrow
`<->`	`<>`	Bi-directional arrow
`-`	`-`	Line
`--`	`--`	Dotted line
`-->`	`-->`	Dotted arrow

- Each line is parsed in sequential order from top to bottom.
- Column names must be unique. New columns are created if a line refers to a new name.

Properties
Properties are key-value pairs enclosed in `[]` brackets that can be appended to column names. Properties are optional.

- **Example**:
  ```text
  Web App [icon: monitor, color: blue] > DB [icon: database, color: green]: Start transaction
  ```

Allowed Properties

Property	Description	Value	Default Value
`icon`	Attached icons	Icon names (e.g., aws-ec2)	
`color`	Stroke and fill color	Color name (e.g., blue) or hex code	
`label`	Text label	Any string, enclosed in double quotes if containing a space.	Name of column
`colorMode`	Fill color lightness	pastel, bold, outline	pastel
`styleMode`	Embellishments	shadow, plain, watercolor	shadow
`typeface`	Text typeface	rough, clean, mono	rough

- **Label Property**: Useful when you want the column's label and name to be distinct.
  - **Example**:
    ```text
    Server1 [label: server]
    Server2 [label: server]
    ```

Blocks

Blocks represent control flow and can be used to express loops, if-else logic, parallel processing, and break execution.

- **Definition**: Blocks are defined with a block type followed by `{}`. They can include an optional `label` property.
- **Example**:
  ```text
  opt [label: if complete] {
    Server > Client: Success
  }
  ```

Block Types

Type	Description
`loop`	Loop
`alt`	Alternative (else)
`opt`	Optional
`par`	Parallel (and)
`break`	Break

- **Connected Blocks**: In the case of `alt` (paired with `else`) and `par` (paired with `and`), blocks can be connected.
  - **Example**:
    ```text
    alt [label: if complete] {
      Server > Client: Success
    }
    else [label: if failed] {
      Server > Client: Failure
    }
    ```

Block Properties

Property	Description	Value
`label`	Adds a label to the block	Block label (any string)
`icon`	Adds an icon to the block label	Icon names (e.g., aws-ec2)
`color`	Specifies a color for the block	Color name or hex code

Activations
Activations represent the time during which a column (an actor or resource) is actively performing an action.

- **Definition**: A pair of `activate` and `deactivate` statements define a single activation.
- **Example**:
  ```text
  Client > Server: Data request
  activate Server
  Server > Client: Return data
  deactivate Server
  ```

Escape String
Certain reserved characters are not allowed in column names. To use these characters, wrap the entire column name in quotes `" "`.

- **Example**:
  ```text
  User > "https://localhost:8080": GET
  ```

Direction
The direction of the sequence diagram can be changed using the `direction` statement. Allowed directions are:

- `direction down`
- `direction up`
- `direction right` (default)
- `direction left`

- **Example**:
  ```text
  direction down
  ```

Styling
Styles can be applied at the diagram level.

Style Properties

Property	Values	Default Value	Syntax Example
`colorMode`	pastel, bold, outline	pastel	`colorMode bold`
`styleMode`	shadow, plain, watercolor	shadow	`styleMode shadow`
`typeface`	rough, clean, mono	rough	`typeface clean`

Cloud Architecture Diagram Syntax Guide

Nodes
A node is the most basic building block in a cloud architecture diagram.

- **Definition**: Nodes are defined with a name followed by an optional set of properties.
- **Example**:
  ```text
  compute [icon: aws-ec2]
  ```
- **Properties**:
  - **Icon**: Assigns an icon to the node (e.g., `aws-ec2`).
  - **Color**: Sets the color of the node.
- **Uniqueness**: Node names must be unique.

Groups
A group is a container that can encapsulate nodes and other groups.

- **Definition**: Groups are defined with a name followed by `{ }`.
- **Example**:
  ```text
  Main Server {
    Server [icon: aws-ec2]
    Data [icon: aws-rds]
  }
  ```
- **Nesting**: Groups can be nested within each other.
  - Example:
    ```text
    VPC Subnet {
      Main Server {
        Server [icon: aws-ec2]
        Data [icon: aws-rds]
      }
    }
    ```
- **Properties**:
  - **Icon**: Assigns an icon to the group.
  - **Color**: Sets the color of the group.
- **Uniqueness**: Group names must be unique.

Properties
Properties are key-value pairs enclosed in `[]` that can be appended to node and group definitions. Properties are optional.

