

# Root query

## FIELDS

---

viewer(token: [String!](#)): [ViewerQuery](#)

\_\_debug: [DjangoDebug](#)

# Mutations

## FIELDS

---

createToken(username: [String!](#), password: [String!](#)): [CreateToken](#)

createPerson(name: [String!](#), primaryPhone: [String](#), image: [String](#), note: [String](#), token: [String!](#), type: [String](#), email: [String](#), primaryLocationId: [Int!](#)): [CreatePerson](#)

createOrganization(name: [String!](#), primaryPhone: [String](#), image: [String](#), note: [String](#), token: [String!](#), type: [String!](#), email: [String](#), primaryLocationId: [Int!](#)): [CreateOrganization](#)

updatePerson(name: [String](#), image: [String](#), email: [String](#), note: [String](#), token: [String!](#), id: [Int!](#), primaryLocationId: [Int!](#)): [UpdatePerson](#)

updateOrganization(name: [String](#), image: [String](#), email: [String](#), note: [String](#), token: [String!](#), id: [Int!](#), primaryLocationId: [Int!](#)): [UpdateOrganization](#)

createProcess(name: [String!](#), plannedStart: [String!](#), note: [String](#), token: [String!](#), scopelId: [Int!](#), plannedDuration: [Int!](#)): [CreateProcess](#)

updateProcess(name: [String](#), plannedStart: [String](#), note: [String](#), token: [String!](#), scopelId: [Int](#), plannedDuration: [Int](#), isFinished: [Boolean](#), id: [Int!](#)): [UpdateProcess](#)

deleteProcess(token: [String!](#), id: [Int!](#)): [DeleteProcess](#)

createCommitment(receiverId: [Int](#), committedNumericValue: [String!](#), planId: [Int](#), isPlanDeliverable: [Boolean](#), committedUnitId: [Int!](#), inputOfId: [Int](#), providerId: [Int](#), plannedStart: [String](#), due: [String!](#), note: [String](#), token: [String!](#), scopelId: [Int](#), involvesId: [Int](#), action: [String!](#), outputOfId: [Int](#), committedResourceClassifiedAsId: [Int!](#)): [CreateCommitment](#)

updateCommitment(receiverId: [Int](#), committedNumericValue: [String](#), inputOfId: [Int](#), committedUnitId: [Int](#), committedResourceClassifiedAsId: [Int](#), providerId: [Int](#), plannedStart: [String](#), due: [String](#), note: [String](#), token: [String!](#), scopelId: [Int](#), involvesId: [Int](#), committedOn: [String](#), action: [String](#), isFinished: [Boolean](#), outputOfId: [Int](#), id: [Int!](#)): [UpdateCommitment](#)

`deleteCommitment(token: String!, id: Int!): DeleteCommitment`

`createEconomicEvent(note: String, providerId: Int, fulfillsCommitmentId: Int, createResource: Boolean, inputOfId: Int, url: String, affectsId: Int, affectedUnitId: Int, resourceImage: String, receiverId: Int, resourceNote: String, resourceCurrentLocationId: Int, start: String, token: String!, scopelId: Int, requestDistribution: Boolean, action: String, affectedNumericValue: String!, outputOfId: Int, affectedResourceClassifiedAsId: Int, resourceTrackingIdentifier: String): CreateEconomicEvent`

`updateEconomicEvent(receiverId: Int, fulfillsCommitmentId: Int, affectedNumericValue: String, inputOfId: Int, url: String, affectedUnitId: Int, affectsId: Int, providerId: Int, note: String, start: String, token: String!, scopelId: Int, action: String, requestDistribution: Boolean, outputOfId: Int, affectedResourceClassifiedAsId: Int, id: Int!): UpdateEconomicEvent`

`deleteEconomicEvent(token: String!, id: Int!): DeleteEconomicEvent`

`createPlan(note: String, token: String!, name: String!, due: String!): CreatePlan`

`updatePlan(note: String, token: String!, id: Int!, name: String, due: String): UpdatePlan`

`deletePlan(token: String!, id: Int!): DeletePlan`

`updateEconomicResource(trackingIdentifier: String, image: String, resourceClassifiedAsId: Int, note: String, currentLocationId: Int, token: String!, id: Int!): UpdateEconomicResource`

`deleteEconomicResource(token: String!, id: Int!): DeleteEconomicResource`

`createNotificationSetting(token: String!, notificationTypeId: Int!, agentId: Int, send: Boolean!): CreateNotificationSetting`

