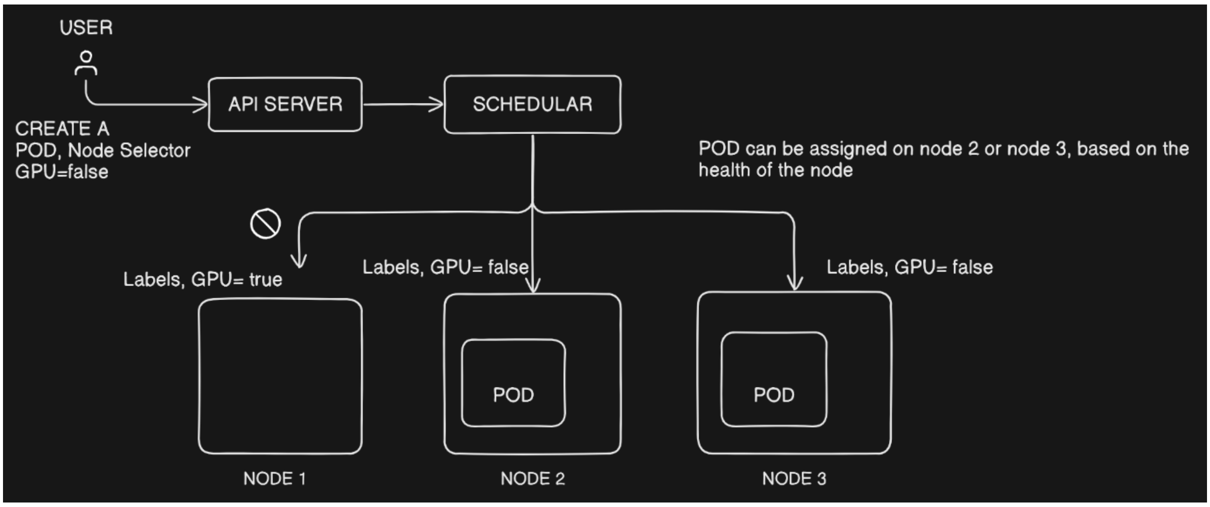
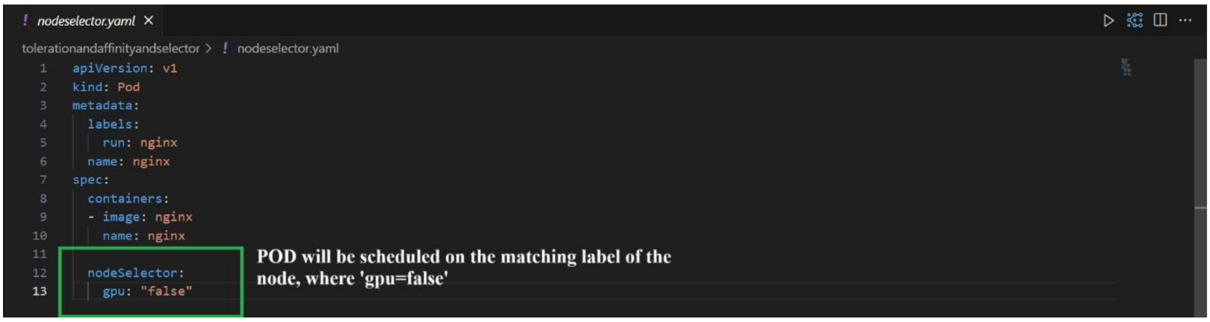
# NODE SELECTOR AND NODE AFFINITY

**Node Selector** is a simple way to constrain which nodes a pod can be scheduled on based on node labels. It allows you to ensure that a pod is only scheduled on nodes that match specific criteria.

### How Node Selector Works

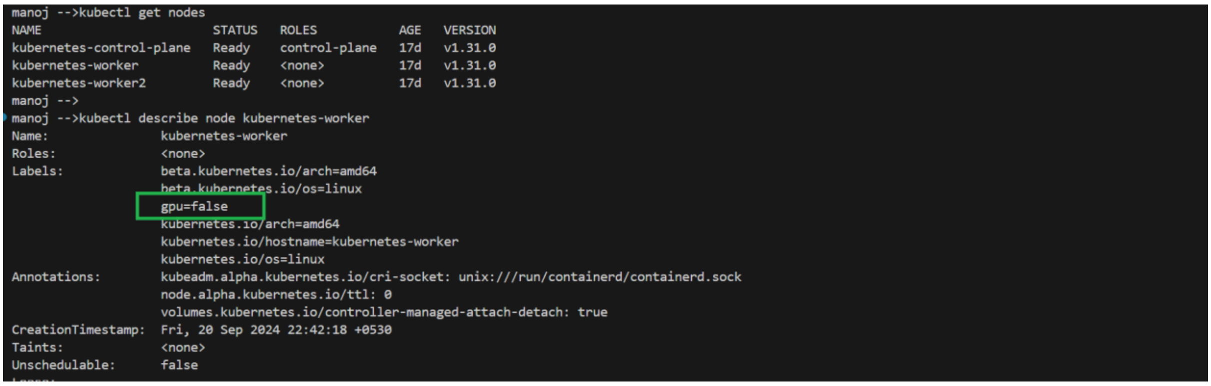
- **Node labels** are key-value pairs attached to nodes. You can add labels to nodes to categorize them (e.g., by hardware, software, or environment).
- A **Node Selector** in a pod's specification matches the node labels. The pod will only be scheduled on nodes that have the specified label(s).