- **Example**:
  ```text
  Main Server [icon: aws-ec2, color: blue] {
    Server [icon: aws-ec2]
    Data [icon: aws-rds]
  }
  ```

- **Allowed Properties**:

Property	Description	Value	Default Value
`icon`	Attached icons	Icon names (e.g., aws-ec2)	
`color`	Stroke and fill color	Color name (e.g., blue) or hex code (e.g., #000000)	
`label`	Text label	Any string. Enclose in double quotes if containing a space.	Name of node or group
`colorMode`	Fill color lightness	pastel, bold, outline	pastel
`styleMode`	Embellishments	shadow, plain, watercolor	shadow
`typeface`	Text typeface	rough, clean, mono	rough

Usage of `label` Property
The `label` property is useful if you want the node's (or group's) label and name to be distinct. By default, the label is set as the node name. Use the `label` property if you have two nodes with the same label.

- **Example**:
  ```text
  Server_A [label: server]
  Server_B [label: server]
  ```

You can set multiple properties by separating them with commas:
```text
Server [icon: server, typeface: mono]
```

Connections
Connections represent relationships between nodes and groups. They can be created between nodes, between groups, and between nodes and groups.

- **Example**:
  ```text
  Compute > Storage
  ```
- **Types of Connectors**:

Connector	Syntax	Description
`>`	`>`	Left-to-right arrow
`<`	`<`	Right-to-left arrow
`<->`	`<>`	Bi-directional arrow
`-`	`-`	Line
`--`	`--`	Dotted line
`-->`	`-->`	Dotted arrow

Connection Labels
It is possible to add a label to a connection.
- **Example**:
  ```text
  Storage > Server: Cache Hit
  ```

One-to-Many Connections
You can create one-to-many connections in a single statement:

- **Example**:
  ```text
  Server > Worker1, Worker2, Worker3
  ```

If a connection statement contains a name not previously defined as a node or group, a blank node with that name will be created.

Escape String
Certain characters are reserved and not allowed in node or group names. To use these characters, wrap the entire name in quotes `" "`.

- **Example**:
  ```text
  User > "https://localhost:8080": GET
  ```

Direction
The direction of the cloud architecture diagram can be changed using the `direction` statement. Allowed directions are:

- `direction down`
- `direction up`
- `direction right` (default)
- `direction left`

The `direction` statement can be placed anywhere in the code:
- **Example**:
  ```text
  direction down
  ```

Styling
Styles can be applied at the diagram level.

Style Properties

Property	Values	Default Value	Syntax Example
`colorMode`	pastel, bold, outline	pastel	`colorMode bold`
`styleMode`	shadow, plain, watercolor	shadow	`styleMode shadow`
`typeface`	rough, clean, mono	rough	`typeface clean`

Eraser Syntax Guide

Nodes
A node is the most basic building block in a flow chart.

- **Definition**: A node is the most basic building block in a flow chart.
- **Syntax**:
  ```text
  NodeName [property: value]
  ```
  Example: `Start [shape: oval]`
- **Properties**:
  - **Shape**: Defines the shape of the node (e.g., oval).
  - **Icon**: Assigns an icon to the node.
  - **Color**: Sets the color of the node.
  - **Label**: Adds a label to the node.
- **Uniqueness**: Node names must be unique.

Groups
A group is a container that can encapsulate nodes and other groups.

- **Definition**: A group is a container that can encapsulate nodes and other groups.
- **Syntax**:
  ```text
  GroupName {
    Node1, Node2, Node3
  }
  ```
  Example:
  ```text
  Loop {
    Issue1, Issue2, Issue3
  }
  ```
  Alternatively, nodes can be separated by new lines:
  ```text
  Loop {
    Issue1
    Issue2
    Issue3
  }
  ```
- **Nesting**: Groups can be nested within each other.
  - Example:
    ```text
    Outer Loop {
      Inner Loop {
        Issue1
        Issue2
      }
      Issue3
    }
    ```

- **Properties**:
  - **Icon**: Assigns an icon to the group.
  - **Color**: Sets the color of the group.
  - **Label**: Adds a label to the group.
- **Uniqueness**: Group names must be unique.

Properties
Properties are key-value pairs enclosed in `[]` brackets that can be appended to definitions of nodes and groups. Properties are optional.

Allowed Properties

Property	Description	Value	Default Value
`shape`	Shape of node	Shape names (e.g., diamond or oval)	rectangle
`icon`	Icon	Icon names (e.g., aws-ec2)	
`color`	Stroke and fill color	Color name (e.g., blue) or hex code (e.g., #000000)	
`label`	Text label	Any string. Enclose in double quotes if containing a space.	Name of node or group
`colorMode`	Fill color lightness	pastel, bold, outline	pastel
`styleMode`	Embellishments	shadow, plain, watercolor	shadow
`typeface`	Text typeface	rough, clean, mono	rough

Shapes List
- rectangle (default)
- cylinder
- diamond
- document
- ellipse
- hexagon
- oval
- parallelogram
- star
- trapezoid
- triangle

Usage of `label` Property
The `label` property is useful if you want the node's (or group's) label and name to be distinct. By default, the label is set as the node name. But because node names are required to be distinct, you will need to use the `label` property if you have two nodes with the exact same label.