`updateNotificationSetting(token: String!, id: Int!, send: Boolean): UpdateNotificationSetting`

## ViewerQuery

### FIELDS

---

`myAgent: Agent`

`agent(id: Int): Agent`

`allAgents: [Agent]`

`usersIsAuthorizedToCreate(scopelId: Int): Boolean`

`agentRelationship(id: Int): AgentRelationship`

allAgentRelationships: [[AgentRelationship](#)]

agentRelationshipRole(id: [Int](#)): [AgentRelationshipRole](#)

allAgentRelationshipRoles: [[AgentRelationshipRole](#)]

organization(id: [Int](#)): [Organization](#)

allOrganizations: [[Organization](#)]

fcOrganizations(joiningStyle: [String](#), visibility: [String](#)): [[Organization](#)]

organizationTypes: [[OrganizationType](#)]

organizationClassification(id: [Int](#)): [OrganizationClassification](#)

allOrganizationClassifications: [[OrganizationClassification](#)]

person(id: [Int](#)): [Person](#)

allPeople: [[Person](#)]

economicResource(id: [Int](#)): [EconomicResource](#)

allEconomicResources: [[EconomicResource](#)]

process(id: [Int](#)): [Process](#)

allProcesses: [[Process](#)]

exchangeAgreement(id: [Int](#)): [ExchangeAgreement](#)

allExchangeAgreements: [[ExchangeAgreement](#)]

transfer(id: [Int](#)): [Transfer](#)

allTransfers: [[Transfer](#)]

economicEvent(id: [Int](#)): [EconomicEvent](#)

allEconomicEvents: [[EconomicEvent](#)]

quantityValue(id: [Int](#)): [QuantityValue](#)

`unit(id: Int): Unit`

`allUnits: [Unit]`

`resourceClassification(id: Int): ResourceClassification`

`allResourceClassifications: [ResourceClassification]`

`resourceClassificationsByProcessCategory(category: EconomicResourceProcessCategory): [ResourceClassification]`

`resourceClassificationsByAction(action: Action): [ResourceClassification]`

`resourceClassificationsByFacetValues(facetValues: String): [ResourceClassification]`

`facet(id: Int): Facet`

`allFacets: [Facet]`

`processClassification(id: Int): ProcessClassification`

`allProcessClassifications: [ProcessClassification]`

`commitment(id: Int): Commitment`

`allCommitments: [Commitment]`

`plan(id: Int): Plan`

`allPlans: [Plan]`

`place(id: Int): Place`

`allPlaces: [Place]`

`notificationSetting(id: Int): NotificationSetting`

`allNotificationSettings: [NotificationSetting]`

`notificationType(id: Int): NotificationType`

`allNotificationTypes: [NotificationType]`

# DjangoDebug

## FIELDS

---

sql: [\[DjangoDebugSQL\]](#)

# CreateToken

## FIELDS

---

token: [String](#)

# CreatePerson

## FIELDS

---

person: [Person](#)

# CreateOrganization

## FIELDS

---

organization: [Organization](#)

# UpdatePerson

## FIELDS

---

person: [Person](#)

# UpdateOrganization

## FIELDS

---

organization: [Organization](#)

# CreateProcess

## FIELDS

---

process: [Process](#)

# UpdateProcess

## FIELDS

---

process: [Process](#)

## DeleteProcess

### FIELDS

---

process: [Process](#)

## CreateCommitment

### FIELDS

---

commitment: [Commitment](#)

## UpdateCommitment

### FIELDS

---

commitment: [Commitment](#)

## DeleteCommitment

### FIELDS

---

commitment: [Commitment](#)

## CreateEconomicEvent

### FIELDS

---

economicEvent: [EconomicEvent](#)

## UpdateEconomicEvent

### FIELDS

---

economicEvent: [EconomicEvent](#)

## DeleteEconomicEvent

### FIELDS

---

economicEvent: [EconomicEvent](#)

## CreatePlan

### FIELDS

---

plan: [Plan](#)

# UpdatePlan

## FIELDS

---

plan: [Plan](#)

# DeletePlan

## FIELDS

---

plan: [Plan](#)

# UpdateEconomicResource

## FIELDS

---

economicResource: [EconomicResource](#)

# DeleteEconomicResource

## FIELDS

---

economicResource: [EconomicResource](#)

# CreateNotificationSetting

## FIELDS

---

notificationSetting: [NotificationSetting](#)

