Job Search documentation (https://cloud.google.com/talent-solution/job-search/)

Documentation

Tenant basics

Tenants (https://cloud.google.com/talent-solution/job-search/docs/reference/rest/v4beta1/projects.tenants) are entities that own any jobs and companies objects assigned to them. They create a middle layer of organization between a Google Cloud Platform project and your uploaded data. Tenants allow you to isolate different groupings of your data without the need for multiple projects by preventing any data from being shared across tenancy barriers. Multi-tenancy is useful in situations where you may have more than one customer and don't want to share data between them, but would like to maintain a single GCP project for internal billing and reporting. For example:

- Job site providers building job sites for organizations with multiple subsidiary companies.
- Hiring agencies building applicant tracking systems for multiple businesses.

Each Google Cloud Platform project is assigned a `tenant_id` for a single default tenant. Optionally, you can change the default by creating more than one tenant within a given project.

Tenants are fully isolated from one another. All APIs ask for only a single tenant to prevent data being queried across multiple tenants in a single API call. Machine learning (ML) similarly treats tenants as discrete units and does not cross tenancy barriers. A project can support as many tenants as required.

Default tenants

In Job Search v4beta1 and newer, a Tenant entity is required. All projects are assigned a `tenant_id` for a single default tenant. If you don't want to use multiple tenants, you can either:

1. **Use your project's default tenant (Recommended)**. You don't need to reference the tenant specifically in order to do this. You can use the format `projects/{project_id}/jobs/{job_id}` and the Cloud Talent Solution backend will assume that you're using the default tenant.
2. **Create a single tenant** (https://cloud.google.com/talent-solution/job-search/docs/tenants#create_a_tenant) **and use it in place of the default tenant**. If you create your own tenant, you need to reference it: `projects/{project_id}/tenants/{tenant_id}/jobs/{job_id}`.

The default `tenant_id` is unique to each project and is never overwritten or replaced if you create additional tenants. You cannot call CRUD methods on the default tenant.

Note: If you choose to use the default tenant, the CTS backend uses your assigned `tenant_id` value automatically without you needing to enter it (Option 1, above). If you would like to find your default `tenant_id` value, create a company (https://cloud.google.com/talent-solution/job-search/docs/companies#create_a_company) without a tenant. The default `tenant_id` appears in the company resource name in the response.

Created tenants (optional)

Creating your own tenants is optional. Each Job Search project is assigned a default tenant by the CTS backend. If you do not want to use multi-tenancy to separate subdivisions of your data, we recommend that you use the default tenant.

Create a tenant

You are required to create a unique `externalId` value and assign it to the new tenant. The call then returns a unique `name` assigned by our backend system, also assigned to that tenant. Be sure to record and store both the `name` and `externalId` values, as they are used to update/delete/reference.

The code sample below creates a new tenant:

JAVA PHP PYTHON RUBY

:/MAIN/JAVA/COM/GOOGLE/CLOUD/EXAMPLES/TALENT/V4BETA1/JOBSEARCHCREATETENANT.JAVA)

FEEDBACK (#)

```
/*
 * Please include the following imports to run this sample.
 *
 * import com.google.cloud.talent.v4beta1.CreateTenantRequest;
 * import com.google.cloud.talent.v4beta1.ProjectName;
 * import com.google.cloud.talent.v4beta1.Tenant;
 * import com.google.cloud.talent.v4beta1.TenantServiceClient;
 */

/** Create Tenant for scoping resources, e.g. companies and jobs */
public static void sampleCreateTenant(String projectId, String externalId) {
  try (TenantServiceClient tenantServiceClient = TenantServiceClient.create()) {
    // projectId = "Your Google Cloud Project ID";
    // externalId = "Your Unique Identifier for Tenant";
    ProjectName parent = ProjectName.of(projectId);
    Tenant tenant = Tenant.newBuilder().setExternalId(externalId).build();
    CreateTenantRequest request =
        CreateTenantRequest.newBuilder().setParent(parent.toString()).setTenant(te
    Tenant response = tenantServiceClient.createTenant(request);
    System.out.println("Created Tenant");
    System.out.printf("Name: %s\n", response.getName());
    System.out.printf("External ID: %s\n", response.getExternalId());
  } catch (Exception exception) {
    System.err.println("Failed to create the client due to: " + exception);
  }
}
```

Retrieve a tenant

JAVA PHP PYTHON RUBY

For more on installing and creating a Cloud Talent Solution client, see Cloud Talent Solution Client Libraries (https://cloud.google.com/talent-solution/job-search/docs/libraries).