Example:
```text
Start_A [label: start]
Start_B [label: start]
```

You can set multiple properties by separating them using commas:
```text
Start [shape: oval, icon: flag]
```

Relationships
Connections represent relationships between nodes and groups. They can be created between nodes, between groups,

and between nodes and groups.

Example of Connection Between Two Nodes:
```text
Issue > Bug
```

Types of Connectors:

Connector	Syntax	Description
`>`	`>`	Left-to-right arrow
`<`	`<`	Right-to-left arrow
`<->`	`<>`	Bi-directional arrow
`-`	`-`	Line
`--`	`--`	Dotted line
`-->`	`-->`	Dotted arrow

Relationship Label
It is possible to add a label to a relationship. Example:
```text
Issue > Bug: Triage
```

Branching Relationships
It is possible to create one-to-many connections in a single statement. Example:
```text
Issue > Bug, Feature
```

Chained Relationships
It is also possible to "chain" a sequence of relationship statements in a single statement. Example:
```text
Issue > Bug > Duplicate?
```

If a connection statement contains a name that has not been previously defined as a node or a group, a blank node with that name will be created.

Escape String
Certain characters are not allowed in node and group names because they are reserved. To use these characters, you can wrap the entire node or group name in quotes `" "`.

Example:
```text
User > "https://localhost:8080": GET
```

Direction
The direction of the flow chart can be changed using the `direction` statement. Allowed directions are:

- `direction down` (default)
- `direction up`
- `direction right`

- `direction left`

The direction statement can be placed anywhere in the code like this:
```text
direction right
```

Styling
Styles can be applied at the diagram level. Below is an overview of the options and syntax.

Style Properties

Property	Values	Default value	Syntax example
`colorMode`	pastel, bold, outline	pastel	`colorMode bold`
`styleMode`	shadow, plain, watercolor	shadow	`styleMode shadow`
`typeface`	rough, clean, mono	rough	`typeface clean`

ERD Syntax Guide

Entities
Entities correspond to database tables or similar. They contain attributes.

- **Definition**: Entities are defined with a name followed by `{ }`.
- **Example**:
  ```text
  users {
    id string
    displayName string
  }
  ```
  Entities can also be empty:
  ```text
  users { }
  ```
- **Uniqueness**: Entity names must be unique.

Attributes
Attributes correspond to database table columns or similar.

- **Definition**: Attributes are defined within an entity with a name, type (optional), and metadata (optional), separated by spaces.
- **Example**:
  ```text
  users {
    id string pk
  }
  ```
- **Referring to Attributes**: Outside of an entity definition, attributes are referred to by following the entity name, separated by a `.`.
  - Example:
    ```text
    users.teamId > teams.id
    ```
- **Inline Relationship**: You can define an attribute and create a relationship in the same line within the entity definition.
  - Example:
    ```text
    users {
      teamId < teams.id
    }
    ```

Properties
Properties are key-value pairs enclosed in `[]` that can be appended to entity definitions. Properties are optional.

- **Example**:
  ```text
  users [icon: user, color: blue] {
    teamId < teams.id
  ```

  }
  ```
- **Allowed Properties**:

| Property | Description | Value | Default Value |
|---|---|---|---|
| `icon` | Attached icons | Icon names (e.g., aws-ec2) | |
| `color` | Stroke and fill color | Color name (e.g., blue) or hex code | |
| `colorMode` | Fill color lightness | pastel, bold, outline | pastel |
| `styleMode` | Embellishments | shadow, plain, watercolor | shadow |
| `typeface` | Text typeface | rough, clean, mono | rough |

## Relationships
Relationships represent attribute-level relations between entities.

- **Example**:
  ```text
  users.teamId > teams.id
  ```
- **Entity-Level Relations**: It is possible to show entity-level relations without specifying attributes.
  - Example:
    ```text
    users > teams
    ```
- **Cardinality Types**:

Connector	Syntax	Description
`<`	`<`	One-to-many
`>`	`>`	Many-to-one
`-`	`-`	One-to-one
`<->`	`<>`	Many-to-many

- **Auto-Creation**: If a relationship statement contains a name not previously defined as an entity or attribute, it will be automatically created.

Escape String
Certain reserved characters are not allowed in entity or attribute names. To use these characters, wrap the entire name in quotes `" "`.

- **Example**:
  ```text
  "CI/CD" [icon: gear] {
    id string pk
  }
  ```

Styling
Styles can be applied at the diagram level. Below are the options and syntax:

Property	Values	Default Value	Syntax Example
`colorMode`	pastel, bold, outline	pastel	`colorMode bold`
`styleMode`	shadow, plain, watercolor	shadow	`styleMode shadow`
  ```