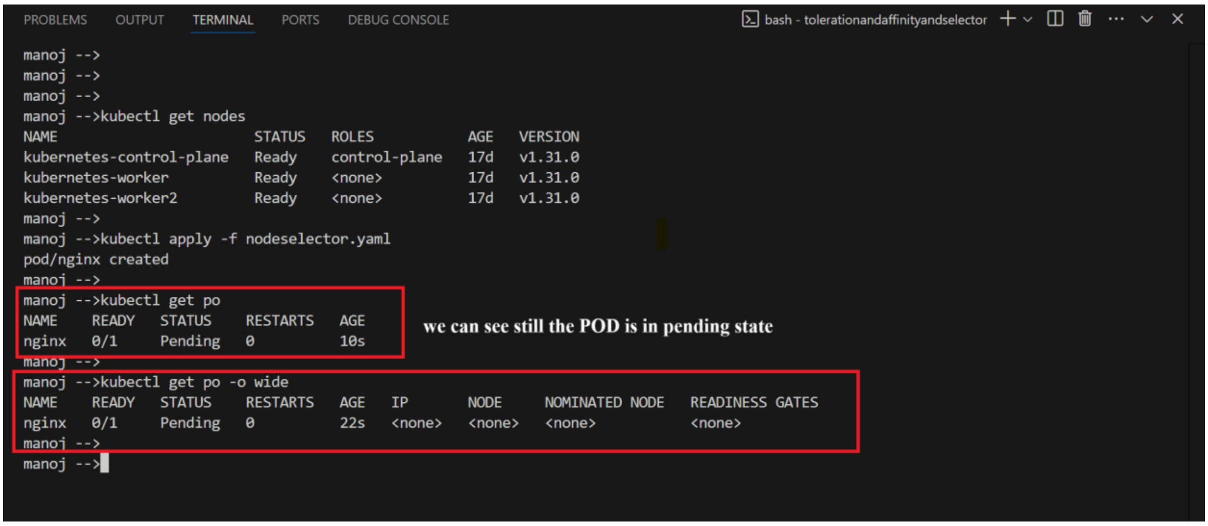
```yaml
apiVersion: v1
kind: Pod
metadata:
  labels:
    run: nginx
  name: nginx
spec:
  containers:
  - image: nginx
    name: nginx

  nodeSelector:
    gpu: "false"
```

POD will be scheduled on the matching label of the node, where 'gpu=false'



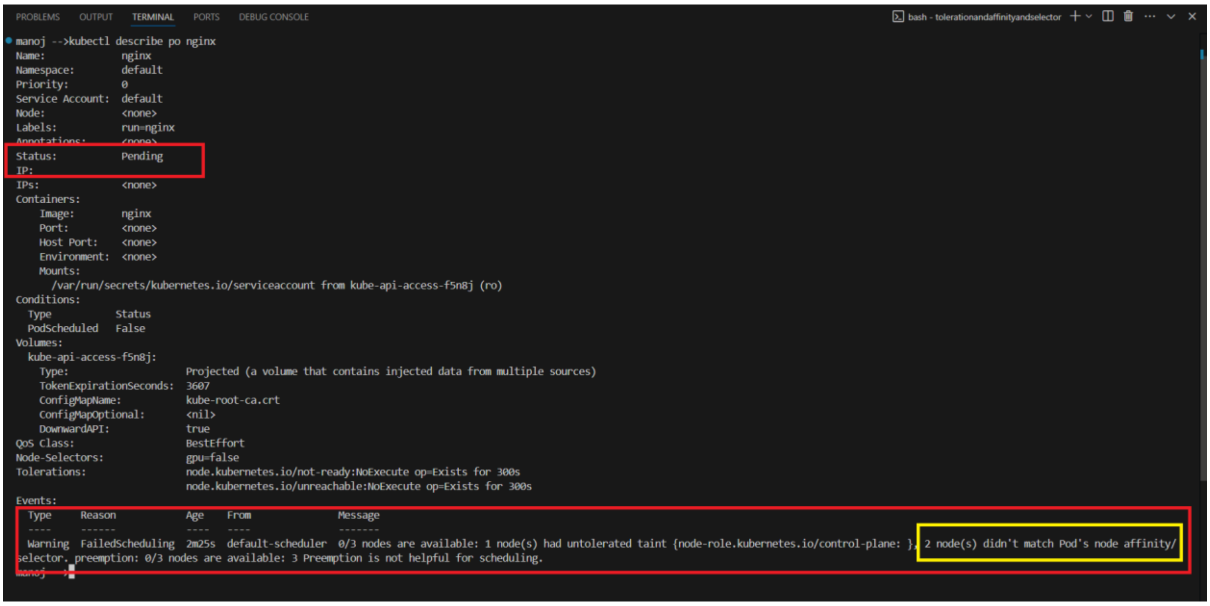
```
manoj -->kubectl get nodes
NAME                      STATUS   ROLES           AGE   VERSION
kubernetes-control-plane  Ready    control-plane   17d   v1.31.0
kubernetes-worker         Ready    <none>          17d   v1.31.0
kubernetes-worker2        Ready    <none>          17d   v1.31.0
manoj -->
manoj -->kubectl describe node kubernetes-worker
Name:               kubernetes-worker
Roles:              <none>
Labels:             beta.kubernetes.io/arch=amd64
                    beta.kubernetes.io/os=linux
                    gpu=false
                    kubernetes.io/arch=amd64
                    kubernetes.io/hostname=kubernetes-worker
                    kubernetes.io/os=linux
Annotations:        kubeadm.alpha.kubernetes.io/cri-socket: unix:///run/containerd/containerd.sock
                    node.alpha.kubernetes.io/ttl: 0
                    volumes.kubernetes.io/controller-managed-attach-detach: true
CreationTimestamp:  Fri, 20 Sep 2024 22:42:18 +0530
Taints:             <none>
Unschedulable:      false
```