# UpdateNotificationSetting

## FIELDS

---

notificationSetting: [NotificationSetting](#)

# Agent

## IMPLEMENTED BY

---

[Organization](#)

[Person](#)

## FIELDS

---

id: [String](#)

name: [String](#)

type: [String](#)

image: [String](#)

note: [String](#)

primaryLocation: [Place](#)

primaryPhone: [String](#)

email: [String](#)

ownedEconomicResources(category: [EconomicResourceCategory](#), resourceClassificationId: [Int](#), page: [Int](#)): [[EconomicResource](#)]

agentProcesses(isFinished: [Boolean](#)): [[Process](#)]

agentPlans(isFinished: [Boolean](#), year: [Int](#), month: [Int](#)): [[Plan](#)]

agentEconomicEvents(latestNumberOfDays: [Int](#), requestDistribution: [Boolean](#)): [[EconomicEvent](#)]

agentCommitments(latestNumberOfDays: [Int](#)): [[Commitment](#)]

agentRelationships(roleId: [Int](#), category: [AgentRelationshipCategory](#)): [[AgentRelationship](#)]

agentRoles: [[AgentRelationshipRole](#)]

agentRecipes: [[ResourceClassification](#)]

faircoinAddress: [String](#)

agentNotificationSettings: [[NotificationSetting](#)]

memberRelationships: [[AgentRelationship](#)]

eventsCount(year: [Int](#), month: [Int](#)): [Int](#)

eventHoursCount(year: [Int](#), month: [Int](#)): [Int](#)

eventPeopleCount(year: [Int](#), month: [Int](#)): [Int](#)

## AgentRelationship



**FIELDS**

---

id: [ID!](#)

subject: [Agent](#)

object: [Agent](#)

relationship: [AgentRelationshipRole](#)

## AgentRelationshipRole

**FIELDS**

---

id: [ID!](#)

label: [String](#)

inverseLabel: [String](#)

category: [AgentRelationshipCategory](#)

## Organization

**IMPLEMENTS**

---

[Agent](#)

**FIELDS**

---

id: [String](#)

name: [String](#)

type: [String](#)

image: [String](#)

note: [String](#)

primaryLocation: [Place](#)

primaryPhone: [String](#)

email: [String](#)

ownedEconomicResources(category: [EconomicResourceCategory](#), resourceClassificationId: [Int](#), page: [Int](#)): [[EconomicResource](#)]

agentProcesses(isFinished: [Boolean](#)): [[Process](#)]

agentPlans(isFinished: [Boolean](#), year: [Int](#), month: [Int](#)): [[Plan](#)]

agentEconomicEvents(latestNumberOfDays: [Int](#), requestDistribution: [Boolean](#)): [[EconomicEvent](#)]

agentCommitments(latestNumberOfDays: [Int](#)): [[Commitment](#)]

agentRelationships(roleId: [Int](#), category: [AgentRelationshipCategory](#)): [[AgentRelationship](#)]

agentRoles: [[AgentRelationshipRole](#)]

agentRecipes: [[ResourceClassification](#)]

faircoinAddress: [String](#)

agentNotificationSettings: [[NotificationSetting](#)]

memberRelationships: [[AgentRelationship](#)]

eventsCount(year: [Int](#), month: [Int](#)): [Int](#)

eventHoursCount(year: [Int](#), month: [Int](#)): [Int](#)

eventPeopleCount(year: [Int](#), month: [Int](#)): [Int](#)

## OrganizationType

### FIELDS

---

id: [ID!](#)

name: [String!](#)

## OrganizationClassification

### FIELDS

---

id: [ID!](#)

name: [String!](#)

note: [String](#)

## Person

### IMPLEMENTS

---

[Agent](#)

### FIELDS

---

id: [String](#)

name: [String](#)

type: [String](#)

image: [String](#)

note: [String](#)

primaryLocation: [Place](#)

primaryPhone: [String](#)

email: [String](#)

ownedEconomicResources(category: [EconomicResourceCategory](#), resourceClassificationId: [Int](#), page: [Int](#)): [[EconomicResource](#)]

agentProcesses(isFinished: [Boolean](#)): [[Process](#)]

agentPlans(isFinished: [Boolean](#), year: [Int](#), month: [Int](#)): [[Plan](#)]

agentEconomicEvents(latestNumberOfDays: [Int](#), requestDistribution: [Boolean](#)): [[EconomicEvent](#)]