| `typeface` | rough, clean, mono | rough | `typeface clean` |

Tech Logos List

A - B
- `adobe`
- `airflow`
- `airplay-audio`
- `airplay-video`
- `algolia`
- `alibaba-cloud`
- `alibaba`
- `alipay`
- `amazon`
- `alexa`
- `amp`
- `android`
- `angular`
- `ant`
- `apache`
- `apollo-graphql`
- `apple`
- `apple-pay`
- `apple-podcasts`
- `app-store`
- `arduino`
- `assemblyscript`
- `atlassian`
- `auth0`
- `authy`
- `babel`
- `bitcoin`
- `bluetooth`
- `bootstrap`
- `box`
- `brave`
- `bytedance`

C - D
- `chromecast`
- `circleci`
- `clojure`
- `cloudflare`
- `cockroach-labs`
- `codepen`
- `codesandbox`
- `coffeescript`
- `confluence`
- `couchbase`
- `cpanel`
- `css3`
- `cypress`
- `dart`
- `databricks`
- `datadog`

- `dbt`
- `debian`
- `deno`
- `discord`
- `django`
- `docker`
- `dot-net`
- `dropbox`
- `drupal`
- `dynamics-365`

E - G
- `eclipse-ide`
- `elastic`
- `elasticsearch`
- `electron`
- `elixir`
- `eslint`
- `ethereum`
- `facebook`
- `fastly`
- `figma`
- `firebase`
- `firefox`
- `flask`
- `flutter`
- `gatsby`
- `git`
- `github`
- `github-actions`
- `gitlab`
- `gmail`
- `gnome`
- `gnu`
- `gnu-bash`
- `gnu-emacs`
- `go`
- `google`
- `google-analytics`
- `google-calendar`
- `chrome`
- `google-cloud`
- `google-drive`
- `google-maps`
- `google-meet`
- `google-sheets`
- `google-tag-manager`
- `grafana`
- `graphql`
- `groovy`

H - J
- `haskell`
- `hasura`

- `heroku`
- `homebrew`
- `html5`
- `hubspot`
- `ibm`
- `ibm-cloud`
- `ibm-watson`
- `instagram`
- `intellij-idea`
- `intercom`
- `internet-explorer`
- `ios`
- `jamstack`
- `javascript`
- `jekyll`
- `jenkins`
- `jest`
- `jetbrains`
- `jira`
- `jquery`
- `json`
- `jupyter`

K - L
- `kafka`
- `kibana`
- `kotlin`
- `kubernetes`
- `laravel`
- `linkedin`
- `linux`
- `lodash`
- `looker`
- `loom`

M - N
- `magento`
- `mapbox`
- `mariadb`
- `markdown`
- `marketo`
- `messenger`
- `meta`
- `meteor`
- `microsoft`
  - `access`
  - `azure`
  - `bing`
  - `edge`
  - `excel`
  - `exchange`
  - `office`
  - `outlook`
  - `powerpoint`

  - `sharepoint`
  - `sql-server`
  - `teams`
  - `word`
- `mongodb`
- `mozilla`
- `mysql`
- `neo4j`
- `netlify`
- `next`
- `nginx`
- `nintendo`
- `node`
- `npm`

O - P
- `oculus`
- `okta`
- `oracle`
- `perl`
- `php`
- `playstation`
- `postgres`
- `postman`
- `power-bi`
- `powershell`
- `prisma`
- `pulumi`
- `puppeteer`
- `python`
- `pytorch`

R - S
- `rabbitmq`
- `railway`
- `raspberry-pi`
- `react`
- `red-hat`
- `redis`
- `redux`
- `rocketmq`
- `rss`
- `rstudio`
- `ruby-on-rails`
- `rust`
- `safari`
- `salesforce`
- `sap`
- `scala`
- `sentry`
- `shopify`
- `slack`
- `snowflake`
- `solr`

- `spark`
- `splunk`
- `sqlite`
- `square`
- `stripe`
- `svelte`
- `swagger`
- `swift`

T - V
- `tableau`
- `tencent-qq`
- `tensorflow`
- `terraform`
- `tomcat`
- `typescript`
- `ubuntu`
- `unity`
- `vercel`
- `vite`
- `vs-code`
- `vue`

W - Z
- `webassembly`
- `webflow`
- `webgl`
- `webpack`
- `webrtc`
- `wechat`
- `whatsapp`
- `windows`
- `wordpress`
- `xbox`
- `youtube`
- `zendesk`
- `zoom`