# NODE SELECTOR AND NODE AFFINITY

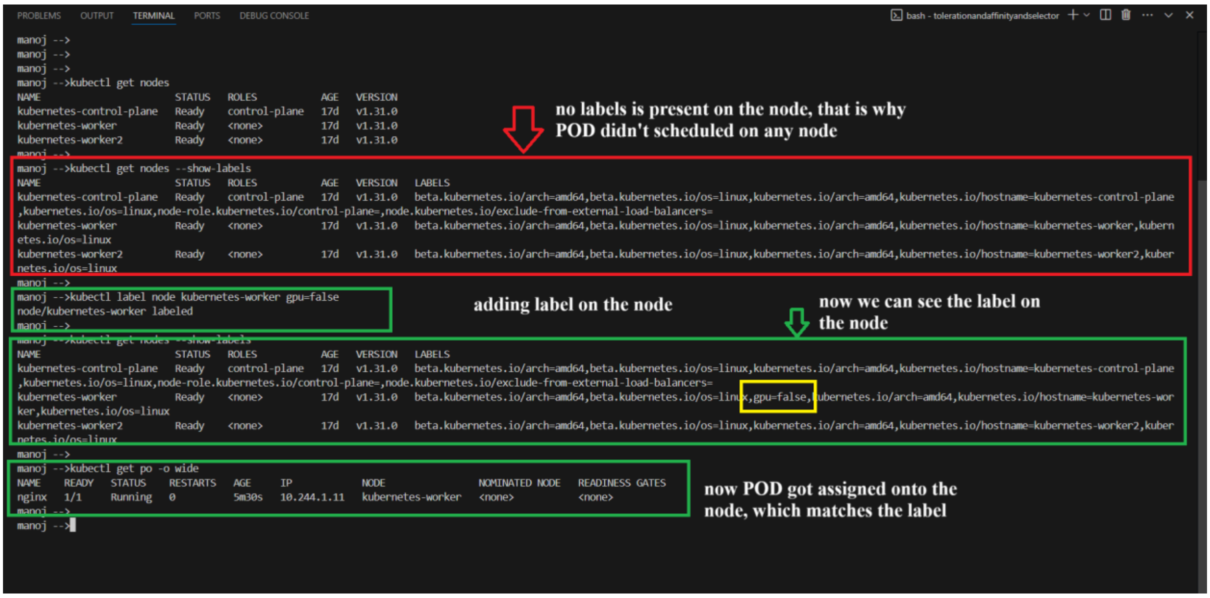


```
manoj -->
manoj -->
manoj -->
manoj -->kubectl get nodes
NAME                      STATUS   ROLES           AGE   VERSION
kubernetes-control-plane  Ready    control-plane   17d   v1.31.0
kubernetes-worker         Ready    <none>          17d   v1.31.0
kubernetes-worker2        Ready    <none>          17d   v1.31.0
manoj -->
manoj -->kubectl apply -f nodeselector.yaml
pod/nginx created
manoj -->
manoj -->kubectl get po
NAME    READY   STATUS    RESTARTS   AGE
nginx   0/1     Pending   0          10s
manoj -->
manoj -->kubectl get po -o wide
NAME    READY   STATUS    RESTARTS   AGE   IP       NODE     NOMINATED NODE   READINESS GATES
nginx   0/1     Pending   0          22s   <none>   <none>   <none>           <none>
manoj -->
manoj -->
```

we can see still the POD is in pending state



```
manoj -->kubectl describe po nginx
Name:             nginx
Namespace:        default
Priority:         0
Service Account:  default
Node:             <none>
Labels:           run=nginx
Annotations:      <none>
Status:           Pending
IP:
IPs:              <none>
Containers:
    Image:        nginx
    Port:         <none>
    Host Port:    <none>
    Environment:  <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-f5n8j (ro)
Conditions:
  Type           Status
  PodScheduled   False
Volumes:
  kube-api-access-f5n8j:
    Type:                    Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds:  3607
    ConfigMapName:           kube-root-ca.crt
    ConfigMapOptional:       <nil>
    DownwardAPI:             true
QoS Class:                   BestEffort
Node-Selectors:              gpu=false
Tolerations:                 node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
                             node.kubernetes.io/unreachable:NoExecute op=Exists for 300s
Events:
  Type     Reason            Age    From               Message
  ----     ------            ----   ----               -------
  Warning  FailedScheduling  2m25s  default-scheduler  0/3 nodes are available: 1 node(s) had untolerated taint {node-role.kubernetes.io/control-plane: }, 2 node(s) didn't match Pod's node affinity/
selector. preemption: 0/3 nodes are available: 3 Preemption is not helpful for scheduling.
manoj -->
```