agentCommitments(latestNumberOfDays: [Int](#)): [[Commitment](#)]

agentRelationships(roleId: [Int](#), category: [AgentRelationshipCategory](#)): [[AgentRelationship](#)]

agentRoles: [[AgentRelationshipRole](#)]

agentRecipes: [[ResourceClassification](#)]

faircoinAddress: [String](#)

agentNotificationSettings: [[NotificationSetting](#)]

memberRelationships: [[AgentRelationship](#)]

eventsCount(year: [Int](#), month: [Int](#)): [Int](#)

eventHoursCount(year: [Int](#), month: [Int](#)): [Int](#)

eventPeopleCount(year: [Int](#), month: [Int](#)): [Int](#)

## EconomicResource

### FIELDS

---

id: [ID!](#)

resourceClassifiedAs: [ResourceClassification](#)

trackingIdentifier: [String](#)

image: [String](#)

currentQuantity: [QuantityValue](#)

note: [String](#)

category: [String](#)

currentLocation: [Place](#)

createdDate: [String](#)

transfers: [[Transfer](#)]

resourceContacts: [[Agent](#)]

## Process

### FIELDS

---

id: [ID!](#)

name: [String!](#)

scope: [Agent](#)

plannedStart: [String](#)

plannedDuration: [String](#)

isStarted: [Boolean](#)

isFinished: [Boolean](#)

processClassifiedAs: [ProcessClassification](#)

note: [String](#)

isDeletable: [Boolean](#)

inputs(action: [Action](#)): [[EconomicEvent](#)]

outputs(action: [Action](#)): [[EconomicEvent](#)]

unplannedEconomicEvents(action: [Action](#)): [[EconomicEvent](#)]

committedInputs(action: [Action](#)): [[Commitment](#)]

committedOutputs(action: [Action](#)): [[Commitment](#)]

nextProcesses: [[Process](#)]

previousProcesses: [[Process](#)]

workingAgents: [[Agent](#)]

processPlan: [Plan](#)

userIsAuthorizedToUpdate: [Boolean](#)

userIsAuthorizedToDelete: [Boolean](#)

## ExchangeAgreement

### FIELDS

---

id: [ID!](#)

name: [String!](#)

plannedStart: [String](#)

scope: [Agent](#)

note: [String](#)

transfers: [[Transfer](#)]

involvedAgents: [[Agent](#)]

## Transfer

### FIELDS

---

id: [ID!](#)

name: [String!](#)

exchangeAgreement: [ExchangeAgreement](#)

plannedStart: [String](#)

scope: [Agent](#)

note: [String](#)

provider: [Agent](#)

receiver: [Agent](#)

resourceClassifiedAs: [ResourceClassification](#)

giveResource: [EconomicResource](#)

takeResource: [EconomicResource](#)

transferQuantity: [QuantityValue](#)

transferDate: [String](#)

transferEconomicEvents: [[EconomicEvent](#)]

giveEconomicEvent: [EconomicEvent](#)

takeEconomicEvent: [EconomicEvent](#)

transferCommitments: [[Commitment](#)]

giveCommitment: [Commitment](#)

takeCommitment: [Commitment](#)

involvedAgents: [[Agent](#)]

## EconomicEvent

### FIELDS

---

id: [ID!](#)

action: [String](#)

inputOf: [Process](#)

outputOf: [Process](#)

provider: [Agent](#)

receiver: [Agent](#)

scope: [Agent](#)

affects: [EconomicResource](#)

affectedQuantity: [QuantityValue](#)

start: [String](#)

url: [String](#)

requestDistribution: [Boolean](#)

note: [String](#)

fulfills: [[Fulfillment](#)]

usersIsAuthorizedToUpdate: [Boolean](#)

usersIsAuthorizedToDelete: [Boolean](#)

## QuantityValue

### FIELDS

---

numericValue: [Float](#)

unit: [Unit](#)

## Unit

### FIELDS

---

id: [ID!](#)

name: [String!](#)

symbol: [String!](#)

## ResourceClassification

### FIELDS

---

id: [ID!](#)

name: [String!](#)

image: [String](#)

note: [String](#)

category: [String](#)

processCategory: [String](#)

classificationResources: [[EconomicResource](#)]

