

Sonar

This will contain all Sonar-related information.

- [Sonar API Meeting \(01/26/2024\)](#)

Sonar API Meeting

(01/26/2024)

<https://docs.sonar.expert/billing>

queries vs mutations

query:

- standard query
- queries can only return up to 10000 results at once, pagination is required after that

mutation:

- adding info
- createVoiceService for voice services

creating services:

- need id

API treats all money values as cents

ID of created entity are always returned

credits applied to balance

Examples (note that all account IDs, etc are for demo purposes):

query all customers and their IDs:

```
query accounts {  
  accounts(paginator: {page: 1, records_per_page: 9999}) {  
    entities {  
      id,  
      name
```

```
}  
}  
}
```

query all services and their IDs:

```
query service {  
  services {  
    entities {  
      id,  
      name,  
      type  
    }  
  }  
}
```

create a new service:

```
mutation createService($input:CreateRecurringServiceMutationInput) {  
  createRecurringService(input:$input) {  
    enabled,  
    name,  
    type,  
    application,  
    amount,  
    display_if_zero,  
    company_id,  
    general_ledger_code_id,  
    tax_definition_id,  
    reverse_tax_definition_id,  
    id,  
    sonar_unique_id,  
    created_at,  
    updated_at  
  }  
}
```

create a new one-time charge:

```
mutation onetimecharge {  
  createAccountOneTimeTransaction(  

```

```
input: {
  service_id: 105,
  account_id: 3,
  description: "Description",
  quantity: 1,
  price_override: 1000
}
) {
  id
}
}
```

query for listing all invoices for a customer:

```
query customerInvoice {
  accounts(id: 2) {
    entities {
      id,
      name,
      invoices {
        entities {
          id,
          date,
          due_date,
          late_fee_applied,
          remaining_due,
          debits {
            entities {
              id,
              amount
            }
          }
          credits {
            entities {
              id,
              amount
            }
          }
        }
      }
    }
  }
}
```

query account transactions:

```
query accounttransactions {  
  transactions(account_id: 2) {  
    entities {  
      id,  
      amount  
    }  
  }  
}
```