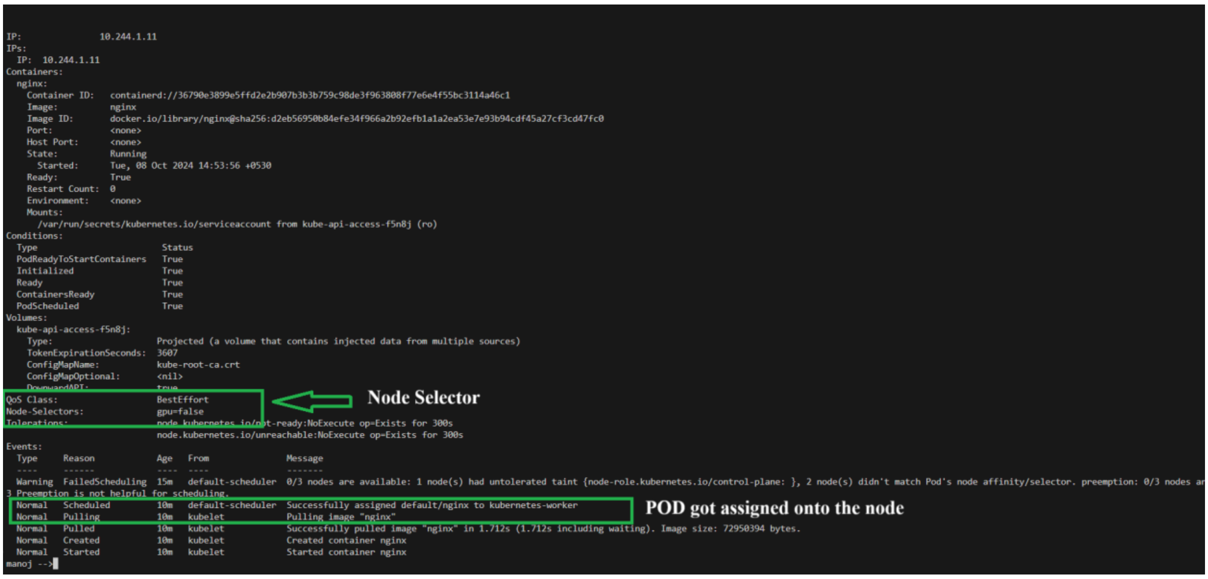
```
manoj -->
manoj -->
manoj -->
manoj -->kubectl get nodes
NAME                      STATUS   ROLES           AGE   VERSION
kubernetes-control-plane  Ready    control-plane   17d   v1.31.0
kubernetes-worker         Ready    <none>          17d   v1.31.0
kubernetes-worker2        Ready    <none>          17d   v1.31.0
manoj -->
manoj -->kubectl get nodes --show-labels
NAME                      STATUS   ROLES           AGE   VERSION   LABELS
kubernetes-control-plane  Ready    control-plane   17d   v1.31.0   beta.kubernetes.io/arch=amd64,beta.kubernetes.io/os=linux,kubernetes.io/arch=amd64,kubernetes.io/hostname=kubernetes-control-plane
,kubernetes.io/os=linux,node-role.kubernetes.io/control-plane=,node.kubernetes.io/exclude-from-external-load-balancers=
kubernetes-worker         Ready    <none>          17d   v1.31.0   beta.kubernetes.io/arch=amd64,beta.kubernetes.io/os=linux,kubernetes.io/arch=amd64,kubernetes.io/hostname=kubernetes-worker,kubern
etes.io/os=linux
kubernetes-worker2        Ready    <none>          17d   v1.31.0   beta.kubernetes.io/arch=amd64,beta.kubernetes.io/os=linux,kubernetes.io/arch=amd64,kubernetes.io/hostname=kubernetes-worker2,kuber
netes.io/os=linux
manoj -->
manoj -->kubectl label node kubernetes-worker gpu=false
node/kubernetes-worker labeled
manoj -->
manoj -->kubectl get nodes --show-labels
NAME                      STATUS   ROLES           AGE   VERSION   LABELS
kubernetes-control-plane  Ready    control-plane   17d   v1.31.0   beta.kubernetes.io/arch=amd64,beta.kubernetes.io/os=linux,kubernetes.io/arch=amd64,kubernetes.io/hostname=kubernetes-control-plane
,kubernetes.io/os=linux,node-role.kubernetes.io/control-plane=,node.kubernetes.io/exclude-from-external-load-balancers=
kubernetes-worker         Ready    <none>          17d   v1.31.0   beta.kubernetes.io/arch=amd64,beta.kubernetes.io/os=linux,gpu=false,kubernetes.io/arch=amd64,kubernetes.io/hostname=kubernetes-wor
ker,kubernetes.io/os=linux
kubernetes-worker2        Ready    <none>          17d   v1.31.0   beta.kubernetes.io/arch=amd64,beta.kubernetes.io/os=linux,kubernetes.io/arch=amd64,kubernetes.io/hostname=kubernetes-worker2,kuber
netes.io/os=linux
manoj -->
manoj -->kubectl get po -o wide
NAME    READY   STATUS    RESTARTS   AGE     IP            NODE                NOMINATED NODE   READINESS GATES
nginx   1/1     Running   0          5m30s   10.244.1.11   kubernetes-worker   <none>           <none>
manoj -->
manoj -->
```

no labels is present on the node, that is why POD didn't scheduled on any node

adding label on the node

now we can see the label on the node

now POD got assigned onto the node, which matches the label



```
IP:           10.244.1.11
IPs:
  IP:  10.244.1.11
Containers:
  nginx:
    Container ID:   containerd://36790e3899e5ffd2e2b907b3b3b759c98de3f963808f77e6e4f55bc3114a46c1
    Image:          nginx
    Image ID:       docker.io/library/nginx@sha256:d2eb56950b84efe34f966a2b92efb1a1a2ea53e7e93b94cdf45a27cf3cd47fc0
    Port:           <none>
    Host Port:      <none>
    State:          Running
      Started:      Tue, 08 Oct 2024 14:53:56 +0530
    Ready:          True
    Restart Count:  0
    Environment:    <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-f5n8j (ro)
Conditions:
  Type                        Status
  PodReadyToStartContainers   True
  Initialized                 True
  Ready                       True
  ContainersReady             True
  PodScheduled                True
Volumes:
  kube-api-access-f5n8j:
    Type:                    Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds:  3607
    ConfigMapName:           kube-root-ca.crt
    ConfigMapOptional:       <nil>
    DownwardAPI:             true
QoS Class:                   BestEffort
Node-Selectors:              gpu=false
Tolerations:                 node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
                             node.kubernetes.io/unreachable:NoExecute op=Exists for 300s
Events:
  Type     Reason            Age   From               Message
  ----     ------            ----  ----               -------
  Warning  FailedScheduling  15m   default-scheduler  0/3 nodes are available: 1 node(s) had untolerated taint {node-role.kubernetes.io/control-plane: }, 2 node(s) didn't match Pod's node affinity/selector. preemption: 0/3 nodes ar
3 Preemption is not helpful for scheduling.
  Normal   Scheduled         10m   default-scheduler  Successfully assigned default/nginx to kubernetes-worker
  Normal   Pulling           10m   kubelet            Pulling image "nginx"
  Normal   Pulled            10m   kubelet            Successfully pulled image "nginx" in 1.712s (1.712s including waiting). Image size: 72950394 bytes.
  Normal   Created           10m   kubelet            Created container nginx
  Normal   Started           10m   kubelet            Started container nginx
manoj -->
```

