

fmx-instance

Version 0.1.32

Values

Key	Type	Default
argoCdNamespace	string	"argocd"
backup.accessKey	string	"minio"
backup.bucketName	string	"backup"
backup.enabled	bool	false
backup.endpoint	string	"http://minio"
backup.region	string	"auto"
backup.secretKey	string	"minio123"
caCertsConfigMap.data	string	""
caCertsConfigMap.enabled	bool	false
caCertsConfigMap.existingConfigMap	bool	false
caCertsConfigMap.fileName	string	"ca-certificates.crt"
caCertsConfigMap.nameOverride	string	""
chartRepoUrl	string	"https://nexus.firemetricshealth.com/re firemetrics/"
components.bootstrap.enabled	bool	true
components.bootstrap.firemetrics.workerCount	int	1
components.bootstrap.securityContext	object	{}
components.bootstrap.valuesOverride	object	{}
components.fuego.enabled	bool	true

components.fuego.env	list	[]
components.fuego.image	string	"ghcr.io/firemetrics/fuego:dfb02edc"
components.fuego.oidcClientSecret.idKey	string	"id"
components.fuego.oidcClientSecret.name	string	"fuego-oidc-client"
components.fuego.oidcClientSecret.secretKey	string	"secret"
components.fuego.securityContext	object	{}
components.fuego.valuesOverride	object	{}
components.fuego.volumeMounts	list	[]
components.fuego.volumes	list	[]
components.ingress.enabled	bool	true
components.ingress.valuesOverride	object	{}
components.keycloak.adminSecret.name	string	"keycloak-admin"
components.keycloak.adminSecret.passwordKey	string	"password"
components.keycloak.adminSecret.usernameKey	string	"username"
components.keycloak.databaseName	string	"keycloak"
components.keycloak.databaseUserSecret.name	string	"keycloak-db-user"
components.keycloak.databaseUserSecret.passwordKey	string	"password"
components.keycloak.databaseUserSecret.usernameKey	string	"username"
components.keycloak.enabled	bool	true
components.keycloak.env	list	[]
components.keycloak.image	string	"quay.io/keycloak/keycloak:26.3.2"
components.keycloak.importRealm	bool	true
components.keycloak.publicPath	string	"/auth"
components.keycloak.securityContext	object	{}
components.keycloak.valuesOverride	object	{}
components.keycloak.volumeMounts	list	[]
components.keycloak.volumes	list	[]
components.minio.chartRepoUrl	string	"https://operator.min.io"
components.minio.enabled	bool	false
components.minio.size	string	"10Gi"

components.minio.valuesOverride	object	{}
components.panel.datasetImportDisabled	bool	false
components.panel.enabled	bool	true
components.panel.env	list	[]
components.panel.image	string	"ghcr.io/firemetrics/fmx-panel:ec65075"
components.panel.nocodeBuilderDisabled	bool	false
components.panel.oidc.audience	string	""
components.panel.oidc.clientId	string	"panel"
components.panel.publicPath	string	"/panel"
components.panel.resourceExplorerDisabled	bool	false
components.panel.securityContext	object	{}
components.panel.sqlEditorDisabled	bool	false
components.panel.valuesOverride	object	{}
components.panel.volumeMounts	list	[]
components.panel.volumes	list	[]
components.postgres.clusterNameOverride	string	""
components.postgres.enabled	bool	true
components.postgres.extraEnv	list	[]
components.postgres.numberOfInstances	int	1
components.postgres.podServiceAccountNameOverride	string	""
components.postgres.postgresql	object	{}
components.postgres.resources.limits.cpu	string	"32"
components.postgres.resources.limits.memory	string	"128Gi"
components.postgres.resources.requests.cpu	string	"10m"
components.postgres.resources.requests.memory	string	"100Mi"
components.postgres.spiloFSGroup	int	103
components.postgres.spiloRunAsGroup	int	103
components.postgres.spiloRunAsUser	int	101
components.postgres.teamId	string	""

components.postgres.valuesOverride	object	{}
components.postgres.volume.size	string	"10Gi"
database.dbname	string	"firemetrics"
database.existingSuperUserSecret	string	""
database.hostnameOverride	string	""
database.port	int	5432
hostname	string	"example.com"
imagePullSecret	string	""
oidc.discoveryUrlOverride	string	""
oidc.enabled	bool	true
oidc.keycloakRealm	string	"firemetrics"
spiloImage	string	"ghcr.io/firemetrics/spilo17:4.0-p2-f04"
syncPolicy	object	{"automated":{"prune":true,"selfHeal":t
tls.certManager.enabled	bool	false
tls.certManager.issuer	string	"letsencrypt"
tls.enabled	bool	false
tls.existingSecret	bool	false
tls.secretNameOverride	string	""