SRC/MAIN/JAVA/COM/GOOGLE/CLOUD/EXAMPLES/TALENT/V4BETA1/JOBSEARCHGETTENANT.JAVA)

FEEDBACK (#)

```
/*
 * Please include the following imports to run this sample.
 *
 * import com.google.cloud.talent.v4beta1.GetTenantRequest;
 * import com.google.cloud.talent.v4beta1.Tenant;
 * import com.google.cloud.talent.v4beta1.TenantName;
 * import com.google.cloud.talent.v4beta1.TenantServiceClient;
 */

/** Get Tenant by name */
public static void sampleGetTenant(String projectId, String tenantId) {
  try (TenantServiceClient tenantServiceClient = TenantServiceClient.create()) {
    // projectId = "Your Google Cloud Project ID";
    // tenantId = "Your Tenant ID";
    TenantName name = TenantName.of(projectId, tenantId);
    GetTenantRequest request = GetTenantRequest.newBuilder().setName(name.toString
    Tenant response = tenantServiceClient.getTenant(request);
    System.out.printf("Name: %s\n", response.getName());
    System.out.printf("External ID: %s\n", response.getExternalId());
  } catch (Exception exception) {
    System.err.println("Failed to create the client due to: " + exception);
  }
}
```

List tenants

JAVA PHP PYTHON RUBY

For more on installing and creating a Cloud Talent Solution client, see Cloud Talent Solution Client Libraries (https://cloud.google.com/talent-solution/job-search/docs/libraries).

RC/MAIN/JAVA/COM/GOOGLE/CLOUD/EXAMPLES/TALENT/V4BETA1/JOBSEARCHLISTTENANTS.JAVA)

FEEDBACK (#)

```
/*
 * Please include the following imports to run this sample.
 *
 * import com.google.cloud.talent.v4beta1.ListTenantsRequest;
 * import com.google.cloud.talent.v4beta1.ProjectName;
 * import com.google.cloud.talent.v4beta1.Tenant;
```

```
 * import com.google.cloud.talent.v4beta1.TenantServiceClient;
 */

/** List Tenants */
public static void sampleListTenants(String projectId) {
  try (TenantServiceClient tenantServiceClient = TenantServiceClient.create()) {
    // projectId = "Your Google Cloud Project ID";
    ProjectName parent = ProjectName.of(projectId);
    ListTenantsRequest request =
        ListTenantsRequest.newBuilder().setParent(parent.toString()).build();
    for (Tenant responseItem : tenantServiceClient.listTenants(request).iterateAll
      System.out.printf("Tenant Name: %s\n", responseItem.getName());
      System.out.printf("External ID: %s\n", responseItem.getExternalId());
    }
  } catch (Exception exception) {
    System.err.println("Failed to create the client due to: " + exception);
  }
}
```

Delete a tenant

JAVA PHP PYTHON RUBY

For more on installing and creating a Cloud Talent Solution client, see Cloud Talent Solution Client Libraries (https://cloud.google.com/talent-solution/job-search/docs/libraries).

:/MAIN/JAVA/COM/GOOGLE/CLOUD/EXAMPLES/TALENT/V4BETA1/JOBSEARCHDELETETENANT.JAVA)

FEEDBACK (#)

```
/*
 * Please include the following imports to run this sample.
 *
 * import com.google.cloud.talent.v4beta1.DeleteTenantRequest;
 * import com.google.cloud.talent.v4beta1.TenantName;
 * import com.google.cloud.talent.v4beta1.TenantServiceClient;
 */

/** Delete Tenant */
public static void sampleDeleteTenant(String projectId, String tenantId) {
  try (TenantServiceClient tenantServiceClient = TenantServiceClient.create()) {
    // projectId = "Your Google Cloud Project ID";
```

```
    // tenantId = "Your Tenant ID)";
    TenantName name = TenantName.of(projectId, tenantId);
    DeleteTenantRequest request =
        DeleteTenantRequest.newBuilder().setName(name.toString()).build();
    tenantServiceClient.deleteTenant(request);
    System.out.println("Deleted Tenant.");
  } catch (Exception exception) {
    System.err.println("Failed to create the client due to: " + exception);
  }
}
```


Last updated December 3, 2019.