Node Selector

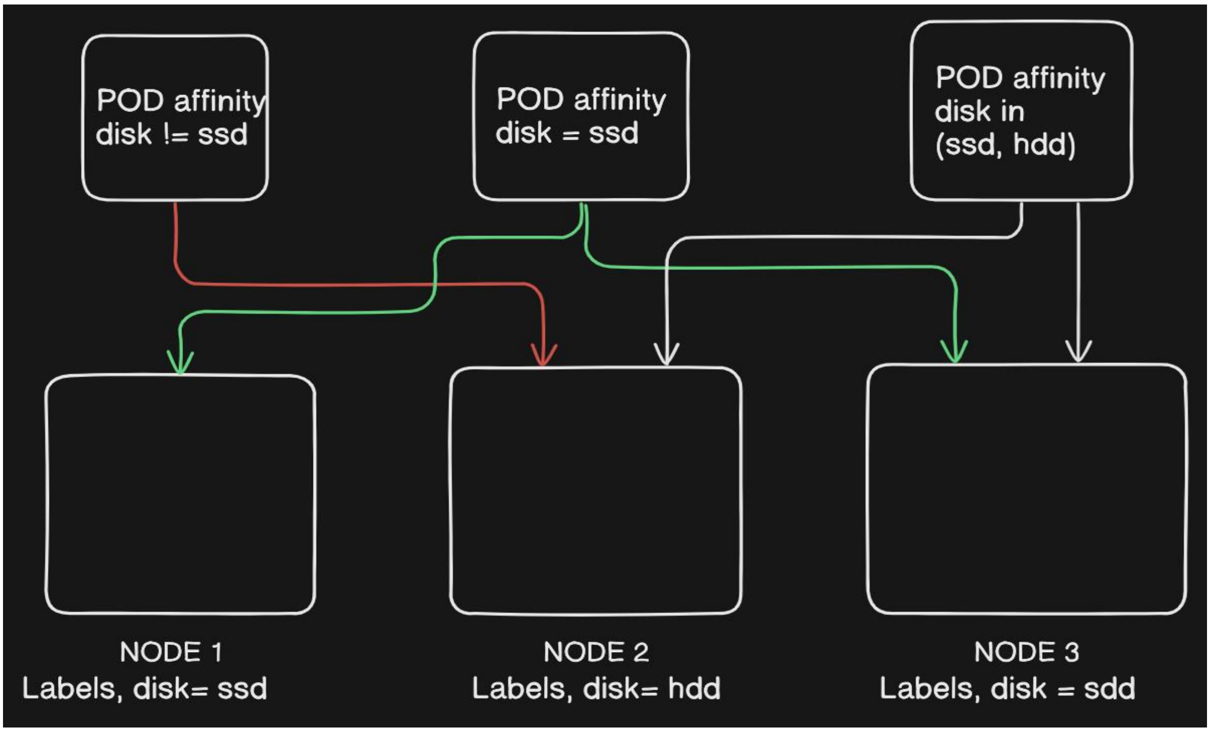
POD got assigned onto the node

### Limitations

- **Node Selector** is a simple, exact match. If you need more complex placement logic (such as multiple conditions or soft constraints), you might want to use **Node Affinity**, which offers more flexibility for scheduling decisions.

---

**Node Affinity** in Kubernetes is an advanced scheduling feature that provides more flexible rules for controlling which nodes a pod can be scheduled on, compared to the simpler **Node Selector**. It allows you to express both hard and soft constraints and supports more complex matching logic based on node labels.



## Key Features of Node Affinity

1. **Hard constraints (requiredDuringSchedulingIgnoredDuringExecution)**: These are mandatory rules. If a node does not meet these conditions, the pod will not be scheduled on that node. This behavior is similar to nodeSelector, but with more complex matching options.

2. **Soft constraints (preferredDuringSchedulingIgnoredDuringExecution)**: These are "preferences." Kubernetes will try to place the pod on a node that meets the soft constraint, but it's not a strict requirement. If no such nodes are available, the pod will still be scheduled on other nodes that don't meet the preference.

3. **Match Expressions**: Unlike nodeSelector, which only allows exact matches, node affinity allows you to use expressions such as:

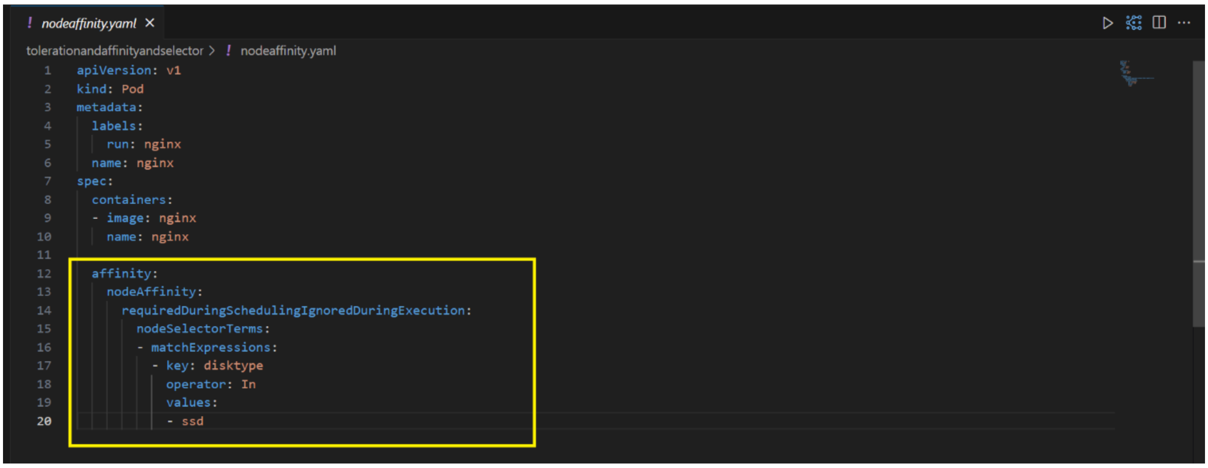
   ○ **In**: Match nodes with any of the listed values for a key.