## EconomicResourceProcessCategory

### POSSIBLE ENUM VALUES

---

NONE



CITED  
CONSUMED  
PRODUCED  
USED

## Action

### POSSIBLE ENUM VALUES

---

NONE  
ACCEPT  
CITE  
CONSUME  
GIVE  
IMPROVE  
PRODUCE  
TAKE  
USE  
WORK

## Facet

### FIELDS

---

id: [ID!](#)

name: [String!](#)

description: [String](#)

facetValues: [[FacetValue](#)]

## ProcessClassification

### FIELDS

---

id: [ID!](#)

name: [String!](#)

note: [String](#)

scope: [Agent](#)

estimatedDuration: [String](#)

## Commitment

### FIELDS

---

id: [ID!](#)

action: [String](#)

inputOf: [Process](#)

outputOf: [Process](#)

provider: [Agent](#)

receiver: [Agent](#)

scope: [Agent](#)

resourceClassifiedAs: [ResourceClassification](#)

involves: [EconomicResource](#)

committedQuantity: [QuantityValue](#)

committedOn: [String](#)

plannedStart: [String](#)

due: [String](#)

isFinished: [Boolean](#)

plan: [Plan](#)

isPlanDeliverable: [Boolean](#)

forPlanDeliverable: [Commitment](#)

note: [String](#)

fulfilledBy(requestDistribution: [Boolean](#)): [[Fulfillment](#)]

involvedAgents: [\[Agent\]](#)

isDeletable: [Boolean](#)

usersIsAuthorizedToUpdate: [Boolean](#)

usersIsAuthorizedToDelete: [Boolean](#)

## Plan

### FIELDS

---

id: [ID!](#)

plannedOn: [String](#)

due: [String](#)

note: [String](#)

name: [String](#)

scope: [\[Agent\]](#)

planProcesses(year: [Int](#), month: [Int](#)): [\[Process\]](#)

workingAgents: [\[Agent\]](#)

plannedNonWorkInputs: [\[Commitment\]](#)

plannedOutputs: [\[Commitment\]](#)

nonWorkInputs: [\[EconomicEvent\]](#)

outputs: [\[EconomicEvent\]](#)

kanbanState: [String](#)

## Place

### FIELDS

---

id: [ID!](#)

name: [String!](#)

address: [String!](#)

latitude: [Float](#)

longitude: [Float](#)

note: [String](#)

placeResources: [[EconomicResource](#)]

placeAgents: [[Agent](#)]

## NotificationSetting

### FIELDS

---

id: [ID!](#)

send: [Boolean!](#)

agent: [Agent](#)

notificationType: [NotificationType](#)

## NotificationType

### FIELDS

---

id: [ID!](#)

label: [String!](#)

display: [String!](#)

description: [String!](#)

## DjangoDebugSQL

### FIELDS

---

vendor: [String](#)

alias: [String](#)

sql: [String](#)

duration: [Float](#)

rawSql: [String](#)

params: [String](#)

startTime: [Float](#)

stopTime: [Float](#)

isSlow: [Boolean](#)

isSelect: [Boolean](#)

transId: [String](#)

transStatus: [String](#)

isoLevel: [String](#)

encoding: [String](#)

## EconomicResourceCategory

### POSSIBLE ENUM VALUES

---

NONE

CURRENCY

INVENTORY

WORK

## AgentRelationshipCategory

### POSSIBLE ENUM VALUES

---

NONE

LEGALPARTNER

MEMBER

PART

PEER

TRADINGPARTNER

# Fulfillment

## FIELDS

---

id: [ID!](#)

fulfilledBy: [EconomicEvent!](#)

fulfills: [Commitment!](#)

fulfilledQuantity: [QuantityValue!](#)

note: [String](#)

# FacetValue

## FIELDS

---

id: [ID!](#)

value: [String!](#)

description: [String](#)

facet: [\[Facet\]](#)

# String

The String scalar type represents textual data, represented as UTF-8 character sequences. The String type is most often used by GraphQL to represent free-form human-readable text.

# Int

The Int scalar type represents non-fractional signed whole numeric values. Int can represent values between  $-(2^{53} - 1)$  and  $2^{53} - 1$  since represented in JSON as double-precision floating point numbers specified by [IEEE 754](#).

# Boolean

The Boolean scalar type represents true or false.

# ID

The ID scalar type represents a unique identifier, often used to refetch an object or as key for a cache. The ID type appears in a JSON response as a String; however, it is not intended to be human-readable.

When expected as an input type, any string (such as "4") or integer (such as 4) input value will be accepted as an ID.

## Float

The Float scalar type represents signed double-precision fractional values as specified by [IEEE 754](#).