   ○ **NotIn**: Exclude nodes with any of the listed values for a key.

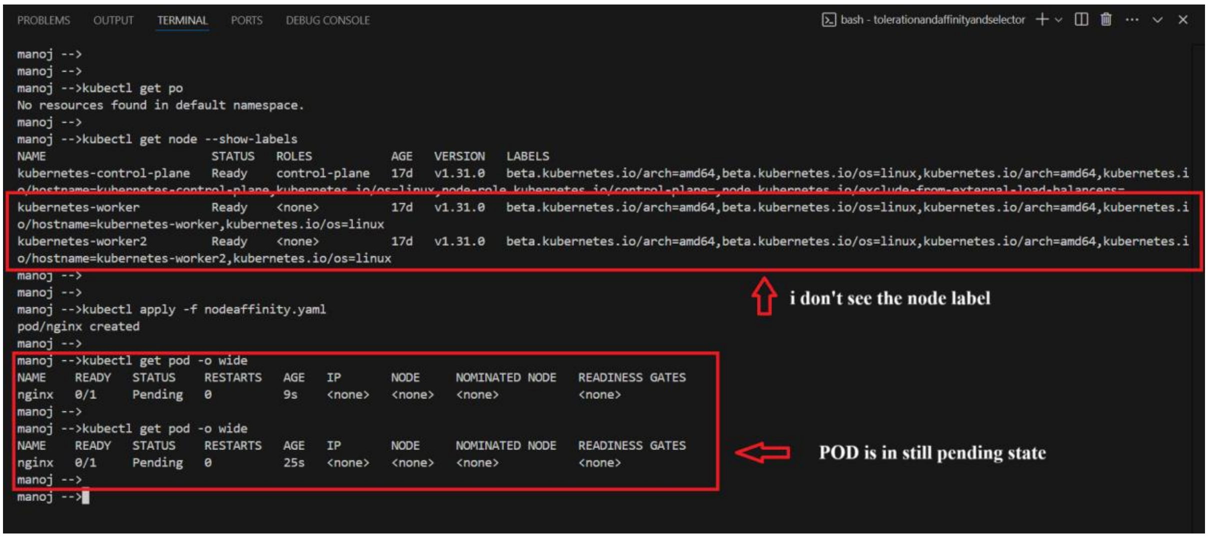
   ○ **Exists**: Match nodes that have the specified key, regardless of its value.

   ○ **DoesNotExist**: Exclude nodes that have the specified key.

   ○ **Gt/Lt**: Match nodes with values greater than or less than a specific number (for numerical label values).



```yaml
apiVersion: v1
kind: Pod
metadata:
  labels:
    run: nginx
  name: nginx
spec:
  containers:
  - image: nginx
    name: nginx

  affinity:
    nodeAffinity:
      requiredDuringSchedulingIgnoredDuringExecution:
        nodeSelectorTerms:
        - matchExpressions:
          - key: disktype
            operator: In
            values:
            - ssd
```

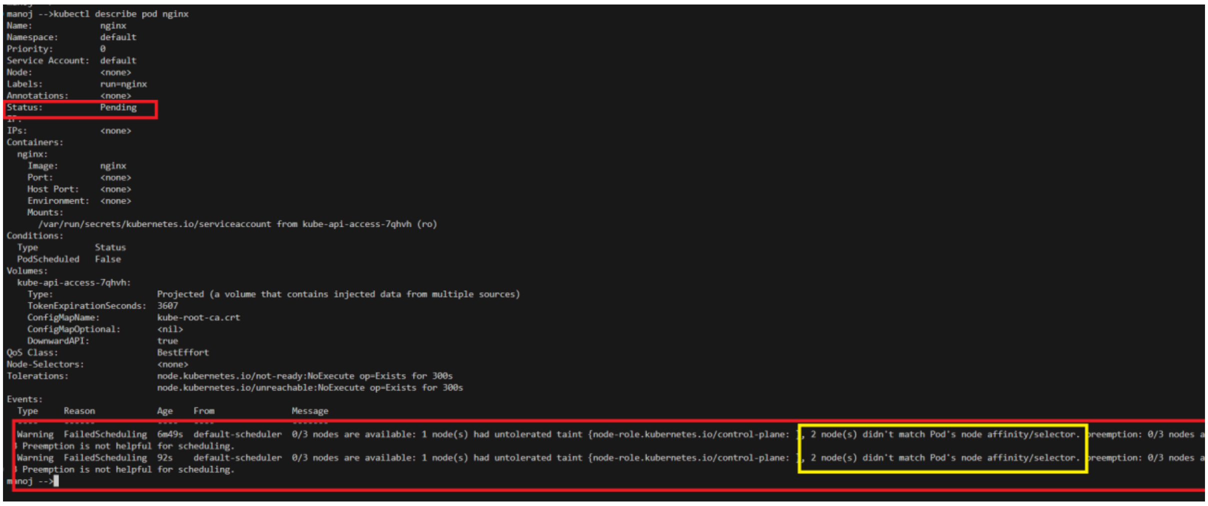


```
manoj -->
manoj -->
manoj -->kubectl get po
No resources found in default namespace.
manoj -->
manoj -->kubectl get node --show-labels
NAME                       STATUS   ROLES           AGE   VERSION   LABELS
kubernetes-control-plane   Ready    control-plane   17d   v1.31.0   beta.kubernetes.io/arch=amd64,beta.kubernetes.io/os=linux,kubernetes.io/arch=amd64,kubernetes.i
o/hostname=kubernetes-control-plane,kubernetes.io/os=linux,node-role.kubernetes.io/control-plane=,node.kubernetes.io/exclude-from-external-load-balancers=
kubernetes-worker          Ready    <none>          17d   v1.31.0   beta.kubernetes.io/arch=amd64,beta.kubernetes.io/os=linux,kubernetes.io/arch=amd64,kubernetes.i
o/hostname=kubernetes-worker,kubernetes.io/os=linux
kubernetes-worker2         Ready    <none>          17d   v1.31.0   beta.kubernetes.io/arch=amd64,beta.kubernetes.io/os=linux,kubernetes.io/arch=amd64,kubernetes.i
o/hostname=kubernetes-worker2,kubernetes.io/os=linux
manoj -->
manoj -->
manoj -->kubectl apply -f nodeaffinity.yaml
pod/nginx created
manoj -->
manoj -->kubectl get pod -o wide
NAME    READY   STATUS    RESTARTS   AGE   IP       NODE     NOMINATED NODE   READINESS GATES
nginx   0/1     Pending   0          9s    <none>   <none>   <none>           <none>
manoj -->
manoj -->kubectl get pod -o wide
NAME    READY   STATUS    RESTARTS   AGE   IP       NODE     NOMINATED NODE   READINESS GATES
nginx   0/1     Pending   0          25s   <none>   <none>   <none>           <none>
manoj -->
manoj -->
```

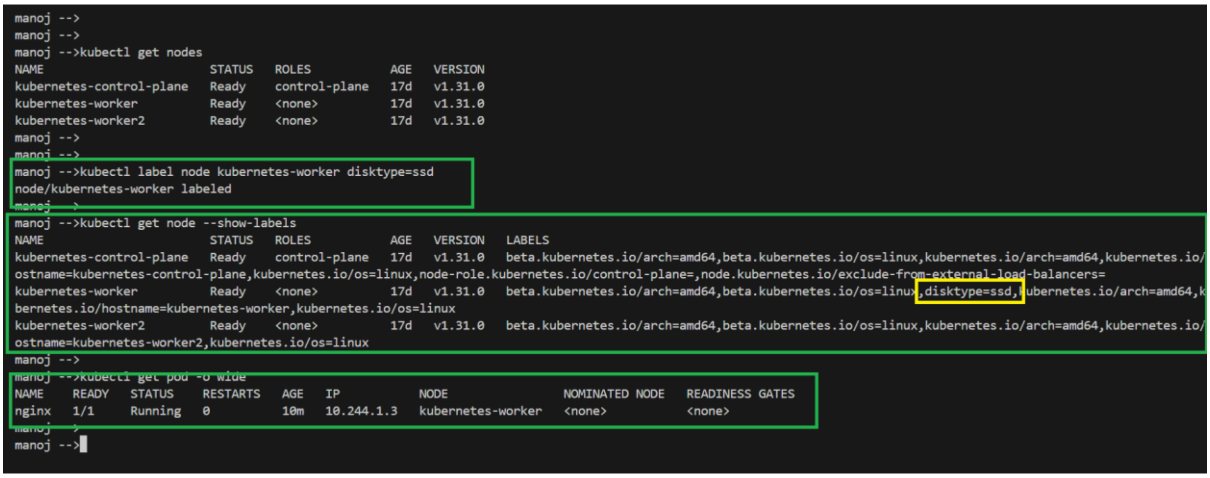
i don't see the node label

POD is in still pending state

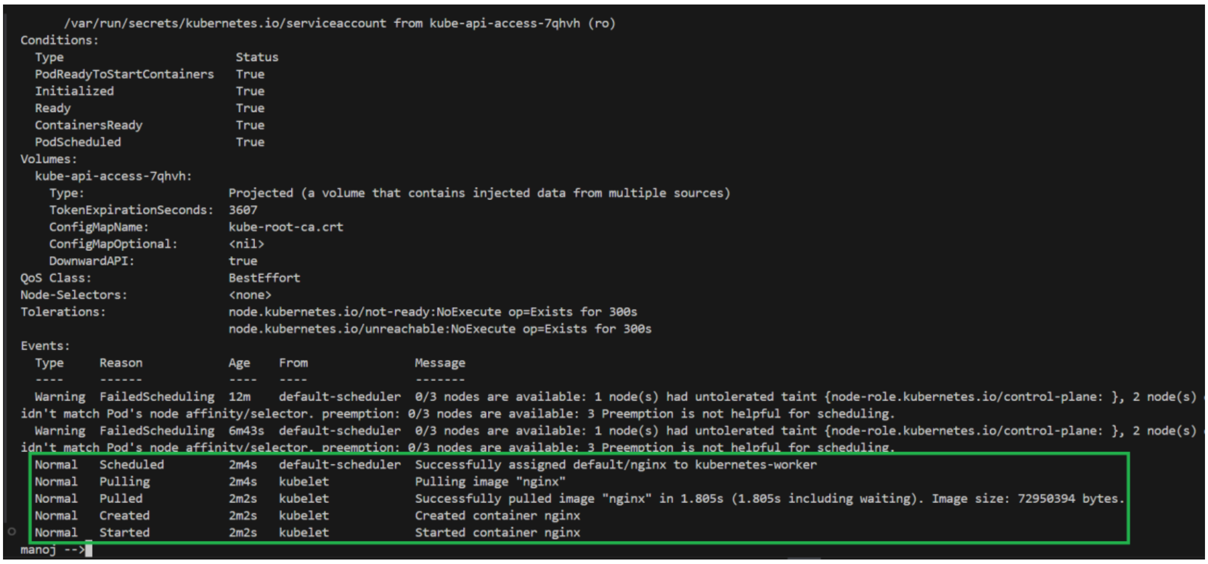
# NODE SELECTOR AND NODE AFFINITY



```
manoj -->kubectl describe pod nginx
Name:             nginx
Namespace:        default
Priority:         0
Service Account:  default
Node:             <none>
Labels:           run=nginx
Annotations:      <none>
Status:           Pending
IP:
IPs:              <none>
Containers:
  nginx:
    Image:        nginx
    Port:         <none>
    Host Port:    <none>
    Environment:  <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-7qhvh (ro)
Conditions:
  Type           Status
  PodScheduled   False
Volumes:
  kube-api-access-7qhvh:
    Type:                    Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds:  3607
    ConfigMapName:           kube-root-ca.crt
    ConfigMapOptional:       <nil>
    DownwardAPI:             true
QoS Class:                   BestEffort
Node-Selectors:              <none>
Tolerations:                 node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
                             node.kubernetes.io/unreachable:NoExecute op=Exists for 300s

Events:
  Type     Reason            Age    From               Message
  ----     ------            ----   ----               -------
  Warning  FailedScheduling  6m49s  default-scheduler  0/3 nodes are available: 1 node(s) had untolerated taint {node-role.kubernetes.io/control-plane: }, 2 node(s) didn't match Pod's node affinity/selector. preemption: 0/3 nodes a
3 Preemption is not helpful for scheduling.
  Warning  FailedScheduling  92s    default-scheduler  0/3 nodes are available: 1 node(s) had untolerated taint {node-role.kubernetes.io/control-plane: }, 2 node(s) didn't match Pod's node affinity/selector. preemption: 0/3 nodes a
3 Preemption is not helpful for scheduling.
manoj -->
```



```
manoj -->
manoj -->
manoj -->kubectl get nodes
NAME                      STATUS   ROLES           AGE   VERSION
kubernetes-control-plane  Ready    control-plane   17d   v1.31.0
kubernetes-worker         Ready    <none>          17d   v1.31.0
kubernetes-worker2        Ready    <none>          17d   v1.31.0
manoj -->
manoj -->
manoj -->kubectl label node kubernetes-worker disktype=ssd
node/kubernetes-worker labeled
manoj -->
manoj -->kubectl get node --show-labels
NAME                      STATUS   ROLES           AGE   VERSION   LABELS
kubernetes-control-plane  Ready    control-plane   17d   v1.31.0   beta.kubernetes.io/arch=amd64,beta.kubernetes.io/os=linux,kubernetes.io/arch=amd64,kubernetes.io/
ostname=kubernetes-control-plane,kubernetes.io/os=linux,node-role.kubernetes.io/control-plane=,node.kubernetes.io/exclude-from-external-load-balancers=
kubernetes-worker         Ready    <none>          17d   v1.31.0   beta.kubernetes.io/arch=amd64,beta.kubernetes.io/os=linux,disktype=ssd,kubernetes.io/arch=amd64,k
bernetes.io/hostname=kubernetes-worker,kubernetes.io/os=linux
kubernetes-worker2        Ready    <none>          17d   v1.31.0   beta.kubernetes.io/arch=amd64,beta.kubernetes.io/os=linux,kubernetes.io/arch=amd64,kubernetes.io/
ostname=kubernetes-worker2,kubernetes.io/os=linux
manoj -->
manoj -->kubectl get pod -o wide
NAME    READY   STATUS    RESTARTS   AGE   IP           NODE                NOMINATED NODE   READINESS GATES
nginx   1/1     Running   0          10m   10.244.1.3   kubernetes-worker   <none>           <none>
manoj -->
manoj -->
```



```
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-7qhvh (ro)
Conditions:
  Type                        Status
  PodReadyToStartContainers   True
  Initialized                 True
  Ready                       True
  ContainersReady             True
  PodScheduled                True
Volumes:
  kube-api-access-7qhvh:
    Type:                    Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds:  3607
    ConfigMapName:           kube-root-ca.crt
    ConfigMapOptional:       <nil>
    DownwardAPI:             true
QoS Class:                   BestEffort
Node-Selectors:              <none>
Tolerations:                 node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
                             node.kubernetes.io/unreachable:NoExecute op=Exists for 300s
Events:
  Type     Reason            Age    From               Message
  ----     ------            ----   ----               -------
  Warning  FailedScheduling  12m    default-scheduler  0/3 nodes are available: 1 node(s) had untolerated taint {node-role.kubernetes.io/control-plane: }, 2 node(s) d
idn't match Pod's node affinity/selector. preemption: 0/3 nodes are available: 3 Preemption is not helpful for scheduling.
  Warning  FailedScheduling  6m43s  default-scheduler  0/3 nodes are available: 1 node(s) had untolerated taint {node-role.kubernetes.io/control-plane: }, 2 node(s) d
idn't match Pod's node affinity/selector. preemption: 0/3 nodes are available: 3 Preemption is not helpful for scheduling.
  Normal   Scheduled         2m4s   default-scheduler  Successfully assigned default/nginx to kubernetes-worker
  Normal   Pulling           2m4s   kubelet            Pulling image "nginx"
  Normal   Pulled            2m2s   kubelet            Successfully pulled image "nginx" in 1.805s (1.805s including waiting). Image size: 72950394 bytes.
  Normal   Created           2m2s   kubelet            Created container nginx
  Normal   Started           2m2s   kubelet            Started container nginx
manoj -->
```

**Note:** we use node affinity, taint and toleration together to make sure pod are accumulating in the nodes that are meant for it.

Eg: if we have large node and that node is run’s on particular type of workload like GPU specific workload or AIML specific workload or node with high performance that is only meant to run data warehousing workload etc, in those case we use this together.